

2024 後期

均衡分析と数理最適化

最適化法概論

経済研究所 大澤 実

osawa.minoru.4z@kyoto-u.ac.jp

目的

変分不等式 (variational inequalities; VI) 理論は，最適化法（数理計画法）の一分野です．VIを学ぶ前提として，最適化法へ入門します．

目標は次の通りです：

- 最適化問題とそれに関わる用語を知る
- 最適化問題の具体例をみる
- 最適化問題の主要な問題クラスを知る
- 特に重要な問題クラスとして凸最適化問題を知る．

あわせて，本講義で使う算数の基礎を復習します．

基本的な定義

最適化問題 (optimization problem)

最適化問題 (数理計画問題) とは, 集合 S と関数 $f : S \rightarrow \mathbb{R}$ が与えられたときに $x \in S$ を満たす x のうち, $f(x)$ を最小にするものを求める問題.

minimize: $f(x)$

subject to: $x \in S$

- minimize を min. とか「 $f(x) \rightarrow$ 最小化」とか書くこともある
- f の最大化問題 $\Leftrightarrow -f$ の最小化問題

最適化問題の性質や解法に関する学問を**最適化法**または**数理計画法**と呼ぶ.

補足 最適化問題 (数理計画問題) は問題そのものを, 最適化法・数理計画法は問題を解く方法の研究および応用の営み全般を指す.

基本用語

minimize: $f(x)$

subject to: $x \in S$

- f を **目的関数** (objective function),
 S を **実行可能領域** or **許容領域** (feasible region),
条件 $x \in S$ を **制約** (constraint) と呼ぶ.
- 制約を満足する点 $x \in S$ を
実行可能解 or **許容解** (feasible solution) と呼ぶ.
- f の最小値を達成する点 $\bar{x} \in S$ を **最適解** (optimal solution),
 $f(\bar{x})$ を **最適値** (optimal value) と呼ぶ.

※ 特に断らない限り, $S \subseteq \mathbb{R}^n$ を仮定する.

記号等

- n 次元数ベクトル $x \in \mathbb{R}^n$ を $(x_i)_{i=1}^n$, (x_1, x_2, \dots, x_n) , (x_i) 等
- 適切な次元の行列 A との積を考えるときは全て列ベクトルとして扱う：
 $Ax, x^\top A$ (\top は転置)
- 同一の次元の数ベクトル x, y に対して $\langle x, y \rangle \equiv \sum_i x_i y_i$ とする.
- 特に断らない限り, 数ベクトル x に対して $\|x\| \equiv \left(\sum_i x_i^2\right)^{1/2}$.

最適解

定義 最適化問題の実行可能解 $\bar{x} \in S$ に対して、ある $\epsilon > 0$ が存在して

$$f(x) \geq f(\bar{x}), \quad \forall x \in S, \|x - \bar{x}\| < \epsilon$$

が成立するとき、 \bar{x} を **局所的最適解** (local optimal solution) と呼ぶ。

特に、あらゆる ϵ に対して $\|x - \bar{x}\| < \epsilon$ なる \bar{x} 以外の局所的最適解 x が存在しないなら **孤立局所的最適解** (isolated local optimal solution) と呼ぶ。

定義 最適化問題の実行可能解 $\bar{x} \in S$ に対して、

$$f(x) \geq f(\bar{x}), \quad \forall x \in S$$

が成立するとき、 \bar{x} を **大域的最適解** (global optimal solution) と呼ぶ。

最適解の存在と実行可能性

最適化問題に対して、以下のいずれかが成立する。

1. **実行可能** (feasible) : 実行可能解が存在する ($S \neq \emptyset$)
 - (a) 実行可能領域 S において目的関数 f の値が下に有界であり、かつ下限を達成する実行可能解 (= 最適解) が存在する。
 - (b) S において f の値が下に有界であるが、下限を達成する実行可能解が S に存在しない。
 - (c) S において f の値が下に有界でない (**非有界**; unbounded)
2. **実行不能** (infeasible) : 実行可能解が存在しない ($S = \emptyset$)

1(a) の場合にのみ最適解が存在する。

Quiz 1(b) となる状況の例を複数考えてみよ。

応用における最適化問題

応用上， S は具体的な制約関数たちによって表現される．

minimize: $f(x)$

subject to: $g_j(x) \leq 0, \quad (j = 1, 2, \dots, J)$

$h_i(x) = 0, \quad (i = 1, 2, \dots, I)$

- 非負制約 $x_i \geq 0$ は $g(x_i) = -x_i \leq 0$

例 効用最大化問題

n 種の財の消費量 $x = (x_i)_{i=1}^n$, 価格 $p = (p_i)_{i=1}^n$, 所得 $y > 0$

効用関数が準凹関数 $U(x)$ で与えられるとき

maximize: $U(x)$

subject to: $\sum_i p_i x_i \leq y, x \geq \mathbf{0}$

目的関数を $f(x) = -U(x)$, 実行可能領域を $S \equiv \{\langle p, x \rangle \leq y, x \geq \mathbf{0}\}$ とすれば最初に記載した形式と一致.

このような静学的問題は, U に扱いやすい関数を選べば解析的に解ける.

例 施設配置問題

n 世帯が居住する村に集会所を設置しよう．村の土地を $S \subset \mathbb{R}^2$ とする．世帯 k の座標を $a^k = (a_1^k, a_2^k) \in S$ で，集会所の座標を $x \in S$ で表す．

1. 学校までの二乗距離を平均的に最小にするなら

$$\min_{x \in S} \frac{1}{n} \sum_{k=1}^n \|a^k - x\|^2$$

2. 学校までの距離を平均的に最小にするなら

$$\min_{x \in S} \frac{1}{n} \sum_{k=1}^n \|a^k - x\|$$

3. 最も遠い家から学校までの距離を最小にするなら

$$\min_{x \in S} \max_k \|a^k - x\|$$

Quiz 1 の最適解を求め， S の形状との関係を考察せよ．

教訓： S の形状は解が良い性質を持つかどうかに影響．

例 線形回帰と最小二乗法

m 個のデータ点 (a_i, b_i) ($i = 1, 2, \dots, m$) が与えられている。

ただし, $a_i = (a_{ij})_{j=1}^n \in \mathbb{R}^n$ であり, $b_i \in \mathbb{R}$.

共通の $x \in \mathbb{R}^n$ で各点を $b_i = a_i^\top x$ の形の線形式で近似したい. この近似のよさは, $A = (a_{ij})$ として**二乗誤差**で測ってみることが考えられる.

$$f(x) \equiv \|Ax - b\|^2 = \sum_{i=1}^m \left(a_i^\top x - b_i \right)^2$$

この f を x について最小化してみることが考えられる: $\min_{x \in \mathbb{R}^n} f(x)$.

解 \bar{x} は線形関数のもとで, 二乗誤差の意味でデータを最もよく近似する.

x は全空間 \mathbb{R}^n を動いてよい. このような問題を**無制約最適化問題**と呼ぶ.

例 栄養問題 (diet problem)

野菜 $j = 1, 2, \dots, n$, 栄養素 $i = 1, 2, \dots, m$ がある.

必要な最低限度の栄養素の量 $b = (b_i)_{i=1}^m$ を満足しつつ,
原料費の合計を最小にする野菜購入量 $x = (x_j)_{j=1}^n$ を決めたい.

野菜 j について, 単位数量あたりの価格は c_j , 栄養素 i の含有量は a_{ij} .

Quiz 最適化問題として表現してみよ.

例 輸送計画問題 (transportation problem)

m 箇所の工場から n 箇所の顧客へ納入したい。ただし、

- 工場 i の生産量の上限は a_i
- 顧客 j の需要量は b_j
- 工場 i から顧客 j への単位あたり輸送費用は c_{ij}

このとき総輸送費用を最小化する輸送量 $x = (x_{ij})$ を決めたい。
ただし x_{ij} は工場 i から顧客 j への輸送量を意味する。

Quiz 最適化問題として表現してみよ。

例 線形計画問題 (linear programming; LP)

栄養計画問題および輸送計画問題は**線形計画問題**，すなわち目的関数・制約のすべてが線形関数で与えられる最適化問題の例である。

$$\text{maximize: } \sum_{j=1}^n c_j x_j,$$

$$\text{subject to: } \sum_{j=1}^n a_{ij} x_j \leq b_i \quad (i = 1, \dots, m),$$

$$x_j \geq 0 \quad (j = 1, \dots, n).$$

最適化問題の分類

- 制約付最適化 (constrained) vs. 無制約最適化 (unconstrained)
- 線形 (linear) vs. 非線形 (nonlinear)
- 凸 (convex) vs. 非凸 (nonconvex)
- 連続 (continuous) vs. 離散 (discrete)
- 静学的 (static) vs. 動学的 (dynamic)

本講義では、基本的に静学的な連続最適化を取り扱う。

凸最適化問題

... in fact, the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity.

– R. Tyrrell Rockafellar, in SIAM Review, 1993

復習 集合と関数の凸性

凸集合 (convex set) S は次の条件を満足する集合である：

$$\theta x + (1 - \theta)y \in S, \quad \forall x, y \in S, \theta \in [0, 1].$$

凸関数 (convex function) f とは次の条件を満足する関数である：

$$\theta f(x) + (1 - \theta)f(y) \geq f(\theta x + (1 - \theta)y), \quad \forall x, y, \forall \theta \in [0, 1].$$

不等号が狭義で成立するならば**狭義凸関数**と呼ぶ。

f の定義域を適当な集合 S へ制限した場合についても同様に定義される。

凸最適化問題

次の最適化問題において、 f が S 上凸関数であり、 S が凸集合であるとき、**凸最適化問題** (convex optimization problem) と呼ぶ。

minimize: $f(x)$

subject to: $x \in S$

- 凸最適化問題には様々な著しくよい性質がある。例えば
 - 局所的最適解が存在するならば大域的最適解となる。
 - 最適解の集合が凸集合となる。
 - 目的関数が狭義凸関数なら、最適解が存在すれば唯一である。
 - **双対性** (duality) により問題の等価変換・計算の効率化が可能。

制約関数による凸最適化問題の特徴づけ

応用上, S は具体的な制約関数たちによって表現された.

minimize: $f(x)$

subject to: $g_j(x) \leq 0, \quad (j = 1, 2, \dots, J)$

$h_i(x) = 0, \quad (i = 1, 2, \dots, I)$

凸最適化問題になるのは以下が成立するときである:

- f が凸関数
- $\{g_j\}$ が全て凸関数
- $\{h_i\}$ が全て affine 関数 ($h_i(x) = \langle a_i, x \rangle + b_i$)

Quiz このとき確かに上の問題が凸最適化問題になることを確かめよ.

余談 解きやすさと“面白さ”

- 現実の問題は**計算機利用が前提**．数千以上の変数を持ち解析解はない．
- 都市・地域の分析においても，均衡モデルを用いた定量分析が盛ん．計算機が大いに活用されている．
- 問題の数理的性質の特徴づけは意味解釈および数値計算の効率化の双方で有用．大抵 f や S のようすは自明ではない．
- 実社会では計算機で解き易い問題に落とし込むことはモデリングの鍵．凸最適化はその代表例．
- しかし，現象理解のための理論としては解き易さ = 興味深さではない．例えば問題の凸性（解き易さ）は均衡の一意性に対応しており，複数均衡を許す理論モデルの面白さ・表現力を取り除く．
- 解きやすくかつ面白いモデルを作るのはとても難しい．
- 凸計画問題であっても表現力を持つクラスは存在．応用が求められる．

数值解法

目的

応用において、最適化問題は計算機によって解くことになる。

よく調べられた性質のよい問題に対してはソルバがある。経済学の研究では自分でプログラムを組む必要がある場合の方が多い。

- 問題が大規模になった場合、素朴な定式化ではメモリ不足になったりするため、自分でアルゴリズムを工夫する必要がある。
- 非線形計画問題の多くに対しては安定したソルバは存在しない。

以上のような背景を念頭に、数値計算法の一般的な考え方を学ぶ：

- 数値計算法における反復法 (iterative method) の考え方を知る
- 無制約最適化問題とそれに対する反復法を知る
- 制約つき最適化問題に対する反復法を知る

反復法 (iterative method)

反復法は問題の前提条件が与えられたとき，その数値解を**反復計算**により求めるための数値計算アルゴリズムの総称．

e.g., 非線形方程式の解法としての **Newton 法**

- **数値解**：可能な限り所望の条件を満足するような変数の値
e.g., 方程式 $F(x) = \mathbf{0}$ に対する数値解： $\|F(x)\| < \delta \ll 1$ なる x
- **反復計算**：初期解 x^0 から x^k を反復的に更新 $x^k \mapsto x^{k+1}$ して，**解候補**の列 $\{x^k\}_{k=0}^{\infty}$ を得ること

対義語：**直接法** (direct method)

有限回の手続きで解が定まるアルゴリズムの総称．

e.g., 線形方程式の解法としての **Gauss の消去法** (掃き出し法)

最適化問題に対する反復法

最適化問題 $\min_{x \in S} f(x)$ に対しては，初期実行可能解 (initial feasible solution) $x^0 \in S$ からスタートして解を更新していく $x^k \mapsto x^{k+1} \in S$ ことで**局所的最適解**を発見しようとする。

- 反復法により**大域的最適解**を発見することは一般の問題に対しては困難。大域的最適化 (global optimization) に対しては**メタヒューリスティクス** (metaheuristics) と総称される様々なアルゴリズムがあるが，本講義では取り扱わない。

e.g., 焼きなまし法，遺伝的アルゴリズム

- 凸最適化問題では全ての局所的最適解が大域的最適解となる。

どのような解更新を行うかによって，それぞれ別の解法となる。

復習 スカラー値関数の勾配と Hesse 行列

$f : \mathbb{R}^n \rightarrow \mathbb{R}$ に対して、その点 x における**勾配** (gradient) $\nabla f(x)$ は

$$\nabla f(x) \equiv \left(\frac{\partial f}{\partial x_1}(x), \frac{\partial f}{\partial x_2}(x), \dots, \frac{\partial f}{\partial x_n}(x) \right)^\top$$

で定義され、また **Hesse 行列** (Hessian matrix) $\nabla^2 f(x)$ は

$$\left[\nabla^2 f(x) \right]_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}(x), \quad i, j = 1, \dots, n$$

で定義される。

復習 スカラー値関数の微分可能性と近似

$\|\delta\| \rightarrow 0$ のとき, $f : \mathbb{R}^n \rightarrow \mathbb{R}$ が

$$f(x + \delta) = f(x) + \nabla f(x)^\top \delta + o(\|\delta\|)$$

$$f(x + \delta) = f(x) + \nabla f(x)^\top \delta + \frac{1}{2} \delta^\top \nabla^2 f(x) \delta + o(\|\delta\|^2)$$

を満足するかどうかを考える. $o(h(\delta))$ は $\delta \rightarrow 0$ において $h(\delta)$ より速く小さくなる項.

- **1回微分可能** であるとは全 x において $\nabla f(x)$ が存在すること.
- **2回微分可能** であるとは全 x において $\nabla^2 f(x)$ が存在すること.

復習 スカラー値関数の勾配の性質

スカラー値関数 f の等高線 (contour) と勾配 ∇f とは直交する。

証明 $c \in \mathbb{R}$ に対して関数 f がその値をとる点の集合を

$$f_c = \left\{ x \in \mathbb{R}^n \mid f(x) = c \right\}$$

とし、実数 $t \in \mathbb{R}$ に対して f_c の一点を対応させる写像を $x(t)$ で表現する。

このとき、 f_c 上で $f(x(t)) = c$ なる関係を両辺微分して $t = 0$ とすると

$$\frac{df(x(0))}{dt} = \nabla f(x(0))^\top \frac{dx(0)}{dt} = 0$$

復習 スカラー値関数の勾配の性質

スカラー値関数 f の勾配 ∇f は f が最も増加する方向を示す。

f が1回微分可能ならば $f(x + \delta) \approx f(x) + \nabla f(x)^\top \delta$ が小さい δ に対して成立する。いま、 δ だけ動いた際の f の増加量を

$$\Delta f(\delta; x) \equiv \langle \nabla f(x), \delta \rangle \quad (\approx f(x + \delta) - f(x))$$

としよう。 \mathbb{R}^n における通常の内積は、 $\nabla f(x)$ と δ のなす角を θ とすれば

$$\langle \nabla f(x), \delta \rangle = |\nabla f(x)| |\delta| \cos(\theta)$$

であったから、 $|\delta|$ が固定のとき、 $\Delta f(\delta; x)$ は $\theta = 0$ で最大値をとる。

反復法の基本

最適化問題に対する反復法の基本形は次の通り。

$$x^{k+1} := x^k + \alpha^k d^k$$

ここで $x^k \in S$ は k 回反復時点における解候補であり、

- $d^k \in \mathbb{R}^n$ を**探索方向** (search direction),
- $\alpha^k \in \mathbb{R}$ を**ステップ幅** (step size), と呼ぶ。

反復法の**収束判定条件**は様々な基準があるが、局所最適解であるための**最適性条件** (optimality conditions) を確認することが基本的な考え方。

最適性条件については後に非線形最適化問題を題材に系統的に学ぶ。

降下方向

素朴には、 $x^{k+1} := x^k + \alpha^k d^k$ のような反復で局所的最適解が求まるためには、探索方向 d^k が目的関数を減少させる方向であることが必要であろうと考えられる。つまり d^k は

$$\Delta f(\delta; x) = \langle \nabla f(x), \delta \rangle < 0$$

を満足していてほしいということを意味する。

この条件を満足する d^k を**降下方向** (descent direction) と呼ぶ。

※一部の反復法 (**加速勾配法**等) ではこの条件は必ずしも満足されない。

反復法の基本

- 探索方向 d^k およびステップ幅 α^k をどう設定するかについて、様々な方針がある。最もわかりやすいのは、**最急降下法** (method of steepest descent) で、 $d^k = -\nabla f(x^k)$ とする。
 - 微分情報を使わない方法 (derivative-free method)
 - 勾配の情報を使う方法 (first-order method/gradient method)
 - Hesse 行列の情報を使う方法 (second-order method)
- 理論上は目的関数の情報をより多く使った方が収束のスピードが速い (解に収束するまでに必要な反復回数が少なくなる)。
- 問題の次元が高くなると、勾配や Hesse 行列の計算機による評価自体のコストが高くなる \Rightarrow 目的関数について高次の微分情報を要求するアルゴリズムが実用的に効率的かどうかは不明。

余談 反復法と進化ダイナミクス

反復法は**進化ゲーム理論**とも関係がある。講義後半では進化ゲーム理論との関係についても触れる。

- 進化ゲームでは、エージェントが戦略を改訂する際に戦略分布 x の時間変化を**進化ダイナミクス** (evolutionary dynamics) でモデル化する。特に離散時間のモデルはある種の反復法と見なすことができる。
- 例えば、 $BR(x)$ を戦略分布 x に対する混合最適応答とすれば、最適応答動学 (best-response dynamic) は

$$x^{k+1} \in BR(x^k)$$

で与えられ、その Nash 均衡への収束性が問題になる。

無制約最適化問題に対する反復法

無制約最適化問題

制約付き最適化問題においては、 d^k をどのように決めるかも非自明.

ここでは単純な反復法の例を導入するため無制約最適化問題を考える：

$$\text{minimize: } f(x) \quad \text{subject to: } x \in \mathbb{R}^n$$

まず**最適性条件**を知り，次の2つのアルゴリズムを学ぶ：

- 最急降下法
- Newton 法

※誤解のない限り $f^k \equiv f(x^k)$, $\nabla f^k \equiv \nabla f(x^k)$, $\nabla^2 f^k \equiv \nabla^2 f(x^k)$

無制約最適化問題の最適性条件

定理 **1次の必要条件** (1st-order necessary condition) : f が1回微分可能とする。もし $\bar{x} \in \mathbb{R}^n$ が無制約最適化問題の局所最適解ならば、次の条件を満足しなければならない。

$$\nabla f(\bar{x}) = \mathbf{0}.$$

このような \bar{x} を**停留点** (stationary point) と呼ぶ。

- 必要条件 : $\nabla f(\bar{x}) = \mathbf{0} \longleftarrow \bar{x}$ が局所最適解。

Quiz 1変数の場合に、十分性 sufficiency が成り立たない場合 (停留点であるが局所最適解ではない場合) を図示せよ。

無制約最適化問題の最適性条件

定理 **2次の必要条件** (2nd-order necessary condition). f が2回微分可能とする. もし $\bar{x} \in \mathbb{R}^n$ が無制約最適化問題の局所最適解ならば, $\nabla^2 f(\bar{x})$ は半正定値である.

- 行列 A が半正定値: $\delta^\top A \delta \geq 0$ ($\forall \delta \neq 0$)
- 必要条件: $\nabla^2 f(\bar{x})$ が半正定値 $\longleftarrow \bar{x}$ が局所最適解.

Quiz 1変数の場合に, 十分性が成り立たない場合を図示せよ.

無制約最適化問題の最適性条件

定理 **2次の十分条件** (2nd-order sufficient condition). f が2回微分可能とする. もし $\bar{x} \in \mathbb{R}^n$ において $\nabla f(\bar{x}) = \mathbf{0}$ かつ $\nabla^2 f(\bar{x})$ が**正定値**ならば, 無制約最適化問題の局所最適解である.

- 行列 A が正定値: $\delta^\top A \delta > 0$ ($\forall \delta \neq 0$)
- 十分条件: $\implies \bar{x}$ が局所最適解.

Quiz 1変数の場合に, 必要性 necessity が成り立たない場合を図示せよ.

以上の最適性条件は, 後に講義する一般の非線形最適化問題に対する最適性条件に特殊ケースとして含まれる.

最急降下法 (steepest descent/gradient descent)

最急降下法の具体的なアルゴリズムは次の通り。

Step 0. 初期化: $x^0 \in \mathbb{R}^n$, $k \leftarrow 0$

Step 1. 停止条件を満足するならば, x^k を数値解として出力して停止.

Step 2. $d^k \leftarrow -\nabla f^k$

Step 3. $\alpha^k \leftarrow \arg \min_{\alpha \geq 0} \phi(\alpha) = f(x^k + \alpha d^k)$

Step 4. $x^{k+1} \leftarrow x^k + \alpha^k d^k$

Step 5. $k \leftarrow k + 1$ として Step 1 \wedge .

- 収束条件としては $|f^{k+1} - f^k| < \delta$ などを考える.
- 実用上は, Step 3 における**直線探索** (line search) は厳密に行わず,
バックトラッキング (backtracking; Armijo's rule) により満足できる程度に関数値を下げる α^k を選ぶ.

理論分析可能な例：凸二次計画問題

(出典：)

次の凸二次関数の無制約最小化問題を考える． Q は $n \times n$ 正定値対称行列．

$$f(x) = \frac{1}{2} (x - \bar{x})^\top Q (x - \bar{x})$$

Quiz $x^k \neq \bar{x}$ を考える．

1. 点 x^k における最急降下方向 d^k を求めよ．
2. 点 x^k における最適ステップ幅 α^k を求めよ．
3. $f(x^{k+1})$ を， $f(x^k)$ および Q の固有値を用いて評価せよ．
4. 3からわかるように，この問題の最急降下法の効率 は行列 Q の **条件数** $\text{cond}(Q) \equiv \lambda_{\max}/\lambda_{\min} \geq 1$ に依存する．その図的意味を考えてみよ．

Quiz 解答

1. $d^k = -\nabla f(x^k)$. ただし, ∇f は, $Q = Q^\top$ であることに注意して

$$\nabla f(x) = \frac{1}{2}Q(x - \bar{x}) + \frac{1}{2}Q^\top(x - \bar{x}) = Q^\top(x - \bar{x})$$

2. $f(x^k + \alpha^k d^k)$ の α についての停留条件を解く. 簡単のため k を省略.

$$\begin{aligned} f(x + \alpha d) &= \frac{1}{2}(x - \bar{x} + \alpha d)^\top Q(x - \bar{x} + \alpha d) \\ &= \frac{1}{2}(x - \bar{x})^\top Q(x - \bar{x}) + \alpha d^\top Q(x - \bar{x}) + \frac{\alpha^2}{2}d^\top Qd \\ &= f(x) - \alpha d^\top d + \frac{\alpha^2}{2}d^\top Qd \end{aligned}$$

よって α についての停留条件は

$$\frac{df(x + \alpha d)}{d\alpha} = -d^\top d + \alpha d^\top Q d = 0$$

よって最適ステップサイズは

$$\alpha^k = \frac{d^\top d}{d^\top Q d} = \frac{\|d\|^2}{d^\top Q d}$$

3. まず, $f(x^k)$ は, Q が対称で正則なら $Q^{-1} = Q^{-\top}$ に注意して

$$f(x) = \frac{1}{2} (Q(x - \bar{x}))^\top Q^{-1} Q(x - \bar{x}) = \frac{1}{2} d^\top Q^{-1} d$$

最適ステップサイズのもとでは

$$\begin{aligned} f^{k+1} &= f^k - \frac{\|d\|^2}{d^\top Q d} \|d\|^2 + \frac{1}{2} \frac{\|d\|^4}{(d^\top Q d)^2} d^\top Q d = f^k - \frac{1}{2} \frac{\|d\|^4}{d^\top Q d} \\ &= \left(1 - \frac{1}{2f^k} \frac{\|d\|^4}{d^\top Q d} \right) f^k = \left(1 - \frac{\|d\|^4}{(d^\top Q d)(d^\top Q^{-1} d)} \right) f^k \end{aligned}$$

ここで、 Q の固有値を $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n > 0$ とすると

$$\lambda_n \|d\|^2 \leq d^\top Q d \leq \lambda_1 \|d\|^2, \quad \lambda_1^{-1} \|d\|^2 \leq d^\top Q^{-1} d \leq \lambda_n^{-1} \|d\|^2.$$

$$f^{k+1} \leq \left(1 - \frac{1}{\lambda_1 / \lambda_n} \right) f^k$$

条件数 $\kappa(A) \equiv \lambda_1 / \lambda_n \geq 1$ が 1 に近いほど収束し易いことがわかる。

補足 条件数 (condition number)

条件数：数値解析で問題の感度を表す指標。条件数が大きいほど、問題は数値的に取り扱いづらく、入力データの摂動が解に大きく影響を与える。

例 線形方程式 $Ax = b$ の場合、非特異な正規行列 A の条件数 $\kappa(A)$ は $\kappa(A) = |\lambda_{\max}| / |\lambda_{\min}|$ ($\lambda_{\max}, \lambda_{\min}$: A の最大・最小固有値)

条件数が小（良条件; well-conditioned）なら数値計算は安定しやすい。

条件数が大（悪条件; ill-conditioned）なら不安定になりやすい。

※ A が特異である場合は条件数 = 無限大

例 線形方程式系 $Ax = b$ について A の条件数は、データ b が変化すると解がどの程度変化するかを表現。 A が特異なとき、 $Ax = b$ の解は一意的ではなかった。条件数 A がどれだけ“特異に近い”かを示す。

Newton 法

最急降下法は目的関数を x^k において線形近似して降下方向を決定した。

Newton 法では，二次近似によって降下方向を決定する。

f の x^k における二次近似は

$$f(x^k + d) \simeq f_k^Q(d) \equiv f(x^k) + \langle \nabla f^k, d \rangle + \frac{1}{2} d^\top \nabla^2 f^k d.$$

f_k^Q の停留条件は $\nabla f_k^Q(d) = \mathbf{0}$ であるから， **f^Q が凸であれば**

$$\nabla f_k^Q(d) = \nabla f^k + \nabla^2 f^k d = \mathbf{0}$$

の解を $d = d^k$ とすれば $x^{k+1} = x^k + d^k$ で f_k^Q が最小。

Newton 法

Newton 法の具体的なアルゴリズムは次の通り。

Step 0. 初期化： $x^0 \in \mathbb{R}^n$, $k \leftarrow 0$

Step 1. 停止条件を満足するならば, x^k を数値解として出力して停止.

Step 2. $d^k \leftarrow - \left(\nabla^2 f^k \right)^{-1} \nabla f^k$

Step 3. $\alpha^k \leftarrow 1$

Step 4. $x^{k+1} \leftarrow x^k + \alpha^k d^k$

Step 5. $k \leftarrow k + 1$ として Step 1 へ.

Quiz 探索方向 d^k が降下方向となるための十分条件を述べよ. なお, 降下方向とは $\langle \nabla f^k, d \rangle < 0$ を満足する d だった.

Quiz 目的関数の形状によっては目的関数値が解更新で改悪する場合, 最終的に解が振動し収束しない場合がある. それぞれ1変数の場合で図示せよ.

Newton法の性質

- 局所最適解の近傍，すなわち関数がある意味「二次関数っぽい」範囲では非常に収束が速い．目的関数が二次なら $f = f^Q$ ゆえ1回で収束．
- 二次近似のよさに決定的に依存．**局所的収束性** (local convergence) は保証されるものの，**大域的収束性** (global convergence) はない．

例：振動・目的関数値の悪化など

- 大域的収束性：任意の初期点から局所最適解へ収束する．
最急降下法は大域収束性が保証されている．
- 局所的収束性：局所最適解 \bar{x} に十分近い初期点から \bar{x} へ収束する．
→ 最急降下法とNewton法系の手法を組み合わせることがある．
- 次元が大きな問題では，Step 2 のHesse行列の評価と線形方程式の求解のコストが高い．Hesse行列の計算を回避し，効率的に最適化を行う手法として**準Newton法** (quasi-Newton method) がある．

制約つき凸最適化問題に対する反復法

許容方向降下法 (feasible direction method)

制約付き最適化問題では、 d^k を自由に選ぶとは限らない。

例えば、最急降下方向 $d^k = -\nabla f^k$ が実行可能領域 S の外側に向かってしまう場合、その方向に降下することはできず、最急降下法は使えない。

各点で、許容領域内であり、かつ目的関数値を減少させる方向 d を選択する方針が考えられる。このような反復法を**許容方向降下法**と呼ぶ。

Frank–Wolfe 法

Frank–Wolfe 法は，凸計画問題に対する古典的な許容方向降下法である．

x^k において最適化問題を解き d^k を決定する．

$$y^k \leftarrow \arg \min_{y \in S} \langle \nabla f(x^k), y \rangle$$

$$d^k \leftarrow y^k - x^k$$

- 各点において目的関数の線形近似を最小化する点の方向へ移動する．
- S を表現する制約関数が全て線形であれば， y^k を決定する問題は線形最適化問題になる．Frank–Wolfe 法は，この線形最適化問題が高速に解けるならば非常に有効．→ 講義後半の congestion game で確認する．

Quiz このようにして得られる方向 d^k が降下方向であることを確認せよ．



- 制約あり凸最適化問題の非常に重要な問題クラスである**線形最適化問題** (linear optimization problem) を学ぶ.