Condition Number Analysis of Logistic Regression, and its Implications for First-Order Solution Methods

Robert M. Freund (MIT)

joint with Paul Grigas (Berkeley) and Rahul Mazumder (MIT)

Penn State University, March 2018

How can optimization inform statistics (and machine learning)?

Paper in preparation (this talk):

Condition Number Analysis of Logistic Regression, and its Implications for First-Order Solution Methods

A “cousin” paper of ours:

A New Perspective on Boosting in Linear Regression via Subgradient Optimization and Relatives

Outline

- Optimization primer: two basic first-order methods for convex optimization

Logistic regression perspectives: statistics "vs." machine learning

A pair of condition numbers for the logistic regression problem:

- when the sample data is non-separable:

  a condition number for the degree of non-separability of the dataset
  informing the convergence guarantees of Greedy Coordinate Descent and Stochastic Gradient Descent (SGD)
  guarantees on reaching linear convergence (thanks to Bach)

  when the sample data is separable:

  a condition number for the degree of separability of the dataset
  informing convergence guarantees to deliver an approximate maximum margin classifier

Review of Two Basic First-Order Methods for Convex Optimization

Two Basic First-Order Methods for Convex Optimization:

- Greedy Coordinate Descent method

  “go in the best coordinate direction”

  Stochastic Gradient Descent (SGD) method

  “go in the direction of the negative of the stochastic estimate of the gradient”

Convex Optimization

The problem of interest is:

$$\begin{array}{rccl} F^* & := & \min\limits_{x} & F(x) \\ & & \text{s.t.} & x \in \mathbb{R}^p \end{array}$$

where $F(\cdot)$ is differentiable and convex:

$$F(\lambda x + (1-\lambda)y) \le \lambda F(x) + (1-\lambda)F(y) \ \text{ for all } x,\ y, \text{ and all } \lambda \in [0,1]$$

Let $\|x\|$ denote the given norm on the variables $x \in \mathbb{R}^p$

Norms and Dual Norms

Let $\|x\|$ be the given norm on the variables $x \in \mathbb{R}^p$

The dual norm is $\|s\|_* := \max_x\{s^T x : \|x\| \leq 1\}$

Some common norms and their dual norms:

Name	Norm	Definition	Dual Norm
ℓ_2-norm	$\|x\|_2$	$\|x\|_2 = \sqrt{\sum_{j=1}^p \lvert x_j\rvert^2}$	$\|s\|_* = \|s\|_2$
ℓ_1-norm	$\|x\|_1$	$\|x\|_1 = \sum_{j=1}^p \lvert x_j\rvert$	$\|s\|_* = \|s\|_\infty$
ℓ_∞-norm	$\|x\|_\infty$	$\|x\|_\infty = \max\{\lvert x_1\rvert, \ldots, \lvert x_p\rvert\}$	$\|s\|_* = \|s\|_1$

Lipschitz constant for the Gradient

$$\begin{array}{rccl} F^* & := & \min\limits_{x} & F(x) \\ & & \text{s.t.} & x \in \mathbb{R}^p \end{array}$$

We say that $\nabla F(\cdot)$ is Lipschitz with parameter L_F if:

$$\|\nabla F(x) - \nabla F(y)\|_* \ \leq \ L_F\|x - y\| \quad \text{for all } x, y \in \mathbb{R}^p$$

$\| \cdot \|_*$ is the dual norm

Matrix Operator Norm

Let M be a linear operator (matrix) $M : \mathbb{R}^p \rightarrow \mathbb{R}^n$ with norm $\|x\|_a$ on $\mathbb{R}^p$ and norm $\|v\|_b$ on $\mathbb{R}^n$

The operator norm of M is given by:

$$\|M\|_{a,b} := \max_{x \neq 0} \frac{\|Mx\|_b}{\|x\|_a}$$

Greedy Coordinate Descent Method: “go in the best coordinate direction”

$$\begin{array}{rccl} F^* & := & \min\limits_{x} & F(x) \\ & & \text{s.t.} & x \in \mathbb{R}^p \end{array}$$

Greedy Coordinate Descent

Initialize at $x^0 \in \mathbb{R}^p$, $k \leftarrow 0$

At iteration k :

- Compute gradient $\nabla F(x^k)$

2 Compute

   - $j_k \in \arg \max\limits_{j \in \{1,\ldots,p\}} \{|\nabla F(x^k)_j|\}$ and
   - $d^k \leftarrow \text{sgn}(\nabla F(x^k)_{j_k}) e_{j_k}$

3 Choose step-size α_k

4 Set $x^{k+1} \qquad x^k - \alpha_k d^k$

Greedy Coordinate Descent ≡ Steepest Descent in the ℓ_1-Norm

$$\begin{array}{rccl} F^* & := & \min\limits_{x} & F(x) \\ & & \text{s.t.} & x \in \mathbb{R}^p \end{array}$$

Steepest Descent method in the ℓ_1-norm

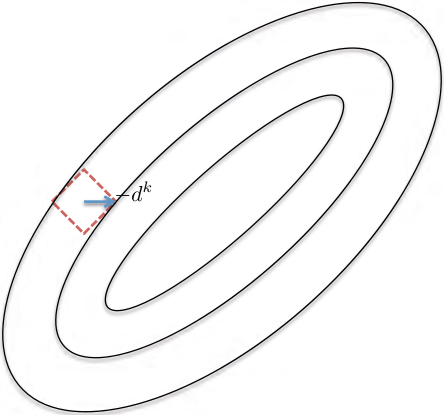
Initialize at $x^0 \in \mathbb{R}^p$, $k \leftarrow 0$

At iteration k :

1. Compute gradient $\nabla F(x^k)$
2. Compute direction: $d^k \leftarrow \arg \max\limits_{\|d\|_1 \leq 1} \{\nabla F(x^k)^T d\}$
3. Choose step-size α_k
4. Set $x^{k+1} \leftarrow x^k - \alpha_k d^k$

Greedy Coordinate Descent ≡ Steepest Descent in the ℓ_1-Norm, cont.

$$d^k \leftarrow \arg\max_{\|d\|_1 \leq 1} \{\nabla F(x^k)^T d\}$$



Metrics for Evaluating Greedy Coordinate Descent

$$\begin{array}{rrll} F^* & := & \min\limits_{x} & F(x) \\ & & \text{s.t.} & x \in \mathbb{R}^p \end{array}$$

Assume $F(\cdot)$ is convex and $\nabla F(\cdot)$ is Lipschitz with parameter L_F:

$$\|\nabla F(x) - \nabla F(y)\|_\infty \ \le \ L_F\|x - y\|_1 \quad \text{for all } x, y \in \mathbb{R}^p$$

Two sets of interest:

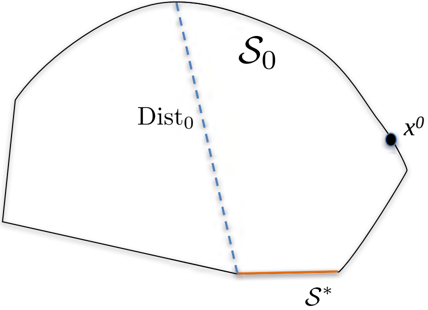
$\mathcal{S}_0 := \{x \in \mathbb{R}^p : F(x) \le F(x^0)\}$ is the level set of the initial point x^0

$\mathcal{S}^* := \{x \in \mathbb{R}^p : F(x) = F^*\}$ is the set of optimal solutions

Metrics for Evaluating Greedy Coordinate Descent, cont.

$\mathcal{S}_0 := \{x \in \mathbb{R}^p : F(x) \le F(x^0)\}$ is the level set of the initial point x^0
$\mathcal{S}^* := \{x \in \mathbb{R}^p : F(x) = F^*\}$ is the set of optimal solutions

$$\mathrm{Dist}_0 := \max_{x \in \mathcal{S}_0} \min_{x^* \in \mathcal{S}^*} \|x - x^*\|_1$$



(In high-dimensional machine learning problems, $\mathcal{S}^*$ can be very big)

Computational Guarantees for Greedy Coordinate Descent

$$\mathrm{Dist}_0 := \max_{x \in \mathcal{S}_0} \min_{x^* \in \mathcal{S}^*} \|x - x^*\|_1$$

Theorem: Objective Function Value Convergence (essentially [Beck and Tetruashvil 2014], [Nesterov 2003])

If the step-sizes are chosen using the rule:

$$\alpha_k = \frac{\|\nabla F(x^k)\|_\infty}{L_F} \quad \text{for all } k \geq 0 \ ,$$

then for each $k \geq 0$ the following inequality holds:

$$F(x^k) - F^* \ \leq \ \frac{2L_F(\mathrm{Dist}_0)^2}{\hat{K}^0 + k} \ < \ \frac{2L_F(\mathrm{Dist}_0)^2}{k}$$

where $\hat{K}^0 := \dfrac{2L_F(\mathrm{Dist}_0)^2}{F(x^0) - F^*}$.

Computational Guarantees for GCD, cont.

Theorem: Gradient Norm Convergence

For any step-size sequence $\{\alpha_k\}$ and for each $k \geq 0$, it holds that:

$$\min_{i \in \{0,\ldots,k\}} \|\nabla F(x^i)\|_\infty \ \leq \ \frac{F(x^0) - F^* + \frac{L_F}{2}\sum_{i=0}^k \alpha_i^2}{\sum_{i=0}^k \alpha_i} \ .$$

If the step-sizes are chosen using the rule:

$$\alpha_k = \frac{\|\nabla F(x^k)\|_\infty}{L_F} \quad \text{for all } k \geq 0 \ ,$$

then for each $k \geq 0$ the following inequality holds:

$$\min_{i \in \{0,\ldots,k\}} \|\nabla F(x^i)\|_\infty \ \leq \ \sqrt{\frac{2L_F(F(x^0) - F^*)}{k+1}} \quad .$$

Computational Guarantees for GCD, cont.

Theorem: Iterate Shrinkage

For any step-size sequence $\{\alpha_k\}$, it holds for each $k \geq 0$ that:

$$\|x^k - x^0\|_1 \ \leq \ \sum_{i=0}^{k-1} \alpha_i \ .$$

If the step-sizes are chosen using the rule:

$$\alpha_k = \frac{\|\nabla F(x^k)\|_\infty}{L_F} \quad \text{for all } k \geq 0 \ ,$$

then for each $k \geq 0$ it holds that:

$$\|x^k - x^0\|_1 \ \leq \ \sqrt{k}\sqrt{\frac{2(F(x^0) - F^*)}{L_F}} \ .$$

Stochastic Gradient Descent (SGD) Method

The problem of interest is:

$$\begin{array}{rcll} F^* & := & \min\limits_{x} & F(x) \\ & & \text{s.t.} & x \in \mathbb{R}^p \end{array}$$

Let $\tilde{\nabla} f(x)$ be a stochastic estimate of the gradient $\nabla F(x)$ at each x

Stochastic Gradient Descent method for minimizing $F(x)$

Initialize at $x^0 \in \mathbb{R}^p$, $k \leftarrow 0$

At iteration k :

1. Compute stochastic gradient $\tilde{\nabla} F(x^k)$
2. Choose step-size α_k
3. Set $x^{k+1} \leftarrow x^k - \alpha_k \tilde{\nabla} F(x^k)$

Stochastic Gradient Descent (SGD) Method, cont.

The canonical setting for SGD is minimizing a large sum (or average) of losses:

$$\begin{array}{rrl} F^* := & \min_x & F(x) := \frac{1}{n}\sum_{j=1}^n F_j(x) \\ & \text{s.t.} & x \in \mathbb{R}^p \end{array}$$

where $n \gg 0$ and $\tilde{\nabla} F(x)$ is computed as follows:

1. Choose $j \sim \{1, \ldots, n\}$ uniformly and independently
2. $\tilde{\nabla} F(x) \leftarrow \nabla F_j(x)$

Then the stochastic gradient is unbiased: $\mathbb{E}[\tilde{\nabla} F(x)|x] = \nabla F(x)$

Computational Guarantees for Stochastic Gradient Descent [Bertsekas, others]

Assume that
(i) the stochastic gradient is unbiased, namely

$$\mathbb{E}[\tilde{\nabla} F(x) | x] = \nabla F(x) \quad \text{for any } x \ , \ \text{and}$$

(ii) $F(\cdot)$ is G-stochastically smooth: there exists G such that:

$$\mathbb{E}[\|\tilde{\nabla} F(x)\|_2^2 \mid x] \leq G^2 \text{ for any } x$$

Theorem: Expected Convergence of Stochastic Gradient Descent

If the step-sizes are constant:

$$\alpha_k = \bar{\alpha} \quad \text{for all } k \geq 0 \ ,$$

then for each $k \geq 0$ the following inequality holds:

$$\mathbb{E}[F(\bar{x}^k)] - F^* \ \leq \ \frac{\bar{\alpha} G^2}{2} + \frac{\|x^0 - x^*\|_2^2}{2\bar{\alpha}(k+1)} \ ,$$

where $\bar{x}^k := \frac{1}{k+1} \sum_{i=0}^{k} x^i$.

Logistic Regression

Logistic Regression

- statistics perspective
- machine learning perspective

Logistic Regression: Statistics Perspective

Logistic Regression: Statistics Perspective

Logistic Regression Statistics Perspective
Example: Predicting Parole Violation

Predict P(violate parole) based on age, gender, time served, offense class, multiple convictions, NYC, etc.

```
     Violator Male  Age TimeServed Class Multiple InCity
1           0    1 49.4       3.15     D        0      1
2           1    1 26.0       5.95     D        1      0
3           0    1 24.9       2.25     D        1      0
4           0    1 52.1      29.22     A        0      0
5           0    1 35.9      12.78     A        1      1
6           0    1 25.9       1.18     C        1      1
7           0    1 19.0       0.54     D        0      0
8           0    1 43.2       1.07     C        0      1
9           0    1 31.6       1.17     E        0      0
10          0    1 40.7       4.64     B        1      1
11          0    1 53.9      21.61     A        0      1
12          0    1 28.5       3.23     D        1      0
13          0    1 36.1       3.71     D        0      1
14          0    1 48.8       1.17     D        0      0
15          0    1 37.6       4.62     C        0      0
16          0    1 42.5       1.75     D        0      1
...       ...  ...  ...        ...   ...      ...    ...
6098        0    1 55.0       0.72     E        0      0
6099        0    1 49.6      29.88     A        0      1
6100        0    1 22.4       2.85     D        0      1
6101        0    1 44.8       1.76     D        1      0
6102        0    0 45.3       1.03     E        0      0
```

Logistic Regression for Prediction

$Y \in \{-1, 1\}$ is a Bernoulli random variable:

$$P(Y = 1) = p$$

$$P(Y = -1) = 1 - p$$

$x = (x_1, \ldots, x_p) \in \mathbb{R}^p$ is the vector of independent variables

$P(Y = 1)$ depends on the values of the independent variables $x_1, \ldots, x_p$

Logistic regression model is:

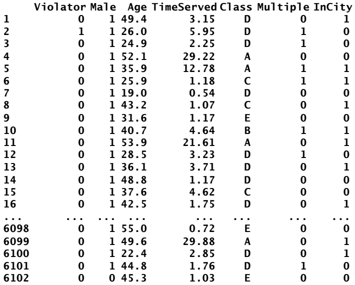
$$P(Y = 1 \mid x) \quad = \quad \frac{1}{1 + e^{-\beta^T x}}$$

Logistic Regression for Prediction, continued

Logistic regression model is:

$$P(Y = 1 \mid x) \quad = \quad \frac{1}{1 + e^{-\beta^T x}}$$

Data records are (x_i, y_i), $i = 1, \ldots, n$



	Violator	Male	Age	TimeServed	Class	Multiple	InCity
1	0	1	49.4	3.15	D	0	1
2	1	1	26.0	5.95	D	1	0
3	0	1	24.9	2.25	D	1	0
4	0	1	52.1	29.22	A	0	0
5	0	1	35.9	12.78	A	1	1
6	0	1	25.9	1.18	C	1	1
7	0	1	19.0	0.54	D	0	0
8	0	1	43.2	1.07	C	0	1
9	0	1	31.6	1.17	E	0	0
10	0	1	40.7	4.64	B	1	1
11	0	1	53.9	21.61	A	0	1
12	0	1	28.5	3.23	D	1	0
13	0	1	36.1	3.71	D	0	1
14	0	1	48.8	1.17	D	0	0
15	0	1	37.6	4.62	C	0	0
16	0	1	42.5	1.75	D	0	1
...	...	...	...	...	...	...	...
6098	0	1	55.0	0.72	E	0	0
6099	0	1	49.6	29.88	A	0	1
6100	0	1	22.4	2.85	D	0	1
6101	0	1	44.8	1.76	D	1	0
6102	0	0	45.3	1.03	E	0	0

Let us construct an estimate of β based on the data (x_i, y_i), $i = 1, \ldots, n$

Logistic Regression: Maximum Likelihood Estimation

$$\max_{\beta}\left(\prod_{y_i=1}\frac{1}{1+e^{-\beta^T x_i}}\right)\left(\prod_{y_i=-1}\left(1-\frac{1}{1+e^{-\beta^T x_i}}\right)\right)$$

$$=\max_{\beta}\left(\prod_{i=1}^{n}\frac{1}{1+e^{-y_i\beta^T x_i}}\right)$$

$$\equiv\min_{\beta}\frac{1}{n}\sum_{i=1}^{n}\ln\left(1+e^{-y_i\beta^T x_i}\right)\quad=:\quad L_n(\beta)$$

Logistic Regression Optimization Problem

Logistic regression optimization problem is:

$$\begin{array}{rccl} L_n^* & := & \min\limits_{\beta} & L_n(\beta) := \frac{1}{n}\sum_{i=1}^{n} \ln(1+\exp(-y_i\beta^T x_i)) \\ & & \text{s.t.} & \beta \in \mathbb{R}^p \end{array}$$

If $y_i = +1$, we ideally want $\beta^T x_i \gg 0$

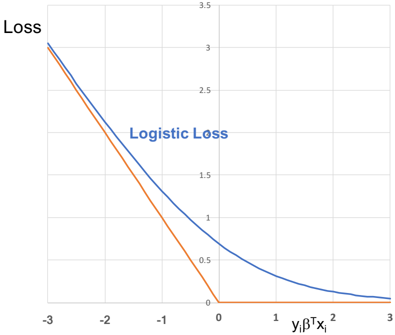
If $y_i = -1$, we ideally want $\beta^T x_i \ll 0$

Therefore we ideally want β for which $y_i\beta^T x_i \gg 0$ for very many i

Logistic Regression Optimization Problem, continued

Logistic regression optimization problem is:

$$\begin{array}{rrll} L_n^* & := & \min\limits_{\beta} & L_n(\beta) := \frac{1}{n}\sum_{i=1}^{n} \ln(1+\exp(-y_i\beta^T x_i)) \\ & & \text{s.t.} & \beta \in \mathbb{R}^p \end{array}$$



Each logistic loss term is a 1-smoothing of $\max\{0, -y_i\beta^T x_i\}$

Properties of the Logistic Loss Function

$$
\begin{array}{rccl}
L_n^* & := & \min\limits_{\beta} & L_n(\beta) := \frac{1}{n}\sum_{i=1}^n \ln(1+\exp(-y_i\beta^T x_i)) \\
 & & \text{s.t.} & \beta \in \mathbb{R}^p
\end{array}
$$

$$
\text{Denote} \quad \mathbf{X} := \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}
$$

Proposition: Lipschitz constant of the gradient of $L_n(\beta)$

$\nabla L_n(\cdot)$ is $L = \frac{1}{4n}\|\mathbf{X}\|_{1,2}^2$-Lipschitz:

$$
\|\nabla L_n(\beta) - \nabla L_n(\beta')\|_\infty \le \frac{1}{4n}\|\mathbf{X}\|_{1,2}^2\|\beta - \beta'\|_1
$$

where $\|\mathbf{X}\|_{1,2} := \max\limits_{\|\beta\|_1 \le 1} \|\mathbf{X}\beta\|_2$

Properties of the Logistic Loss Function, continued

$$\begin{array}{rlll} L_n^* & := & \min\limits_{\beta} & L_n(\beta) := \frac{1}{n}\sum_{i=1}^n \ln(1+\exp(-y_i\beta^T x_i)) \\ & & \text{s.t.} & \beta \in \mathbb{R}^p \end{array}$$

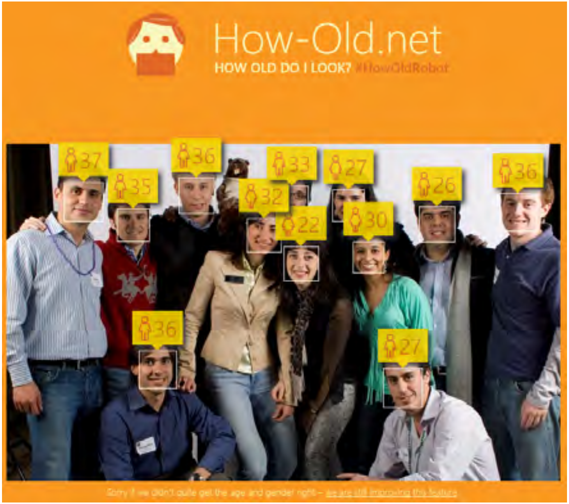
- $L_n(\cdot)$ is convex
- $L_n^* \geq 0$
- If $L_n^* = 0$, then the optimum is not attained (something is "wrong" or "very wrong")
- We will see later that "very wrong" is actually very good....
- For $\beta^0 := 0$ it holds that $L_n(\beta^0) = \ln(2)$

Logistic Regression: Machine Learning Perspective

Logistic Regression: Machine Learning Perspective

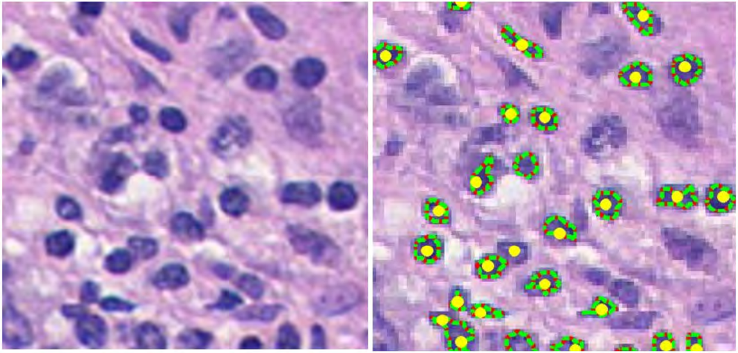
Logistic Regression Machine Learning Perspective
Example: Gender Classification

Classify (predict) gender based on image


How-Old.net
HOW OLD DO I LOOK? #HowOldRobot
37
36
35
33
27
32
22
30
26
36
36
27
Sorry if we didn't quite get the age and gender right – we are still improving this feature

Another Example: Cancer/noncancerous cells

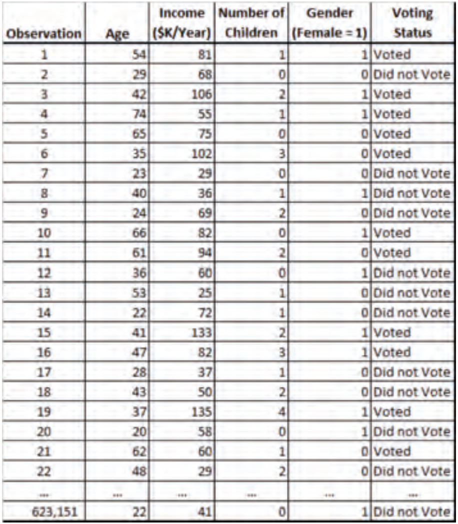
Classify (predict) cancer/noncancer cells based on image



Dots = annotations
Boundaries = detections

Another Example: Voters-nonvoters

Classify (predict) voters vs. nonvoters based on election data



Observation	Age	Income ($K/Year)	Number of Children	Gender (Female = 1)	Voting Status
1	54	81	1	1	Voted
2	29	68	0	0	Did not Vote
3	42	106	2	1	Voted
4	74	55	1	1	Voted
5	65	75	0	0	Voted
6	35	102	3	0	Voted
7	23	29	0	0	Did not Vote
8	40	36	1	1	Did not Vote
9	24	69	2	0	Did not Vote
10	66	82	0	1	Voted
11	61	94	2	0	Voted
12	36	60	0	1	Did not Vote
13	53	25	1	0	Did not Vote
14	22	72	1	0	Did not Vote
15	41	133	2	1	Voted
16	47	82	3	1	Voted
17	28	37	1	0	Did not Vote
18	43	50	2	0	Did not Vote
19	37	135	4	1	Voted
20	20	58	0	1	Did not Vote
21	62	60	1	0	Voted
22	48	29	2	0	Did not Vote
...	...	...	...	...	...
623,151	22	41	0	1	Did not Vote

Binary Classification

Data: $(x_i, y_i) \in \mathbb{R}^p \times \{-1, 1\}, \ i = 1, \ldots, n$

- $x = (x_1, \ldots, x_p) \in \mathbb{R}^p$ is the vector of features (indep. variables)
- $y \in \{-1, 1\}$ is the set of possible responses/labels

Task: predict y based on the linear function $\beta^T x$

- $\beta \in \mathbb{R}^p$ are the model coefficients

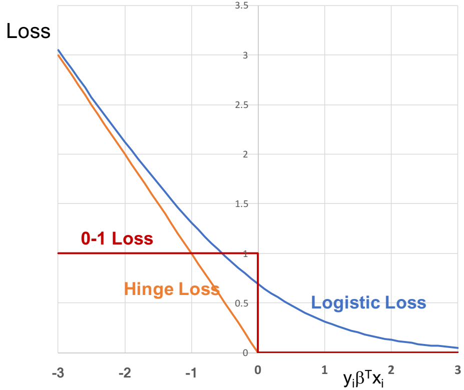
Loss function: $\ell(y, \beta^T x)$ represents the loss incurred when the truth is y but our classification/prediction is based on $\beta^T x$

Empirical Loss Minimization Problem: $\min_{\beta} \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \beta^T x_i)$

Loss Functions for Binary Classification

Some common loss functions used for binary classification

- 0-1 loss: $\ell(y, \beta^T x) := \mathbf{1}(y\beta^T x < 0)$
- Hinge loss: $\ell(y, \beta^T x) := \max(0, 1 - y\beta^T x)$
- Logistic loss: $\ell(y, \beta^T x) := \ln(1 + \exp(-y\beta^T x))$



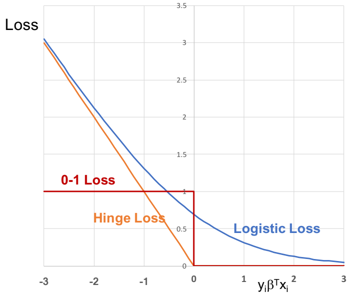
Advantages of Logistic Loss Function

Why use the logistic loss function for classification?

- Computational advantages: convex, smooth
- Fits previous statistical model of conditional probablity:

$$P(Y = y \mid x) = \frac{1}{1+\exp(-y\beta^T x)}$$

- Makes sense when the data is non-separable
- Robust to misspecification of class labels



Logistic Regression Problem of Interest, continued

Alternate versions of optimization problem add regularization and/or sparsification:

$$\begin{array}{rrll} L_n^* & := & \min\limits_{\beta} & L_n(\beta) := \frac{1}{n}\sum_{i=1}^n \ln(1+\exp(-y_i\beta^T x_i)) + \lambda\|\beta\|_p \\ & & \text{s.t.} & \beta \in \mathbb{R}^p \\ & & & \|\beta\|_0 \le k \end{array}$$

Overall aspirations:

- Good predictive performance on new (out of sample) observations
- Models that are more interpretable (e.g., sparse)

Computational Experiment: Greedy Coordinate Descent (GCD)

Consider Greedy Coordinate Descent (GCD) for Logistic Regression

Greedy Coordinate Descent for Logistic Regression

Greedy Coordinate Descent for Logistic Regression

Initialize at $\beta^0 \leftarrow 0$, $k \leftarrow 0$

At iteration $k \geq 0$:

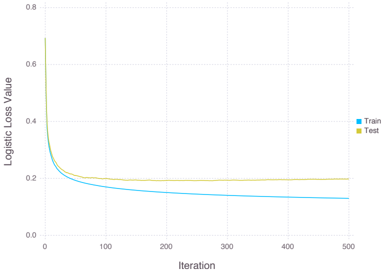
1. Compute $\nabla L_n(\beta^k)$
2. Compute $j_k \in \arg \max_{j \in \{1,\ldots,p\}} |\nabla L_n(\beta^k)_j|$
3. Set $\beta^{k+1} \leftarrow \beta^k - \alpha_k \mathrm{sgn}(\nabla L_n(\beta^k)_{j_k}) e_{j_k}$

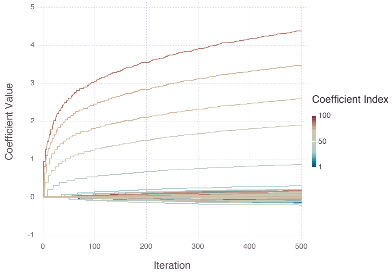
Why use Greedy Coordinate Descent for Logistic Regression?

- Scalable and effective when $n, p \gg 0$ and maybe $p > n$
- GCD performs variable selection
- GCD imparts implicit regularization
- Just one tuning parameter (number of iterations)

Implicit Regularization and Variable Selection Properties

Artificial example: $n = 1000, p = 100$, true model has 5 non-zeros





Compare with explicit regularization schemes (ℓ_1, ℓ_2, etc.)

How do GCD and SGD Inform Logistic Regression?

Some questions:

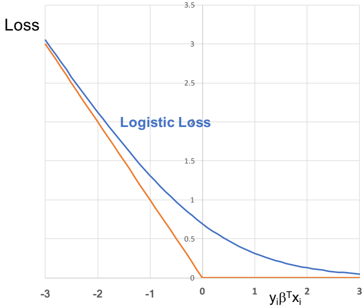
- How do the computational guarantees for Greedy Coordinate Descent and Stochastic Gradient Descent specialize to the case of Logistic Regression?
- Can we say anything further about the convergence properties of these methods in the special case of Logistic Regression?
- What role does problem structure/conditioning play in these guarantees?

Elementary Properties of the Logistic Loss Function

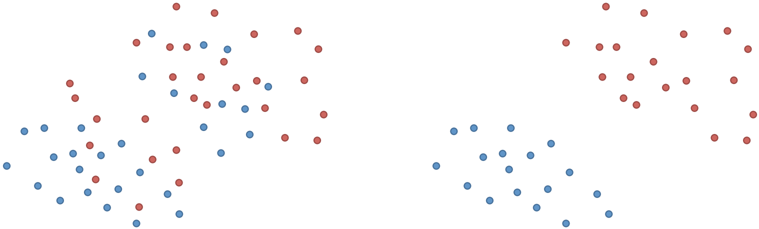
$$L_n^* \quad := \quad \min_{\beta} \quad L_n(\beta) := \tfrac{1}{n} \textstyle\sum_{i=1}^n \ln(1 + \exp(-y_i \beta^T x_i))$$

Recall that logistic regression "ideally" seeks β for which $y_i x_i^T \beta \gg 0$ for all i :

- $y_i = +1 \Rightarrow x_i^T \beta \gg 0$
- $y_i = -1 \Rightarrow x_i^T \beta \gg 0$



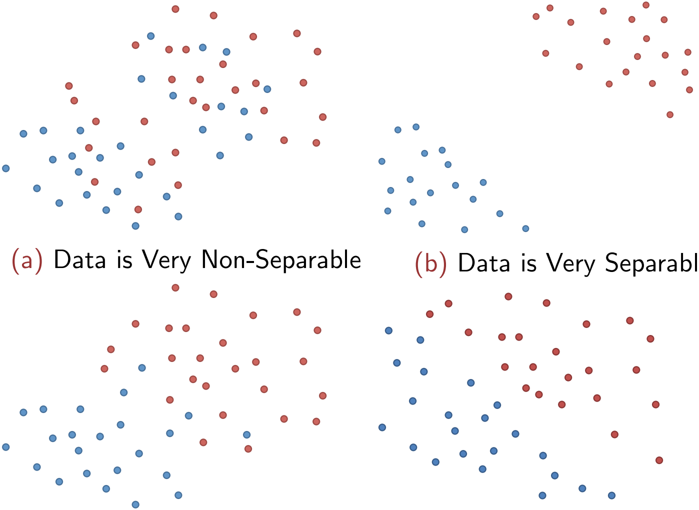
Geometry of the Data: Separable and Non-Separable Data



(a) Data is Non-Separable

(b) Data is Separable

Very/Mild Separable/Non-Separable Data



(a) Data is Very Non-Separable

(b) Data is Very Separable

(c) Data is Mildly Non-Separable

(d) Data is Mildly Separable

Separable and Non-Separable Data

Separable Data

The data is separable if there exists $\bar{\beta}$ for which

$$y_i \cdot (\bar{\beta})^T x_i > 0 \quad \text{for all } i = 1, \ldots, n$$

Non-Separable Data

The data is non-separable if it is not separable, namely, every β satisfies

$$y_i \cdot (\beta)^T x_i \leq 0 \quad \text{for at least one } i \in \{1, \ldots, n\}$$

Separable Data and Non-Attainment of Optimum

$$L_n^* \quad := \quad \min_{\beta} \quad L_n(\beta) := \tfrac{1}{n}\textstyle\sum_{i=1}^n \ln(1+\exp(-y_i\beta^T x_i))$$

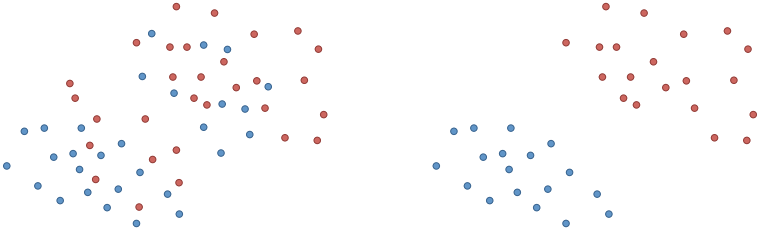
The data is separable if there exists $\bar{\beta}$ for which

$$y_i \cdot (\bar{\beta})^T x_i > 0 \quad \text{for all } i = 1, \ldots, n$$

If $\bar{\beta}$ separates the data, then $L_n(\theta\bar{\beta}) \to 0 \ (= L_n^*)$ as $\theta \to +\infty$

Perhaps trying to optimize the logistic loss function is unlikely to be effective at finding a “good” linear classifier ....

Separable and Non-Separable Data



(a) Data is Non-Separable

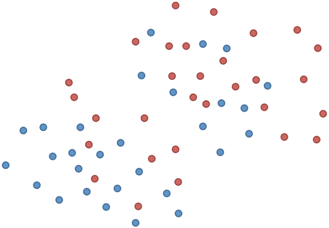
(b) Data is Separable

Results in the Non-Separable Case

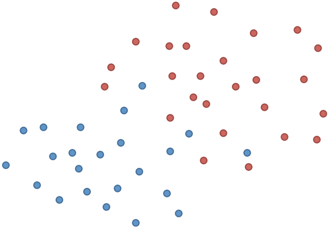
Results in the Non-Separable Case

Non-Separable Data and Problem Behavior/Conditioning

Let us quantify the degree of non-separability of the data.



(a) Very non-separable data



(b) Mildly non-separable data

We will relate this to problem behavior/conditioning....

Non-Separability Condition Number $\mathrm{DegNSEP}^*$

Definition of Non-Separability Condition Number $\mathrm{DegNSEP}^*$

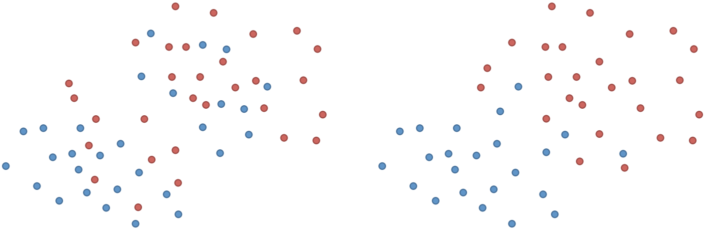
$$\begin{aligned} \mathrm{DegNSEP}^* := \quad \min_{\beta \in \mathbb{R}^p} \quad & \tfrac{1}{n}\textstyle\sum_{i=1}^n [y_i \beta^T x_i]^- \\ \text{s.t.} \quad & \|\beta\|_1 = 1 \end{aligned}$$

$\mathrm{DegNSEP}^*$ is the least average misclassification error (over all normalized classifiers)

$\mathrm{DegNSEP}^* > 0$ if and only if the data is strictly non-separable

Non-Separability Measure DegNSEP*

$$\mathrm{DegNSEP}^* := \min_{\beta\in\mathbb{R}^p} \quad \tfrac{1}{n}\textstyle\sum_{i=1}^n [y_i\beta^T x_i]^- \qquad \text{s.t.} \quad \|\beta\|_1 = 1$$



(a) DegNSEP* is large (b) DegNSEP* is small

DegNSEP* and Problem Behavior/Conditioning

$$L_n^* \quad := \quad \min_{\beta} \quad L_n(\beta) := \tfrac{1}{n}\sum_{i=1}^n \ln(1+\exp(-y_i\beta^T x_i))$$

$$\begin{array}{rcl} \mathrm{DegNSEP}^* := & \displaystyle\min_{\beta\in\mathbb{R}^p} & \tfrac{1}{n}\sum_{i=1}^n [y_i\beta^T x_i]^- \\ & \text{s.t.} & \|\beta\|_1 = 1 \end{array}$$

Theorem: Non-Separability and Sizes of Optimal Solutions

Suppose that the data is non-separable and $\mathrm{DegNSEP}^* > 0$. Then

1. the logistic regression problem LR attains its optimum,
2. for every optimal solution β^* of LR it holds that
   $\|\beta^*\|_1 \le \dfrac{L_n^*}{\mathrm{DegNSEP}^*} \le \dfrac{\ln(2)}{\mathrm{DegNSEP}^*}$, and
3. for any β it holds that $\|\beta\|_1 \le \dfrac{L_n(\beta)}{\mathrm{DegNSEP}^*}$.

Computational Guarantees for Greedy Coordinate Descent: Non-Separable Case

Theorem: Computational Guarantees for Greedy Coordinate Descent: Non-Separable Case

Consider the GCD applied to the Logistic Regression problem with step-sizes $\alpha_k := \frac{4n\|\nabla L_n(\beta^k)\|_\infty}{\|\mathbf{X}\|_{1,2}^2}$ for all $k \geq 0$, and suppose that the data is non-separable. Then for each $k \geq 0$ it holds that:

(i) (training error): $L_n(\beta^k) - L_n^* \ \leq \ \frac{2(\ln(2))^2\|\mathbf{X}\|_{1,2}^2}{k\cdot n\cdot(\mathrm{DegNSEP}^*)^2}$

(ii) (gradient norm): $\min\limits_{i\in\{0,\ldots,k\}} \|\nabla L_n(\beta^i)\|_\infty \ \leq \ \|\mathbf{X}\|_{1,2}\sqrt{\frac{(\ln(2)-L_n^*)}{2n\cdot(k+1)}}$

(iii) (regularization): $\|\beta^k\|_1 \ \leq \ \sqrt{k}\left(\frac{1}{\|\mathbf{X}\|_{1,2}}\right)\sqrt{8n(\ln(2) - L_n^*)}$

Computational Guarantees for Greedy Coordinate Descent: Non-Separable Case

Theorem: Computational Guarantees for Greedy Coordinate Descent: Non-Separable Case

Consider the GCD applied to the Logistic Regression problem with step-sizes $\alpha_k := \frac{4n\|\nabla L_n(\beta^k)\|_\infty}{\|\mathbf{X}\|_{1,2}^2}$ for all $k \geq 0$, and suppose that the data is non-separable. Then for each $k \geq 0$ it holds that:

(i) (training error): $L_n(\beta^k) - L_n^* \leq \frac{2(\ln(2))^2\|\mathbf{X}\|_{1,2}^2}{k \cdot n \cdot (\mathrm{DegNSEP}^*)^2}$

(ii) (gradient norm): $\min_{i \in \{0,\ldots,k\}} \|\nabla L_n(\beta^i)\|_\infty \leq \|\mathbf{X}\|_{1,2}\sqrt{\frac{(\ln(2)-L_n^*)}{2n\cdot(k+1)}}$

(iii) (regularization): $\|\beta^k\|_1 \leq \sqrt{k}\left(\frac{1}{\|\mathbf{X}\|_{1,2}}\right)\sqrt{8n(\ln(2) - L_n^*)}$

Computational Guarantees for Stochastic Gradient Descent: Non-Separable Case

Theorem: Computational Guarantees for Stochastic Gradient Descent: Non-Separable Case

Consider SGD applied to the Logistic Regression problem with step-sizes $\alpha_i := \frac{\sqrt{8n\ln(2)}}{\sqrt{k+1}\|\mathbf{X}\|_{2,2}\|\mathbf{X}\|_{2,\infty}}$ for $i = 0, \ldots, k$, and suppose that the data is non-separable. Then it holds that:

(i) (training error):

$$\mathbb{E}[\min_{0\le i\le k} L_n(\beta^i)] - L_n^* \ \le \ \frac{1}{\sqrt{k+1}}\left(\frac{(L_n^*)^2\|\mathbf{X}\|_{2,\infty}^2}{4\sqrt{2\ln(2)}(\text{DegNSEP}^*)^2} + \frac{\sqrt{2\ln(2)n}\|\mathbf{X}\|_{2,\infty}}{\|\mathbf{X}\|_{2,2}}\right)$$

(ii) (gradient norm):

$$\mathbb{E}\left[\min_{i\in\{0,\ldots,k\}} \|\nabla L_n(\beta^i)\|_2^2\right] \ \le \ \frac{1}{\sqrt{k+1}}\left(\frac{\sqrt{\ln(2)}\|\mathbf{X}\|_{2,2}\|\mathbf{X}\|_{2,\infty}}{\sqrt{2n}}\right)$$

(iii) (regularization): $\|\beta^k\|_2 \ \le \ \sqrt{k+1}\left(\frac{\sqrt{8n\ln(2)}}{\|\mathbf{X}\|_{2,2}}\right)$

Computational Guarantees for Stochastic Gradient Descent: Non-Separable Case

Theorem: Computational Guarantees for Stochastic Gradient Descent: Non-Separable Case

Consider SGD applied to the Logistic Regression problem with step-sizes $\alpha_i := \frac{\sqrt{8n\ln(2)}}{\sqrt{k+1}\|\mathbf{X}\|_{2,2}\|\mathbf{X}\|_{2,\infty}}$ for $i = 0, \ldots, k$, and suppose that the data is non-separable. Then it holds that:

(i) (training error):

$$\mathbb{E}[\min_{0\le i\le k} L_n(\beta^i)] - L_n^* \;\le\; \frac{1}{\sqrt{k+1}}\left(\frac{(L_n^*)^2\|\mathbf{X}\|_{2,\infty}^2}{4\sqrt{2\ln(2)}(\mathrm{DegNSEP}*)^2} + \frac{\sqrt{2\ln(2)n}\|\mathbf{X}\|_{2,\infty}}{\|\mathbf{X}\|_{2,2}}\right)$$

(ii) (gradient norm):

$$\mathbb{E}\left[\min_{i\in\{0,\ldots,k\}} \|\nabla L_n(\beta^i)\|_2^2\right] \;\le\; \frac{1}{\sqrt{k+1}}\left(\frac{\sqrt{\ln(2)}\|\mathbf{X}\|_{2,2}\|\mathbf{X}\|_{2,\infty}}{\sqrt{2n}}\right)$$

(iii) (regularization): $\|\beta^k\|_2 \;\le\; \sqrt{k+1}\left(\frac{\sqrt{8n\ln(2)}}{\|\mathbf{X}\|_{2,2}}\right)$

Reaching Linear Convergence

Reaching Linear Convergence using Greedy Coordinate Descent for Logistic Regression

For logistic regression, does GCD exhibit linear convergence?

Some Definitions/Notation

Definitions:

- $R := \max_{i \in \{1,\ldots,n\}} \|x_i\|_2$ (maximum ℓ_2 norm of the feature vectors)
- $H(\beta^*)$ denotes the Hessian of $L_n(\cdot)$ at an optimal solution β^*
- $\lambda_{\mathrm{pmin}}(H(\beta^*))$ denotes the smallest non-zero (and hence positive) eigenvalue of $H(\beta^*)$

Reaching Linear Convergence of GCD for Logistic Regression

Theorem: Reaching Linear Convergence of GCD for Logistic Regression

Consider GCD applied to the Logistic Regression problem with step-sizes $\alpha_k := \frac{4n\|\nabla L_n(\beta^k)\|_\infty}{\|\mathbf{X}\|_{1,2}^2}$ for all $k \geq 0$, and suppose that the data is non-separable. Define:

$$\check{k} := \frac{16p\ln(2)^2\|\mathbf{X}\|_{1,2}^4 R^2}{9n^2(\mathrm{DegNSEP}^*)^2\lambda_{\mathrm{pmin}}(H(\beta^*))^2} \ .$$

Then for all $k \geq \check{k}$, it holds that:

$$L_n(\beta^k) - L_n^* \leq (L_n(\beta^{\check{k}}) - L_n^*)\left(1 - \frac{\lambda_{\mathrm{pmin}}(H(\beta^*))n}{p \cdot \|\mathbf{X}\|_{1,2}^2}\right)^{k-\check{k}} \ .$$

Reaching Linear Convergence of GCD for Logistic Regression, cont.

Some comments:

- Proof relies on (a slight generalization of) the "generalized self-concordance" property of the logistic loss function due to [Bach 2014]
- Furthermore, we can bound:

$$\lambda_{\mathrm{pmin}}(H(\beta^*)) \;\geq\; \tfrac{1}{4n}\lambda_{\mathrm{pmin}}(\mathbf{X}^T\mathbf{X}) \exp\left(-\frac{\ln(2)\|\mathbf{X}\|_{1,\infty}}{\mathrm{DegNSEP}^*}\right)$$

- As compared to results of a similar flavor for other algorithms, here we have an exact characterization of when the linear convergence "kicks in" and also what the rate of linear convergence is guaranteed to be
- Q: Can we exploit this generalized self-concordance property in other ways? (still ongoing . . .)

DegNSEP* and "Perturbation to Separability"

$$\mathrm{DegNSEP}^* := \min_{\beta\in\mathbb{R}^p} \quad \frac{1}{n}\sum_{i=1}^{n}[y_i\beta^T x_i]^-$$

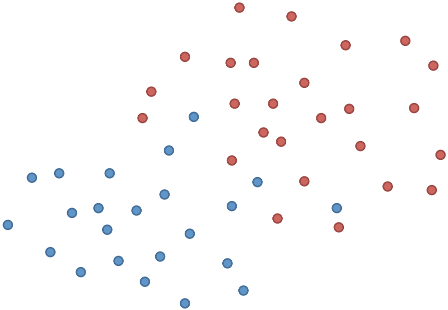
$$\text{s.t.} \quad \|\beta\|_1 = 1$$

Theorem: DegNSEP* is the "Perturbation to Separability"

$$\mathrm{DegNSEP}^* = \inf_{\Delta x_1,\ldots,\Delta x_n} \quad \frac{1}{n}\sum_{i=1}^{n}\|\Delta x_i\|_\infty$$

$$\text{s.t.} \quad (x_i+\Delta x_i, y_i), i=1,\ldots,n \text{ are separable}$$

Illustration of Perturbation to Separability

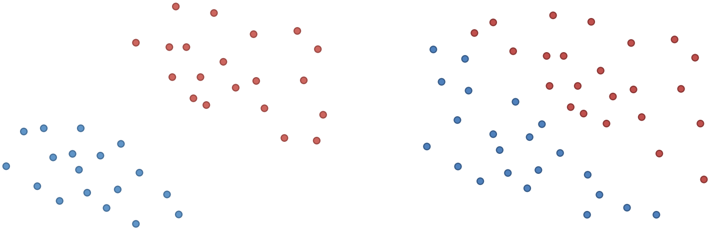


Results in the Separable Case

Results in the Separable Case

Separable Data and Problem Behavior/Conditioning

Let us quantify the degree of separability of the data.



(a) Very separable data

(b) Barely separable data

We will relate this to problem behavior/conditioning....

Separability Condition Number DegSEP^*

Definition of Separability Condition Number DegSEP^*

$$\begin{aligned} \mathrm{DegSEP}^* := \quad & \max_{\beta \in \mathbb{R}^p} \quad \min_{i \in \{1,\ldots,n\}} [y_i \beta^T x_i] \\ & \text{s.t.} \quad \|\beta\|_1 \le 1 \end{aligned}$$

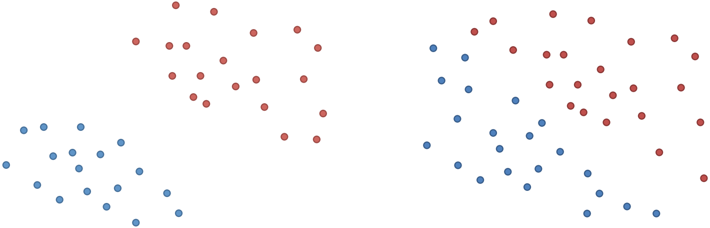
DegSEP^* maximizes the minimal classification value $[y_i \beta^T x_i]$ (over all normalized classifiers)

DegSEP^* is simply the “maximum margin” in machine learning parlance

$\mathrm{DegSEP}^* > 0$ if and only if the data is separable

Separability Measure DegSEP^*

$$
\begin{aligned}
\mathrm{DegSEP}^* := \quad & \max_{\beta \in \mathbb{R}^p} \quad \min_{i \in \{1,\ldots,n\}} [y_i \beta^T x_i] \\
& \text{s.t.} \quad \|\beta\|_1 \leq 1
\end{aligned}
$$



(a) DegSEP^* is large

(b) DegSEP^* is small

DegSEP* and Problem Behavior/Conditioning

$$L_n^* \quad := \quad \min_{\beta} \quad L_n(\beta) := \tfrac{1}{n}\textstyle\sum_{i=1}^n \ln(1+\exp(-y_i\beta^T x_i))$$

$$\begin{aligned} \mathrm{DegSEP}^* := \quad & \max_{\beta\in\mathbb{R}^p} \quad \min_{i\in\{1,\ldots,n\}} [y_i\beta^T x_i] \\ & \text{s.t.} \quad \|\beta\|_1 \le 1 \end{aligned}$$

Theorem: Separability and Non-Attainment

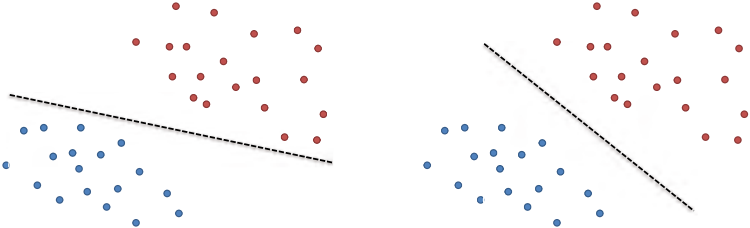
Suppose that the data is separable. Then $\mathrm{DegSEP}^* > 0$, $L_n^* = 0$, and LR does not attain its optimum.

Despite this, it turns out that the Steepest Descent family and also Stochastic Gradient Descent are reasonably effective at finding an approximate margin maximizer as we shall shortly see....

Margin function $\rho(\beta)$

Margin function $\rho(\beta)$

$$\rho(\beta) := \min_{i \in \{1,\ldots,n\}} [y_i \beta^T x_i]$$



(a) $\rho(\beta)$ is small

(b) $\rho(\beta)$ is large

Computational Guarantees for Greedy Coordinate Descent: Separable Case

Theorem: Computational Guarantees for Greedy Coordinate Descent: Separable Case

Consider GCD applied to the Logistic Regression problem with step-sizes $\alpha_k := \frac{4n\|\nabla L_n(\beta^k)\|_\infty}{\|\mathbf{X}\|_{1,2}^2}$ for all $k \geq 0$, and suppose that the data is separable.

(i) (margin bound): there exists $i \leq \left\lfloor \frac{3.7n\|\mathbf{X}\|_{1,2}^2}{(\mathrm{DegSEP}^*)^2} \right\rfloor$ for which the normalized iterate $\bar{\beta}^i := \beta^i/\|\beta^i\|_1$ satisfies

$$\rho(\bar{\beta}^i) \;\geq\; \frac{.18 \cdot \mathrm{DegSEP}^*}{n} \; .$$

(ii) (shrinkage): $\|\beta^k\|_1 \;\leq\; \sqrt{k}\left(\frac{1}{\|\mathbf{X}\|_{1,2}}\right)\sqrt{8n\ln(2)}$

(iii) (gradient norm): $\min\limits_{i\in\{0,\ldots,k\}} \|\nabla L_n(\beta^i)\|_\infty \;\leq\; \|\mathbf{X}\|_{1,2}\sqrt{\frac{\ln(2)}{2n\cdot(k+1)}}$

Computational Guarantees for Greedy Coordinate Descent: Separable Case

Theorem: Computational Guarantees for Greedy Coordinate Descent: Separable Case

Consider GCD applied to the Logistic Regression problem with step-sizes $\alpha_k := \frac{4n\|\nabla L_n(\beta^k)\|_\infty}{\|\mathbf{X}\|_{1,2}^2}$ for all $k \geq 0$, and suppose that the data is separable.

(i) (margin bound): there exists $i \leq \left\lfloor \frac{3.7n\|\mathbf{X}\|_{1,2}^2}{(\mathrm{DegSEP}^*)^2} \right\rfloor$ for which the normalized iterate $\bar{\beta}^i := \beta^i/\|\beta^i\|_1$ satisfies

$$\rho(\bar{\beta}^i) \ \geq \ \frac{.18 \cdot \mathrm{DegSEP}^*}{n} \ .$$

(ii) (shrinkage): $\|\beta^k\|_1 \ \leq \ \sqrt{k}\left(\frac{1}{\|\mathbf{X}\|_{1,2}}\right)\sqrt{8n\ln(2)}$

(iii) (gradient norm): $\min\limits_{i\in\{0,\ldots,k\}} \|\nabla L_n(\beta^i)\|_\infty \ \leq \ \|\mathbf{X}\|_{1,2}\sqrt{\frac{\ln(2)}{2n\cdot(k+1)}}$

Computational Guarantees for Stochastic Gradient Descent: Separable Case

Theorem: Computational Guarantees for Stochastic Gradient Descent: Separable Case

Consider SGD applied to the Logistic Regression problem with step-sizes $\alpha_i := \frac{\sqrt{8n\ln(2)}}{\sqrt{k+1}\|\mathbf{X}\|_{2,2}\|\mathbf{X}\|_{2,\infty}}$ for $i = 0, \ldots, k$, where

$$k := \left\lfloor \frac{28.1n^3\|\mathbf{X}\|_{2,2}^2\|\mathbf{X}\|_{2,\infty}^2}{\gamma^2(\mathrm{DegSEP}^*)^4} \right\rfloor$$

and $\gamma \in (0,1]$. If the data is separable, then :

$$\mathbb{P}\left(\exists i \in \{0,\ldots,k\} \text{ s.t. } \rho(\bar{\beta}^i) \geq \frac{\gamma(\mathrm{DegSEP}^*)^2}{20n^2\|\mathbf{X}\|_{2,\infty}}\right) \geq 1-\gamma \ .$$

where $\bar{\beta}^i := \beta^i/\|\beta^i\|_1$ are the normalized iterates of SGD.

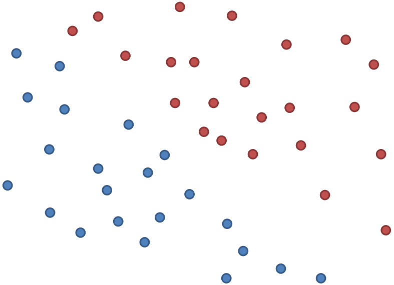
DegSEP* and "Perturbation to Non-Separability"

$$\begin{array}{rll} \mathrm{DegSEP}^* := & \max_{\beta \in \mathbb{R}^p} & \min_{i \in \{1,\ldots,n\}} [y_i \beta^T x_i] \\ & \text{s.t.} & \|\beta\|_1 \leq 1 \end{array}$$

Theorem: DegSEP* is the "Perturbation to Non-Separability"

$$\begin{array}{rll} \mathrm{DegSEP}^* = & \inf_{\Delta x_1,\ldots,\Delta x_n} & \max_{i \in \{1,\ldots,n\}} \|\Delta x_i\|_\infty \\ & \text{s.t.} & (x_i + \Delta x_i, y_i), i = 1, \ldots, n \text{ are non-separable} \end{array}$$

Illustration of Perturbation to Non-Separability



Other Issues

Some other topics not mentioned (still ongoing):

- Other first-order methods for logistic regression (gradient descent, accelerated gradient descent, other randomized methods, etc.
- High-dimensional regime $p > n$, define $\mathrm{DegNSEP}^*_k$ and DegSEP^*_k for restricting β to satisfy $\|\beta\|_0 \leq k$
- Numerical experiments comparing methods
- Other...

Summary

- Some old and new results for Greedy Coordinate Descent and Stochastic Gradient Descent
- Analyzing these methods for Logistic Regression: separable/non-separable cases
- Non-Separable case
  - condition number $\mathrm{DegNSEP}^*$
  - computational guarantees for Greedy Coordinate Descent and Stochastic Gradient Descent, including reaching linear convergence
- Separable case
  - condition number DegSEP^*
  - computational guarantees for Greedy Coordinate Descent and Stochastic Gradient Descent, including computing an approximate maximum margin classifier

Results for Some other Methods

Results for Some other Methods

Standard Accelerated Gradient Method (AGM)

$$P: \quad F^* := \quad \text{minimum}_x \quad F(x)$$
$$\text{s.t.} \quad x \in \mathbb{R}^p$$

Lipschitz gradient: $\|\nabla f(y) - \nabla f(x)\|_2 \leq L\|y - x\|_2$ for all $x, y \in \mathbb{R}^p$

Accelerated Gradient Method (AGM)

Given $x^0 \in \mathbb{R}^p$ and $z^0 := x^0$, and $i \leftarrow 0$. Define step-size parameters $\theta_i \in (0, 1]$ recursively by $\theta_0 := 1$ and θ_{i+1} satisfies $\frac{1}{\theta_{i+1}^2} - \frac{1}{\theta_{i+1}} = \frac{1}{\theta_i^2}$.

At iteration k:

1. Update : $y^k \leftarrow (1 - \theta_k)x^k + \theta_k z^k$

   $x^{k+1} \leftarrow y^k - \frac{1}{L}\nabla f(y^k)$

   $z^{k+1} \leftarrow z^k + \frac{1}{\theta_k}(x^{k+1} - y^k)$

Computational Guarantees for Accelerated Gradient Method (AGM) for Logistic Regression

Theorem: Computational Guarantees for Accelerated Gradient Method (AGM) for Logistic Regression

Consider the AGM applied to the Logistic Regression problem initiated at $\beta^0 := 0$, and suppose that the data is non-separable. Then for each $k \geq 0$ it holds that:

$$\text{(training error):} \qquad L_n(\beta^k) - L_n^* \ \leq \ \frac{2(\ln(2))^2 \|\mathbf{X}\|_{2,2}^2}{n \cdot (k+1)^2 \cdot (\mathrm{DegNSEP}^*)^2}$$

AGM with Simple Re-Starting (AGM-SRS)

Assume that $0 < F^* := \text{minimum}_x \; F(x)$

Accelerated Gradient Method with Simple Re-Starting (AGM-SRS)

Initialize with $x^0 \in \mathbb{R}^p$.
Set $x_{1,0} \leftarrow x^0$, $i \leftarrow 1$.

At outer iteration i:

1. **Initialize inner iteration.** $j \leftarrow 0$
2. **Run inner iterations.** At inner iteration j:

```
If $\frac{F(x_{i,j})}{F(x_{i,0})} \geq 0.8$ , then:
        $x_{i,j+1} \leftarrow \text{AGM}(F(\cdot),\; x_{i,0},\; j+1)$ ,
        $j \leftarrow j+1$, and Goto step 2.
Else $x_{i+1,0} \leftarrow x_{i,j}$, $i \leftarrow i+1$, and Goto step 1.
```

"$x_{i,j} \leftarrow \text{AGM}(F(\cdot),\; x_{i,0},\; j)$" denotes assigning to $x_{i,j}$ the j^{th} iterate of AGM applied with objective function $F(\cdot)$ using the initial point $x_{i,0} \in \mathbb{R}^p$

Computational Guarantee for AGM with Simple Re-Starting for Logistic Regression

Computational Guarantee for Accelerated Gradient Method with Simple Re-Starting for Logistic Regression

Consider the AGM with Simple Re-Starting applied to the Logistic Regression problem initiated at $\beta^0 := 0$, and suppose that the data is non-separable. Within a total number of computed iterates k that does not exceed

$$\frac{5.8\|\mathbf{X}\|_{2,2}}{\sqrt{n}\cdot \mathrm{DegNSEP}^*} + \frac{8.4\|\mathbf{X}\|_{2,2}\cdot L_n^*}{\sqrt{n}\cdot \mathrm{DegNSEP}^*\cdot\sqrt{\varepsilon}} ,$$

the algorithm will deliver an iterate β^k for which

$$L_n(\beta^k) - L_n^* \leq \varepsilon .$$

Back-up Slides: Related Results for AdaBoost

Back-up Slides: Related Results for AdaBoost

AdaBoost: First Problem of Interest

AdaBoost is also Greedy Coordinate Descent, but replaces the logistic loss function with the log-exponential loss:

$$L_l^* := \quad \min_{\lambda \geq 0} \qquad L_l(\lambda) \quad = \quad \ln\left(\tfrac{1}{m}\textstyle\sum_{i=1}^m \exp\left(-(A\lambda)_i\right)\right) \ .$$

- Data: $(x_1, y_1) \ldots, (x_m, y_m)$ where $x_i \in \mathbb{R}^n$ is the i^{th} feature vector and $y_i \in \{-1, +1\}$
- Here $A := Y\mathbf{X}$, i.e., $A_{ij} := y_i(x_i)_j$

Note that λ^* is a linear separator of the data if and only if $A\lambda^* > 0$

Assume for convenience that for every column A_j, $-A_j$ is also a column of A

AdaBoost: Second Problem of Interest

$\Delta_n := \{x \in \mathbb{R}^n : e^T x = 1, x \geq 0\}$ is the standard simplex in $\mathbb{R}^n$

Recall that λ^* is a linear separator of the data if and only if $A\lambda^* > 0$

The margin of a classifier $\lambda \in \mathbb{R}^n$ is:

$$p(\lambda) := \min_{i \in \{1,\ldots,m\}} (A\lambda)_i = \min_{w \in \Delta_m} \; w^T A\lambda$$

It makes sense to look for a classifier with large margin, i.e., to solve:

$$\mathrm{M} \;\; : \quad \rho^* := \max_{\lambda \in \Delta_n} \; p(\lambda) \; .$$

Dual of the Maximum Margin Problem

The "edge" of a vector of weights on the data, $w \in \Delta_m$, is:

$$f(w) := \max_{j \in \{1,\ldots,n\}} \; w^T A_j = \max_{\lambda \in \Delta_n} \; w^T A\lambda$$

The (linear programming) dual of the maximum margin problem is the problem of minimizing the edge:

$$\mathrm{E} \;:\quad f^* := \min_{w \in \Delta_m} \; f(w) \ ,$$

AdaBoost is three algorithms:

- A boosting method based on a scheme for (multiplicatively) updating a vector of weights on the data
- Greedy Coordinate Descent applied to minimize the log-exponential loss function
- A version of the Mirror Descent method applied to the above problem E

Computational Guarantees for AdaBoost

Theory for Greedy Coordinate Descent and Mirror Descent leads to computational guarantees for AdaBoost:

Step-Size Strategy		Separable Data	Non-Separable Data	
		Margin Bound $\rho^* - p(\lambda^{k+1})$	Gradient Bound $\min\limits_{i \in \{0,\ldots,k\}} \Vert\nabla L_l(\hat{\lambda}^i)\Vert_\infty$	Loss Bound $L_l(\hat{\lambda}^k) - L_l^*$
“edge rule:”	$\alpha_k = \Vert\nabla L_l(\hat{\lambda}^k)\Vert_\infty$		$\sqrt{\frac{2\ln(m)}{k+1}}$	$\frac{8\ln(m)^2}{(\mathrm{NSEP}_l^*)^2 k}$
“line-search:”	$\alpha_k = \frac{1}{2}\ln\left(\frac{1+r_k}{1-r_k}\right)$		$\sqrt{\frac{2\ln(m)}{k+1}}$	$\frac{8\ln(m)^2}{(\mathrm{NSEP}_l^*)^2 k}$
“constant:”	$\alpha_i := \sqrt{\frac{2\ln(m)}{k+1}}$ for $i = 0, \ldots, k$	$\sqrt{\frac{2\ln(m)}{k+1}}$	$\sqrt{\frac{2\ln(m)}{k+1}}$	
“adaptive:”	$\alpha_k = \sqrt{\frac{2\ln(m)}{k+1}}$	$\frac{\sqrt{\frac{\ln(m)}{2}}[2+\ln(k+1)]}{2(\sqrt{k+2}-1)}$	$\frac{\sqrt{\frac{\ln(m)}{2}}[2+\ln(k+1)]}{2(\sqrt{k+2}-1)}$	

NSEP_l^* is a “non-separability condition number” for log-exponential loss