Computational Optimization

with Applications to Machine Learning

A Series of Lecture Notes at Missouri S&T

Wenqing Hu *

Contents

*Department of Mathematics and Statistics, Missouri University of Science and Technology (formerly University of Missouri, Rolla). Web: https://huwenqing0606.github.io/ Email: huwen@mst.edu

1 Motivating Examples

1.1 Supervised Learning

EMPIRICAL RISK MINIMIZATION

Supervised Learning: Given training data points $(x_1, y_1), ..., (x_n, y_n)$, construct a learning model $y = g(x, \omega)$ that best fits the training data. Here ω stands for the parameters of the learning model, say $\omega = (\omega_1, ..., \omega_d)$.

Here (x_i, y_i) comes from an independent identically distributed family $(x_i, y_i) \sim p(x, y)$, where $p(x, y)$ is the joint density. The model $x \to y$ is a black-box $p(y|x)$, which is to be fit by $g(x, \omega)$.

"Loss function" $L(g(x, \omega), y)$, for example, can be $L(g(x, \omega), y) = (g(x, \omega) - y)^2$.

Empirical Risk Minimization (ERM):

$$\omega_n^* = \arg\min_{\omega} \frac{1}{n} \sum_{i=1}^{n} L(y_i, g(x_i, \omega)) \ . \tag{1.1}$$

Regularized Empirical Risk Minimization (R-ERM):

$$\omega_n^* = \arg\min_{\omega} \frac{1}{n} \sum_{i=1}^{n} L(y_i, g(x_i, \omega)) + \lambda R(\omega) \ . \tag{1.2}$$

For example, we can take $R(\omega) = \|\omega\|^2 = \omega_1^2 + ... + \omega_d^2$. This regularization helps to control very irregular minimizers (unwanted ω).

In general let $f_i(\omega) = L(y_i, g(x_i, \omega))$ or $f_i(\omega) = L(y_i, g(x_i, \omega)) + \lambda R(\omega)$, then the optimization problem is

$$\omega_n^* = \arg\min_{\omega} \frac{1}{n} \sum_{i=1}^{n} f_i(\omega) \ . \tag{1.3}$$

Key features of nonlinear optimization problem in machine learning: large–scale, nonconvex, ... etc.

Key problems in machine learning: optimization combined with generalization. "Population Loss": $\mathbf{E}L(g(x, \omega), y)$, minimizer

$$\omega^* = \arg\min_{\omega} \mathbf{E}L(g(x, \omega), y) \ .$$

Generalization Error: $\mathbf{E}L(y, g(x, \omega_n^*))$. Consistency: Do we have $\omega_n^* \to \omega^*$? At what speed?

Key problems in optimization: convergence, acceleration, variance reduction.

How can optimization be related to generalization? There are quite abstract notions related to this topic, such as Vapnik–Chervonenkis dimension (VC dimension) and Radmacher complexity, which we might touch later. Also, we want to look at the geometry of the loss landscape, which is closely related to neural network structure.

Loss Functions

Classification Problems: label $y = 1$ or -1. Choice of Loss function $L(y, g)$, $y = 1, -1$. 0/1 Loss: $\ell_{0/1}(y, g) = 1$ if $yg < 0$ and $\ell_{0/1}(y, g) = 0$ otherwise.

(1) Hinge Loss.

$$L(y, g) = \max(0, 1 - yg) \ ; \tag{1.4}$$

(2) Exponential Loss.

$$L(y, g) = \exp(-yg) \ ; \tag{1.5}$$

(3) Cross Entropy Loss.

$$L(y, g) = -\left(I_{\{y=1\}} \ln \frac{e^g}{e^g + e^{-g}} + I_{\{y=-1\}} \ln \frac{e^{-g}}{e^g + e^{-g}} \right) \ . \tag{1.6}$$

This is to use $p(y) = \dfrac{e^{yg}}{e^{yg} + e^{-yg}}$, $y = \pm 1$ and the binary cross entropy as

$$-(I_{\{y=1\}} \ln p(1) + I_{\{y=-1\}} \ln p(-1)) \ .$$

Regression Problems: Choice of Loss function $L(y, g)$.

(1) L^2–Loss.

$$L(y, g) = |y - g|_2^2 \ . \tag{1.7}$$

L^2–norm: $|x|_2^2 = x_1^2 + ... + x_d^2$

(2) L^1–Loss.

$$L(y, g) = |y - g|_1 \ . \tag{1.8}$$

L^1–norm: $|x|_1 = |x_1| + ... + |x_d|$.

(3) L^0–Loss.

$$L(y, g) = |y - g|_0 \ . \tag{1.9}$$

L^0–norm: $|x|_0 = \sharp\{i : x_i \neq 0, 1 \leq i \leq d\}$.

Regularized (penalize) term $R(\omega)$:

L^1–regularized $R(\omega) = |\omega|_1$;

L^0–regularized $R(\omega) = |\omega|_0$.

Learning Models

(1) Linear regression: $g(x, \omega) = \omega^T x$. $g(x, \omega) = \dfrac{1}{1 + \exp(-\omega^T x)}$.

Least squares problem:

$$\min_{\omega \in \mathbb{R}^d} \frac{1}{2m} \sum_{j=1}^{m} (x_j^T \omega - y_j)^2 = \frac{1}{2m} |A\omega - y|_2^2 \tag{1.10}$$

Here training data $(x_1, y_1), ..., (x_m, y_m)$ where $x_i \in \mathbb{R}^d$, $y \in \mathbb{R}$, and $A = \begin{pmatrix} x_1^T \\ ... \\ x_m^T \end{pmatrix}$.

Tikhonov regularization:

$$\min_{\omega} \frac{1}{2m} |A\omega - y|_2^2 + \lambda |\omega|_2^2 \ . \tag{1.11}$$

LASSO (Least Absolute Shrinkage and Selection Operator):

$$\min_{\omega} \frac{1}{2m} |A\omega - y|_2^2 + \lambda |\omega|_1 \ . \tag{1.12}$$

See [4].

(2) Support Vector Machines (SVM):

Set-up: $x_j \in \mathbb{R}^n$, $y_j \in \{1, -1\}$. Separating hyperplane $\omega^T x + \beta = 0$ where $\omega \in \mathbb{R}^n$ and $\beta \in \mathbb{R}$.

Classification Problem: Goal is to find a hyperplane $\omega^T x + \beta = 0$ such that it classifies the two kinds of data points most efficiently. The signed distance of any point $x \in \mathbb{R}^n$ to the hyperplane is given by $r = \dfrac{\omega^T x + \beta}{|\omega|}$. If the classification is good enough, we expect to have $\omega^T x_j + \beta > 0$ when $y = 1$ and $\omega^T x_j + \beta < 0$ when $y = -1$. After rescaling ω and β, we can then formulate the problem as looking for optimal ω and β such that $\omega^T x_j + \beta \geq 1$ when $y_j = 1$ and $\omega^T x_j + \beta \leq -1$ when $y_i = -1$. The closest few data points that match these two inequalities are called "support vectors". The distance to the separating hyperplane created by two support vectors of opposite type is

$$\left|\frac{1}{|\omega|}\right| + \left|\frac{-1}{|\omega|}\right| = \frac{2}{|\omega|} \ .$$

So we can formulate the following optimization problem

$$\max_{\omega \in \mathbb{R}^n, \beta \in \mathbb{R}} \frac{2}{|\omega|} \text{ such that } y_j(\omega^T x_j + \beta) \geq 1 \text{ for } j = 1, 2, ..., m \ .$$

Or in other words we have the *constrained* optimization problem

$$\min_{\omega \in \mathbb{R}^n, \beta \in \mathbb{R}} \frac{1}{2} |\omega|^2 \text{ such that } y_j(\omega^T x_j + \beta) \geq 1 \text{ for } j = 1, 2, ..., m \ . \tag{1.13}$$

"Soft margin": We allow the SVM to make errors on some training data points but we want to minimize the error. In fact, we allow some training data to violate $y_j(\omega^T x_j + \beta) \geq 1$, so that ideally we minimize

$$\min_{\omega \in \mathbb{R}^n, \beta \in \mathbb{R}} \frac{1}{2} |\omega|^2 + C \sum_{j=1}^{m} \ell_{0/1}(y_j(\omega^T x_j + \beta) - 1) \ .$$

Here the 0/1 loss is $\ell_{0/1}(z) = 1$ if $z < 0$ and $\ell_{0/1}(z) = 0$ otherwise, and $C > 0$ is a penalization parameter. We can then turn the 0/1 loss to Hinge Loss, that is why Hinge Loss comes in:

$$\min_{\omega\in\mathbb{R}^n,\beta\in\mathbb{R}} \frac{1}{2}|\omega|^2 + C\sum_{j=1}^{m} \max(0, 1 - y_j(\omega^T x_j + \beta)) \ . \tag{1.14}$$

We can introduce "slack variables" $\xi_i \geq 0$ to introduce weights to classification errors in the above problem. This leads to "Soft margin SVM with slack variables":

$$\min_{\omega\in\mathbb{R}^n,\beta\in\mathbb{R}} \frac{1}{2}|\omega|^2 + C\sum_{j=1}^{m} \xi_j \text{ s.t. } y_j(\omega^T x_j + b) \geq 1 - \xi_j \ , \xi_j \geq 0 \ , j = 1, 2, ..., m \ . \tag{1.15}$$

See [6].

(3) Neural Network: "activation function" σ.

Sigmoid:

$$\sigma(z) = \frac{1}{1 + \exp(-z)} \ ; \tag{1.16}$$

tanh:

$$\sigma(z) = \frac{\exp(z) - \exp(-z)}{\exp(z) + \exp(-z)} \ ; \tag{1.17}$$

ReLU (Rectified Linear Unit):

$$\sigma(z) = \max(0, z) \ . \tag{1.18}$$

Vector–valued activation: if $z \in \mathbb{R}^n$ then $\sigma : \mathbb{R}^n \to \mathbb{R}^n$ is defined by $(\sigma(z))_i = \sigma(z_i)$ where each $\sigma(z_i)$ is the scalar activation function.

Fully connected neural network prediction function

$$g(x, \omega) = a^T \left(\sigma \left(W^{(H)}(\sigma(W^{(H-1)}(...(\sigma(W^{(1)}x + b_1))...) + b_{H-1}) + b_H\right)\right) \ . \tag{1.19}$$

Optimization

$$\min_{\omega} \frac{1}{2}\sum_{i=1}^{n}(g(x_i, \omega) - y_i)^2 \ .$$

Many other different network structures that we do not expand here: convolutional, recurrent (Gate: GRU, LSTM), ResNet, Transformer, ...

Two layer (one hidden layer) fully connected ReLU neural network has specific loss function structure: our $W^{(1)} = \begin{pmatrix} \omega_1^T \\ ... \\ \omega_m^T \end{pmatrix}$ and

$$g(x, \omega) = \sum_{r=1}^{m} a_r \max\left(\omega_r^T x, 0\right) \ . \tag{1.20}$$

Optimization problem is given by

$$\min_{\omega} \frac{1}{2} \sum_{i=1}^{n} \left(\sum_{r=1}^{m} a_r \max\left(\omega_r^T x_i, 0\right) - y_i \right)^2 .$$

Non–convexity issues: see [2].

1.2 Matrix Optimizations

Many machine learning/statistical learning problems are related to matrix optimizations.

(1) Matrix Completion: Each A_j is $n \times p$ matrix, and we seek for another $n \times p$ matrix $\widehat{X}$ such that

$$\widehat{X} = \arg\min_{X} \frac{1}{2m} \sum_{j=1}^{m} (\langle A_j, X\rangle - y_j)^2 \tag{1.21}$$

where $\langle A, B\rangle = \mathrm{tr}(A^T B)$. We can think of the A_j as "probing" the unknown matrix X. In other words, we want the best X such that $\langle A_j, X\rangle \approx y_j$ for all $1 \leq j \leq m$.

(2) Nonnegative Matrix Factorization: If the full matrix $Y \in \mathbb{R}^{n\times p}$ is observed, then we seek for $L \in \mathbb{R}^{n\times r}$ and $R \in \mathbb{R}^{p\times r}$ such that

$$\min_{L,R} \|LR^T - Y\|_F^2 \text{ subject to } L \geq 0 \text{ and } R \geq 0 . \tag{1.22}$$

Here $\|A\|_F = \left(\sum\sum |a_{ij}|^2\right)^{1/2}$ is the Frobenius norm of a matrix A. This is used very often in recommendation systems (see [1]).

See [3] for an overview.

(3) Principle Component Analysis (PCA): Let S be a positive-definite (non-negative definite) matrix of size $n \times n$. Then we can diagonalize it as $Se_i = \lambda_i e_i$, $1 \leq i \leq n$, $\lambda_1 \geq ... \geq \lambda_n > 0$ (or ≥ 0). Let $v \in \mathbb{R}^n$ be written as $v = v_1 e_1 + ... + v_n e_n$. Then

$$v^T S v = \lambda_1 v_1^2 + ... + \lambda_n v_n^2$$

is a quadratic form. If we restrict $v_1^2 + ... + v_n^2 = 1$, then the maximum of above quadratic form will give the direction of e_1 and value λ_1 (principle component).

PCA:

$$\max_{v\in\mathbb{R}^n} v^T S v \text{ such that } \|v\|_2 = 1 \ , \ \|v\|_0 \leq k \ . \tag{1.23}$$

Here S is a positive-definite (or non-negative definite) matrix. The objective function is convex, but if you take into account the constraint, then this problem becomes non–convex. A picture for dimension 1 example can be shown below.

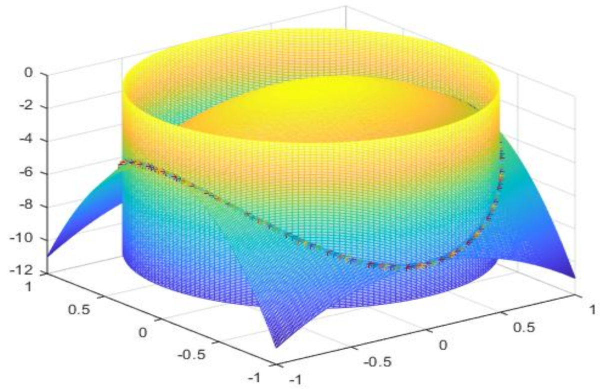

Figure 1: Loss Landscape of 1-dimensional PCA.

Online PCA: see [5].

(4) Sparse inverse covariance matrix estimation: Sample covariance matrix $S = \dfrac{1}{m-1}\sum\limits_{j=1}^{m} a_j a_j^T$. $S^{-1} = X$. "Graphical LASSO":

$$\min_{X \in \text{ Symmetric } \mathbb{R}^{n\times n}, X \succeq 0} \langle S, X\rangle - \ln\det X + \lambda \|X\|_1 \tag{1.24}$$

where $\|X\|_1 = \sum |X_{ij}|$.

2 Convex Functions

2.1 Optimization problem: set-up and its solutions

For the objective function $f : \mathbb{R}^n \supset \mathcal{D} \to \mathbb{R}$, we have various notions of minimizers:

- Local minimizer: $x^* \in \mathcal{D}$ is a local minimizer if there exist a neighborhood $\mathcal{N}$ containing x^*, such that $f(x) \geq f(x^*)$ for all $x \in \mathcal{N} \cap \mathcal{D}$;
- Global minimizer: $x^* \in \mathcal{D}$ is a global minimizer if $f(x) \geq f(x^*)$ for all $x \in \mathcal{D}$;
- Strict Local Minimizer: $x^* \in \mathcal{D}$ is a strict local minimizer iff x^* is a local minimizer and $f(x) \geq f(x^*)$ if $x \neq x^*$, $x \in \mathcal{N}$;
- Isolated Local Minimizer: $x^* \in \mathcal{D}$ is an isolated local minimizer iff there exists a neighborhood $\mathcal{N}$ containing x^* such that $f(x) \geq f(x^*)$ for all $x \in \mathcal{N} \cap \mathcal{D}$ and $\mathcal{N}$ does not contain any other local minimizers.

Constrained optimization problem

$$\min_{x\in\Omega} f(x)\ , \tag{2.1}$$

where $\Omega \subset \mathcal{D} \subset \mathbb{R}^n$ is a closed set.

x^* is a *local solution*: there exist a neighborhood $\mathcal{N}$ containing x^* such that $f(x) \geq f(x^*)$ for any $x \in \mathcal{N} \cap \Omega$.

x^* is a *global solution*: $f(x) \geq f(x^*)$ for any $x \in \Omega$.

2.2 Convexity

Definition 2.1 (Convex Set)**.** *A set Ω is called a convex set if for any $x, y \in \Omega$ we have*

$$(1-\alpha)x + \alpha y \in \Omega$$

for all $\alpha \in [0,1]$.

Given convex set $\Omega \subset \mathbb{R}^n$, the *projection operator* $P : \mathbb{R}^n \to \Omega$ is given by

$$P(y) = \arg\min_{z \in \Omega} \|z - y\|_2^2 \ . \tag{2.2}$$

$P(y)$ is the point in Ω that is closest to y in the sense of Euclidean norm. If $x \in \Omega$, then $P(x) = x$.

Definition 2.2 (Convex Function)**.** *A function $\phi : \mathbb{R}^n \to \mathbb{R} \cup \{\pm\infty\}$ is convex if for all $x, y \in \mathbb{R}^n$ we have*

$$\phi((1-\alpha)x + \alpha y) \leq (1-\alpha)\phi(x) + \alpha\phi(y) \tag{2.3}$$

for all $\alpha \in [0,1]$.

Definition 2.3 (Strongly Convex Function)**.** *A convex function ϕ is called strongly convex with modulus of convexity $m > 0$ if*

$$\phi((1-\alpha)x + \alpha y) \leq (1-\alpha)\phi(x) + \alpha\phi(y) - \frac{1}{2}m\alpha(1-\alpha)\|x-y\|_2^2 \tag{2.4}$$

for all x, y in the domain of ϕ.

Let $\Omega \subset \mathbb{R}^n$ be convex. We define the indicator function

$$I_\Omega(x) = \begin{cases} 0 & \text{if } x \in \Omega \ , \\ +\infty & \text{otherwise} \ . \end{cases} \tag{2.5}$$

Constrained optimization problem $\min_{x \in \Omega} f(x)$ is the same thing as the unconstrained optimization problem $\min[f(x) + I_\Omega(x)]$. We can set a sequence of functions $F_{\lambda\Omega} \uparrow I_\Omega(x)$ as $\lambda \uparrow \infty$, and then we solve the relaxed problem $\min_{x \in \mathbb{R}^d}[f(x) + F_{\lambda\Omega}(x)]$.

Theorem 2.1 (Minimizers for convex functions)**.** *If the function f is convex and the set Ω is closed and convex, then*

(a) Any local solution of (2.1) is also global;

(b) The set of global solutions of (2.1) is a convex set.

A few more auxiliary notions:

- "Effective domain" of ϕ: $\{x \in \Omega : \phi(x) < \infty\}$;
- "Epigraph" of ϕ:
$$\mathrm{epi}(\phi) = \{(x,t) \in \Omega \times \mathbb{R} : t \geq \phi(x)\} \ .$$
- ϕ is a "proper convex function" if $\phi(x) < \infty$ for some $x \in \Omega$ and $\phi(x) > -\infty$ for all $x \in \Omega$;
- ϕ is a "closed proper convex function" if ϕ is a proper convex function and $\{x \in \Omega : \phi(x) \leq t\}$ is a closed set for all $t \in \mathbb{R}$.

2.3 Taylor's theorem and convexity in Taylor's expansion

Theorem 2.2 (Taylor's theorem)**.** *Given a continuously differentiable function* $f : \mathbb{R}^n \to \mathbb{R}$ *and given* $x, p \in \mathbb{R}^n$ *we have*

$$f(x+p) = f(x) + \int_0^1 \nabla f(x+\gamma p)^T p d\gamma \ , \tag{2.6}$$

$$f(x+p) = f(x) + \nabla f(x+\gamma p)^T p \ , \quad \textit{for some } \gamma \in (0,1) \ . \tag{2.7}$$

If f *is twice continuously differentiable, then*

$$\nabla f(x+p) = \nabla f(x) + \int_0^1 \nabla^2 f(x+\gamma p)^T p d\gamma \ , \tag{2.8}$$

$$f(x+p) = f(x) + \nabla f(x)^T p + \frac{1}{2} p^T \nabla^2 f(x+\gamma p) p \ , \quad \textit{for some } \gamma \in (0,1) \ . \tag{2.9}$$

Proof. (1) Let $g(\gamma) = f(x+\gamma p)$, then $g'(\gamma) = \nabla f(x+\gamma p) \cdot p = (\nabla f(x+\gamma p))^T p$. Then by Newton-Leibniz we have

$$g(1) - g(0) = \int_0^1 g'(\gamma) d\gamma = \int_0^1 (\nabla f(x+\gamma p))^T p d\gamma \ ,$$

which gives (2.6).

(2) Using mean-value theorem $\int_a^b h(t)dt = (b-a)h(\xi)$ for some $\xi \in (a,b)$, we get

$$\int_0^1 (\nabla f(x+\gamma p))^T p d\gamma = (\nabla f(x+\gamma_0 p))^T p$$

for some $\gamma_0 \in (0,1)$, which is (2.7).

Another way is to use the fact that

$$f(y) = f(x) + (\nabla f(x+\gamma_0(y-x)))^T (y-x) \ .$$

Set $p = y - x$, this gives

$$f(x+p) = f(x) + (\nabla f(x+\gamma_0 p))^T p \ .$$

(3) We apply (2.8) to each of $\dfrac{\partial f}{\partial x_i}$, and we get

$$\frac{\partial f}{\partial x_i}(x+p) = \frac{\partial f}{\partial x_i}(x) + \int_0^1 \left[\nabla\left(\frac{\partial f}{\partial x_i}\right)(x+\gamma p)\right]^T p d\gamma \ .$$

Introduce the Hessian matrix $\nabla^2 f(x) = \left(\dfrac{\partial^2 f}{\partial x_i \partial x_j}\right)_{1\le i,j\le n}$. Then when we put the above equation for all $1 \le i \le n$, we get (2.8).

(4) We combine (2.8) with (2.7) to get

$$\begin{aligned} f(x+p) &= f(x) + \nabla f(x+\gamma p)^T p \\ &= f(x) + \left(\nabla f(x) + \nabla^2 f(x+\mu\gamma p)^T p\right)^T p \\ &= f(x) + \nabla f(x)^T p + p^T \nabla^2 f(x+\mu\gamma p) p \end{aligned}$$

for some $\mu \in (0,1), \gamma \in (0,1)$. □

Definition 2.4 (L–Lipschitz)**.** *If $f : \mathbb{R}^n \to \mathbb{R}$ is such that*

$$|f(x) - f(y)| \le L\|x - y\| \ , \tag{2.10}$$

for $L > 0$ and any $x, y \in \mathbb{R}^n$, then $f(x)$ is L–Lipschitz.

Optimization literatures often assume that ∇f is L–Lipschitz

$$\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\| \ . \tag{2.11}$$

Theorem 2.3. *(1) If f is continuously differentiable and convex then*

$$f(y) \ge f(x) + (\nabla f(x))^T (y - x) \tag{2.12}$$

for any $x, y \in dom(f)$.

(2) If f is differentiable and m–strongly convex then

$$f(y) \ge f(x) + (\nabla f(x))^T (y - x) + \frac{m}{2}\|y - x\|^2 \tag{2.13}$$

for any $x, y \in dom(f)$.

(3) If ∇f is uniformly Lipschitz continuous with Lipschitz constant $L > 0$ and f is convex then

$$f(y) \le f(x) + (\nabla f(x))^T (y - x) + \frac{L}{2}\|y - x\|^2 \tag{2.14}$$

for any $x, y \in dom(f)$.

Theorem 2.4. *Suppose that the function f is twice continuously differentiable on $\mathbb{R}^n$. Then*

(1) f is strongly convex with modulus of convexity m if and only if $\nabla^2 f(x) \succeq mI$ for all x;

(2) ∇f is Lipschitz continuous with Lipschitz constant L if and only if $\nabla^2 f(x) \preceq LI$ for all x.

Proof. (Part (1)) Suppose that the function f is strongly–m convex with the standard inequality (2.4) above for m–convexity. Set $z_\alpha = (1-\alpha)x + \alpha y$. Then by (2.4) we have

$$\alpha f(y) - \alpha f(x) \geq f(z_\alpha) - f(x) + \frac{1}{2} m\alpha(1-\alpha)\|x-y\|^2 \ .$$

Making use of Taylor's expansion

$$\alpha f(y) - \alpha f(x) \geq (\nabla f(x))^T (z_\alpha - x) + O(\|z_\alpha - x\|^2) + \frac{1}{2} m\alpha(1-\alpha)\|x-y\|^2 \ .$$

Since $z_\alpha \to x$ as $\alpha \to 0$, and $z_\alpha - x = \alpha(y-x)$, we can set $\alpha \to 0$ to get from above that for any $x, y \in \mathbb{R}^n$

$$f(y) \geq f(x) + (\nabla f(x))^T (y-x) + \frac{m}{2}\|y-x\|^2 \ . \tag{2.15}$$

Set $u \in \mathbb{R}^n$ and $\alpha > 0$, then we consider the Taylor's expansion

$$f(x+\alpha u) = f(x) + \alpha \nabla f(x)^T u + \frac{1}{2}\alpha^2 u^T \nabla^2 f(x + t\alpha u) u \tag{2.16}$$

for some $0 \leq t \leq 1$. We apply (2.15) with $y = x + \alpha u$ so that

$$f(x+\alpha u) \geq f(x) + \alpha(\nabla f(x))^T u + \frac{m}{2}\alpha^2 \|u\|^2 \ . \tag{2.17}$$

Comparing (2.16) and (2.17) we see that for arbitrary choice of $u \in \mathbb{R}^n$ we have

$$u^T \nabla^2 f(x + t\alpha u) u \geq m\|u\|^2 \ .$$

This implies that $\nabla^2 f(x) \succeq mI$ as claimed. This shows the "only if" part.

Indeed one can also show the "if" part. To this end assume that $\nabla^2 f(x) \succeq mI$. Then for any $z \in \mathbb{R}^n$ we have that $(x-z)^T \nabla^2 f(z + t(x-z))(x-z) \geq m\|x-z\|^2$. Thus

$$\begin{aligned} f(x) &= f(z) + (\nabla f(z))^T (x-z) + \frac{1}{2}(x-z)^T \nabla^2 f(z + t(x-z))(x-z) \\ &\geq f(z) + (\nabla f(z))^T (x-z) + \frac{m}{2}\|x-z\|^2 \ . \end{aligned} \tag{2.18}$$

Similarly

$$\begin{aligned} f(y) &= f(z) + (\nabla f(z))^T (y-z) + \frac{1}{2}(y-z)^T \nabla^2 f(z + t(y-z))(y-z) \\ &\geq f(z) + (\nabla f(z))^T (y-z) + \frac{m}{2}\|y-z\|^2 \ . \end{aligned} \tag{2.19}$$

We consider $(1-\alpha)$(2.18)+α(2.19) and we set $z = (1-\alpha)x + \alpha y$. This gives

$$\begin{aligned}
&(1-\alpha)f(x) + \alpha f(y) \\
&\geq (\alpha + (1-\alpha))f(z) + (\nabla f(z))^T((1-\alpha)(x-z) + \alpha(y-z)) + \frac{m}{2}\left((1-\alpha)\|x-z\|^2 + \alpha\|y-z\|^2\right) \\
&= f(z) + (\nabla f(z))^T((1-\alpha)(x-z) + \alpha(y-z)) + \frac{m}{2}\left((1-\alpha)\|x-z\|^2 + \alpha\|y-z\|^2\right) \ .
\end{aligned} \tag{2.20}$$

Since $x-z = \alpha(x-y)$ and $y-z = (1-\alpha)(y-x)$, we see that $((1-\alpha)(x-z)+\alpha(y-z)) = 0$. Moreover, this means that

$$(1-\alpha)\|x-z\|^2 + \alpha\|y-z\|^2 = \left[(1-\alpha)\alpha^2 + \alpha(1-\alpha)^2\right]\|x-y\|^2 = \alpha(1-\alpha)\|x-y\|^2 \ .$$

From these we see that (2.20) is the same as saying

$$(1-\alpha)f(x) + \alpha f(y) \geq f((1-\alpha)x + \alpha y) + \frac{m}{2}\alpha(1-\alpha)\|x-y\|^2 \ ,$$

which is (2.4). □

Theorem 2.5 (Strongly convex functions have unique minimizers)**.** *Let f be differentiable and strongly convex with modulus $m > 0$. Then the minimizer x^* of f exists and is unique.*

More technical issues...

Theorem 2.6. *Let f be convex and continuously differentiable and ∇f be L–Lipschitz. Then for any $x, y \in dom(f)$ the following bounds hold:*

$$f(x) + \nabla f(x)^T(y-x) + \frac{1}{2L}\|\nabla f(x) - \nabla f(y)\|^2 \leq f(y) \ , \tag{2.21}$$

$$\frac{1}{L}\|\nabla f(x) - \nabla f(y)\|^2 \leq (\nabla f(x) - \nabla f(y))^T(x-y) \leq L\|x-y\|^2 \ . \tag{2.22}$$

In addition, let f be strongly convex with modulus m and unique minimizer x^*. Then for any $x, y \in dom(f)$ we have that

$$f(y) - f(x) \geq -\frac{1}{2m}\|\nabla f(x)\|^2 \ . \tag{2.23}$$

Generalized Strong Convexity: $(f^* = f(x^*))$

$$\|\nabla f(x)\|^2 \geq 2m[f(x) - f^*] \text{ for some } m > 0 \ . \tag{2.24}$$

It holds in situations other than when f is strongly convex.

2.4 Optimality Conditions

Smooth unconstrained problem

$$\min_{x \in \mathbb{R}^n} f(x) \ , \tag{2.25}$$

where $f(x)$ is a smooth function.

Theorem 2.7 (Necessary conditions for smooth unconstrained optimization)**.** *We have*

- *(a) (first–order necessary condition) Suppose that f is continuously differentiable. Then if x^* is a local minimizer of* (2.25)*, then $\nabla f(x^*) = 0$;*
- *(b) (second–order necessary condition) Suppose that f is twice continuously differentiable. Then if x^* is a local minimizer of* (2.25)*, then $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*)$ is positive definite.*

Theorem 2.8 (Sufficient conditions for convex functions in smooth unconstrained optimization)**.** *If f is continuously differentiable and convex, then if $\nabla f(x^*) = 0$, then x^* is a global minimizer of* (2.25)*. In addition, if f is strongly convex, then x^* is the unique global minimizer.*

Theorem 2.9 (Sufficient conditions for nonconvex functions in smooth unconstrained optimization)**.** *Suppose that f is twice continuously differentiable and that for some x^* we have $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*)$ is positive definite, then x^* is a strict local minimizer of* (2.25)*.*

References

[1] https://en.wikipedia.org/wiki/Matrix_factorization_(recommender_systems).

[2] Alon Brutzkus, Amir Globerson, Eran Malach, and Shai Shalev-Shwartz. Sgd learns over–parameterized networks that provably generalize on linearly separable data. *ICLR, arXiv:1710.10174*, 12:1–12, 2018.

[3] Y. Chi, Y.M. Lu, and Y. Chen. Nonconvex optimization meets low-rank matrix factorization: An overview. *IEEE Transactions on Signal Processing*, 67(20), 2019.

[4] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009.

[5] C.J Li, M. Wang, H. Liu, and T. Zhang. Near–optimal stochastic approximation for online principal component estimation. *Mathematical Programming*, 167(1):75–97, 2018.

[6] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 2000.