Definitions and notations

Zhehao Wang

November 11, 2018

1 Calculus and linear algebra

1.1 Limit

Let $f(x)$ be a function defined on an interval that contains $x = a$, except possibly at $x = a$, then we say that

$$\lim_{x \to a} f(x) = L$$

if for every $\epsilon > 0$ there is some number $\delta > 0$ such that

$$|f(x) - L| < \epsilon \text{ whenever } 0 < |x - a| < \delta$$

1.2 Gradient

Given $f(\vec{x})$ where $\vec{x} = (x_1, \ldots, x_n)$ on $\mathbb{R}^n$

$$\bigtriangledown f(a_1, \ldots, a_n) = (\frac{\partial f}{\partial x_1}(a_1, \ldots, a_n), \ldots, \frac{\partial f}{\partial x_n}(a_1, \ldots, a_n))$$

Intuition: gradient is a vector (the rate of change of your function, when you move in a certain direction), which in a two-dimensional space, tangents the curve at a given point.

1.3 Directional derivative

Homework notation

$$f'(x; u) = \lim_{h \to 0} \frac{f(x + hu) - f(x)}{h}$$

Wikipedias notation

$$\bigtriangledown_v f(\vec{x}) = \lim_{h \to 0} \frac{f(\vec{x} + h\vec{v}) - f(\vec{x})}{h}$$

Intuition: the rate-of-change of a function $f(\vec{x})$ on direction $\vec{v}$. Remember that this is a scalar (since we are given the direction).

$$f'(x; u) = \bigtriangledown f(x)^T u$$

Meaning gradient on a certain direction vector u is $f(x)$'s directional derivative on u.

1.4   Vector norm

A norm is a function that assigns a stricly positive length or size to each vector in a vector space (except the zero vector which is assigned a length of 0).
Absolute value norm is a norm on the one-dimensional vector spaces formed by real or complex numbers.
$$\|x\| = |x|$$

Euclidean norm on a Euclidean space $\mathbb{R}^n$ is such

$$\|\vec{x}\|_2 = \sqrt{x_1^2 + \cdots + x_n^2}$$

Manhattan or taxicab norm

$$\|\vec{x}\|_1 = \sum_{i=1}^{n} |x_i|$$

p**-norm**

$$\|\vec{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}$$

Note that when $p = 1$, we get Manhattan norm, and when $p = 2$, we get Euclidean norm.
When $p = \infty$

$$\|\vec{x}\|_\infty = \max_i |x_i|$$

1.5   argmax

Points of the domain of some function at which the function values are maximized.
Given an arbitrary set X, a totally ordered set Y and a function $f : X \to Y$, the $\arg\max$ over some subset S of X is defined by

$$\arg\max_{x \in S \subseteq X} f(x) = \{x \mid x \in S \land \forall y \in S : f(y) \le f(x)\}$$

1.6   Dot product, inner product

Maps two equal-size vectors to a real value.

$$\cdot : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$$

Defined geometrically, given two vectors a and b,

$$a \cdot b = \|a\| \, \|b\| \, cos(\theta)$$

where θ is the angle between a and b.
Defined algebraically,

$$a \cdot b = ab^T$$

1.7   Cauchy-Schwarz inequality

Given two vectors u and v

$$|\langle u, v \rangle|^2 \leq |\langle u, u \rangle| \cdot |\langle v, v \rangle|$$

Where $|\langle u, u \rangle|$ is the inner product. The equality holds only when u and v are linearly independent (parallel).

1.8   Jacobian matrix

Jacobian matrix is the matrix of all first-order partial derivatives of a vector-valued function.
Suppose $\vec{f} : \mathbb{R}^n \to \mathbb{R}^m$, the Jacobian matrix $\vec{J}$ of $\vec{f}$ is defined as follows

$$\vec{J} = \begin{bmatrix} \frac{\partial \vec{f}}{\partial x_1} & \cdots & \frac{\partial \vec{f}}{\partial x_n} \end{bmatrix}$$

or component-wise $\vec{J}_{ij} = \frac{\partial \vec{f_i}}{\partial x_j}$, meaning

$$\vec{J} = \begin{bmatrix} \frac{\partial \vec{f_1}}{\partial x_1} & \cdots & \frac{\partial \vec{f_1}}{\partial x_n} \\ & \cdots & \\ \frac{\partial \vec{f_m}}{\partial x_1} & \cdots & \frac{\partial \vec{f_m}}{\partial x_n} \end{bmatrix}$$

1.9   Order of a matrix

1.10   Inversibility of a matrix

1.11   Eigenvalue

2   Probability

2.1   Random variable

A random variable $X : \Omega \to E$ is a measurable function from a set of possible outcomes Ω to a measurable space E. Often times $E = \mathbb{R}$
The probability that X takes on a value in a measurable set $S \subseteq E$ is written as

$$Pr(X \in S) = P(\omega \in \Omega | X(\omega) \in S)$$

Intuition: mapping outcomes of a random process to numbers, like this definition of X

$$X = \begin{cases} 0, & \text{if heads} \\ 1, & \text{if tails} \end{cases}$$

Instead of a traditional algebraic variable that can be solved for one value, a random variable can have different values (each with a probability) under different conditions.

2.2 Law of large numbers

$X_i, X_2, \ldots$ is an infinite sequence of independent and identically distributed (iid) random variables with expected value $E(X_1) = E(X_2) = \cdots = \mu$, and

$$\bar{X}_n = \frac{1}{n}(X_1 + \cdots + X_n) \, ^*$$

The weak law states that for any positive number ϵ

$$\lim_{n \to \infty} Pr(|\bar{X}_n - \mu| > \epsilon) = 0$$

The strong law states that

$$Pr(\lim_{n \to \infty} \bar{X}_n = \mu) = 1$$

Intuition: law of large numbers is a theorem that describes the average of the results obtained from a large number of trials should be close to the expected value, and will tend to become closer as more trials are performed.

3 Statistical learning

Let $\mathbf{X}$ denote the input space, $\mathbf{Y}$ denote the output space, and $\mathbf{A}$ denote the action space.
A **decision function / prediction function** $f : \mathbf{X} \to \mathbf{A}$ maps an input to an action.
A **loss function** $l : \mathbf{A} \times \mathbf{Y} \to \mathbb{R}$ evaluates an action in the context of an output, and maps the error to a real. In traditional problems we can usually assume action has no effect on the output (like stock market prediction).

3.1 Risk, Bayesian function, empirical risk

Assume there is a data generating distribution $P_{\mathbf{X} \times \mathbf{Y}}$.
Risk of a decision function can then be defined as

$$R(f) = \mathbb{E}l(f(x), y)$$

---

*What does the sum of random variables mean? Seems that each X_i here means 'sampled' values following the distribution defined by the random variable, like, the outcome of an experiment

Where an input/output pair (x, y) is generated independently and identically distributed from $P_{\mathbf{X} \times \mathbf{Y}}$.

A **Bayesian decision function / target function** $f^* : \mathbf{X} \to \mathbf{A}$ is a function that achieves the minimal risk among all possible functions.

$$f^* = \arg\min_f R(f)$$

Risk cannot be calculated as we are not given $P_{\mathbf{X} \times \mathbf{Y}}$.

Let $\mathbf{D}_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ be drawn iid from $P_{\mathbf{X} \times \mathbf{Y}}$.

The **empirical risk** of $f : \mathbf{X} \to \mathbf{A}$ with respect to $\mathbf{D}_n$ is

$$\hat{R}_n(f) = \frac{1}{n} \sum_{i=1}^{n} l(f(x_i), y_i)$$

By strong law of large numbers (2.2), when n is big enough, empirical risk can be used to approximate risk.

3.2   Empirical risk minimization, constrained empirical risk minimization

A function $\hat{f}$ is an **empirical risk minimizer** if

$$\hat{f} = \arg\min_f \hat{R}_n(f)$$

The prediction function that produces the smallest empiricall risk over set $\mathbf{D}_n$. This naturally leads to overfit.

So we minimize empirical risk, but only within a hypothesis space $\mathbf{F}$, being a set of prediction functions.

Constrained empirical risk minimizer can then be defined as

$$\hat{f}_n = \arg\min_{f \in \mathbf{F}} \frac{1}{n} \sum_{i=1}^{n} l(f(x_i), y_i)$$

If we plug Risk in instead, a **constrained risk minimizer** becomes

$$f_{\mathbf{F}}^* = \arg\min_{f \in \mathbf{F}} \mathbb{E} l(f(x), y)$$

Approximation error is the risk difference between the constrained risk minimizer (in $\mathbf{F}$) and the target

$$R(f_{\mathbf{F}}) - R(f^*)$$

Estimation error is the risk difference between the constrained empirical risk minimizer and the constrained risk minimizer (both in $\mathbf{F}$)

$$R(\hat{f}_n) - R(f_{\mathbf{F}})$$

Optimization error is the risk difference between a function $\tilde{f}_n$ (which we find in practice) and the constrained empirical risk minimizer [†]

$$R(\tilde{f}_n) - R(\hat{f}_n)$$

Excess risk is the risk between a function and the target

$$\text{Excess Risk}(\hat{f}_n) = \text{Estimation Error} + \text{Approximation Error} = R(\hat{f}_n) - R(f^*)$$

$$\text{Excess Risk}(\tilde{f}_n) = \text{Optimization Error} + \text{Estimation Error} + \text{Approximation Error}$$
$$= R(\tilde{f}_n) - R(f^*)$$

3.3 Complexity measure, l_1, l_2 regularization, ridge regression, lasso regression, coordinate gradient descent

Regularization is a mechanism to reduce overfitting. By narrowing down the hypothesis space to only those functions satisfying being within a complexity measure.
Complexity measure $\Omega : \mathbf{F} \to [0, \infty)$ for linear decision functions $f(x) = w^T x$ can be defined using the p-norm (to the p) of w.

- l_0 complexity is then the number of non-zero coefficients

- l_1 **lasso** complexity is $\sum_{i=1}^{d} |w_i|$

- l_2 **ridge** complexity is $w^T w = \sum_{i=1}^{d} w_i^2$

To factor complexity measure in constrained empirical risk minimization, we can use **Ivanov** or **Tikhonov** regularization. where

$$\arg\min_{f \in \mathbf{F}} \frac{1}{n} \sum_{i=1}^{n} l(f(x_i), y_i) \text{ s.t. } \Omega(f) \leq r$$

is Ivanov regularization, and

$$\arg\min_{f \in \mathbf{F}} \frac{1}{n} \sum_{i=1}^{n} l(f(x_i), y_i) + \lambda \Omega(f)$$

is Tikhonov regularization. The two can be shown to be equivalent for many performance measure of f, and Ω, using Lagrangian duality theory.
Plugging in the lasso / ridge definition of Ω to either form, we get lasso / ridge regression.
For example, ridge regression in Tikhonov form looks like

[†]Optimization can be negative, however, this $\hat{R}(\tilde{f}_n) - \hat{R}(\hat{f}_n)$ can't. In this hypothesis space, this function can fit the overall world better than the constrained empirical risk minimizer, but it can't on the training set, on which the constrained ERM performs best in this space

$$\hat{w} = \arg\min_{w \in \mathbf{R}^d} \frac{1}{n} \sum_{i=1}^{n} (w^T x_i - y_i)^2 + \lambda \|w\|_2^2$$

Lasso regression typically lead to feature sparsity (more coefficient equal to 0). This can be shown intuitively by comparing the two on the intersection of contour plot (of empirical risk minimizer) and the area constrained by lasso (diamond on a $f(x) = ax + b$ prediction function) / ridge (circle on the same prediction function) regression, as show in Figure 1[‡].
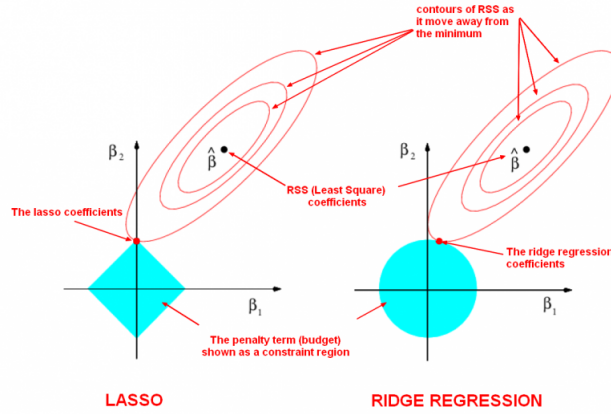

Figure 1: Why lasso regression tends to lead to feature sparsity

Lasso regression (l_1-norm) is not differentiable, thus we cannot apply gradient descent as-is. We split each coefficient into positive and negative parts: rewrite the vector $w = w^+ - w^-$ ($|w| = w^+ + w^-$), and we have this equivalent problem[§]:

$$\arg\min_{w^+, w^-} \frac{1}{n} \sum_{i=1}^{n} ((w^+ - w^-)^T x_i - y_i)^2 + \lambda(w^+ + w^-)$$

subject to $w_i^+ \geq 0 \ \forall i$, $w_i^- \leq 0 \ \forall i$

Now we have a constraint and two vectors, to find the minimum, we can use **projected SGD**[¶] or **coordinate descent**.

Coordinate gradient descent on a vector w works by adjusting only a single w_i in each step, as opposed to possibly alter the entire w in one step as in regular gradient descent. We iteratively adjust each coordinate several times, where we can pick a random one to adjust, or do cyclic adjustment.

[‡]https://niallmartin.wordpress.com/2016/05/12/shrinkage-methods-ridge-and-lasso-regression/

[§]whose equivalence can be proved. And this can be plugged in to a quadratic solver to give us w^+ and w^-

[¶]Normal gradient descent but project / reset each coordinate to the constrained space after each step

Coordinate gradient descent has a **closed form solution** for lasso (in the form below[||]), where we can use a formula to solve each w_i, instead of having to loop over them.

$$\hat{w}_j = \underset{w_j \in \mathbf{R}}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} (w^T x_i - y_i)^2 + \lambda \|w\|_1$$

4 Exercises

4.1 Deriving gradient affine form

1. Given $f(w) = c^t w$, $\triangledown f(w)$?

$$f'(x; u) = \lim_{h \to 0} \frac{f(w + hu) - f(w)}{h} = \lim_{h \to 0} \frac{c^T hu}{h} = c^T u$$

This shows

$$\triangledown f(x) = c$$

2. Given $f(w) = w^T A w$, $\triangledown f(w)$?

$$
\begin{aligned}
f'(w; u) &= \lim_{h \to 0} \frac{f(w + hu) - f(w)}{h} \\
&= \lim_{h \to 0} \frac{(w + hu)^T A(w + hu) - w^T A w}{h} \\
&= \lim_{h \to 0} \frac{w^T A w + h w^T u + h u^T A w + h^2 u^T A u - w^T A w}{h} \\
&= u^T A w + w^T A u \\
&= w^T A^T u + w^T A u
\end{aligned}
$$

This shows
$$\triangledown f(x) = (w^T A^T + w^T A)^T = (A + A^T)w$$

3. Given $f(w) = \|Aw - y\|_2^2$, $\triangledown f(w)$?

$$f(w) = \|Aw - y\|_2^2 = (Aw - y)^T (Aw - y)$$

[||] This form is not differentiable, but it can be shown that for coordinate gradient descent to work there is a weaker condition, which lasso satisfies. *This would indicate that the split into positive and negative shown above is not required, for solving using coordinate gradient descent?*

$$f'(w;u) = \lim_{h \to 0} \frac{(A(w+hu)-y)^T(A(w+hu)-y) - (Aw-y)^T(Aw-y)}{h}$$
$$= \lim_{h \to 0} (A(w+hu)-y)^T Au + (Au)^T(A(w+hu)-y)$$
$$= (Aw-y)^T Au + (Au)^T(Aw-y)$$
$$= 2(Aw-y)^T Au$$

This shows

$$\nabla f(x) = 2((Aw-y)^T A)^T = 2A^T(Aw-y) = 2A^T Aw - 2A^T y$$

4. Given $f(w) = \|Aw - y\|_2^2 + \lambda \|w\|_2^2$, express $f(w) = \|Bw - z\|_2^2$?
The gradient of the two need to be the same, using previous result, we need to solve

$$\begin{cases} A^T A + \lambda & = B^T B \\ A^T y & = B^T z \end{cases}$$

Note the equivalence between extending a matrix and addition, let

$$B = \begin{pmatrix} A \\ \sqrt{\lambda} I_{n \times n} \end{pmatrix} \text{ and } z = \begin{pmatrix} y \\ 0_{n \times 1} \end{pmatrix}$$

written in block-matrix form.

4.2 Recap: linear regression with square loss