最適化理論

http://www.cs.miyazaki-u.ac.jp/~date/lectures/optimization/

伊達 章

宮崎大学 工学部 情報システム工学科

2017 年 6 月 9 日

講義のスケジュール（案）

1. 講義の概要
2. 数学的準備：曲線と曲面
3. 数学的準備：1 次形式と 2 次形式
4. 数学的準備：2 次形式の標準形
5. 関数の極値： 関数の勾配と等高線，関数の極値
6. 関数の極値：ラグランジュの未定乗数法
9. 統計的最適化：正規分布，最尤推定
10. 動的計画法 （その 1）
11. 動的計画法 （その 2）
7. 関数の最適化：勾配法・ニュートン法
8. 関数の最適化：共役勾配法
12. 最小二乗法：連立一次方程式，特異値分解と一般化逆行列
13. 最小二乗法（その 2）
14. まとめ
15. 定期試験，解説

目標

- 最適化手法の原理と計算法を学ぶ
→ 数理最適化ソルバー（ソフトウェア）を使いこなせるようになる
- 最適化理論に必要な数学
  - 曲線と曲面の方程式，法線ベクトル
  - 2 次形式の最大・最小化
- 関数の極値
  - ヘッセ行列やラグランジュの未定乗数法の意味
  - 最適化（勾配法など）
- 統計的最適化： 最尤推定
- 動的計画法
- 線形計画法の基本的な手法

基本知識（確率・統計の復習）

- 平均 μ，分散 σ^2，標準偏差 σ
- 確率分布：一様分布，正規分布
- 擬似乱数の生成
- 最尤推定
- 同時確率，条件付き確率
- マルコフ的情報源
- ベイズの公式，事前確率・事後確率
- 事後確率最大化

- 動的計画法（第８章）

基本知識（確率・統計の復習）

- 平均 μ，分散 σ^2，標準偏差 σ
- 確率分布：一様分布，正規分布
- 擬似乱数の生成
- 最尤推定
- 同時確率，条件付き確率
- マルコフ的情報源
- ベイズの公式，事前確率・事後確率
- 事後確率最大化

- 動的計画法（第８章）

平均，分散

- 平均 μ，期待値 $\mathsf{E}[x]$

$$\mu = \mathsf{E}[x] = \sum_{i=1}^{n} x_i p(x_i), \quad \int_{-\infty}^{\infty} x p(x) dx$$

平均，分散

- 平均 μ，期待値 $\mathsf{E}[x]$

$$\mu = \mathsf{E}[x] = \sum_{i=1}^{n} x_i p(x_i), \quad \int_{-\infty}^{\infty} x p(x) dx$$

平均は分かった．例：数学のテストの平均 70 点
その周りにどの程度ばらついているかも知りたい！

平均，分散

- 平均 μ，期待値 $\mathsf{E}[x]$

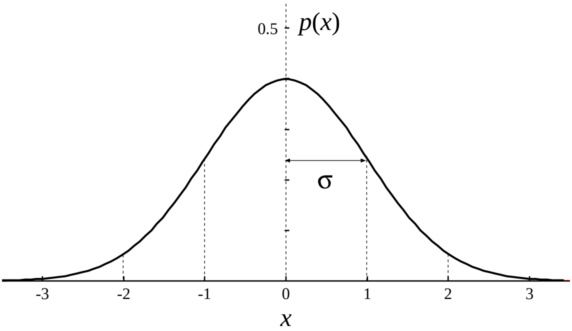
$$\mu = \mathsf{E}[x] = \sum_{i=1}^{n} x_i p(x_i), \quad \int_{-\infty}^{\infty} xp(x)dx$$

平均は分かった．例：数学のテストの平均 70 点
その周りにどの程度ばらついているかも知りたい！

- 分散 σ^2，標準偏差 σ ： 【一つの指標】

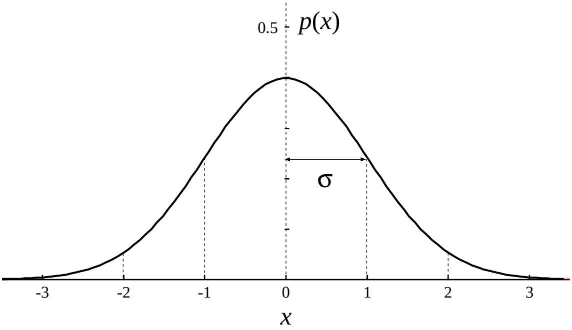
$$\sigma^2 = \mathsf{E}[(x-\mu)^2]$$

正規分布，ガウス分布



「$x_1, x_2, \cdots .x_{100}$ を平均 μ, 分散 σ^2 の互いに独立なガウス分布に従う確率変数とする」

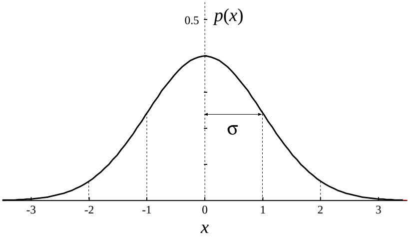
正規分布，ガウス分布 $\mathcal{N}(\mu, \sigma^2)$



$$p(x;\theta,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\mathrm{e}^{-\frac{(x-\mu)^2}{2\sigma^2}},\ p(x;0,1) = \frac{1}{\sqrt{2\pi}}\mathrm{e}^{-\frac{x^2}{2}}$$

$$\int_{-\infty}^{\infty} p(x)dx = 1$$

正規分布，ガウス分布

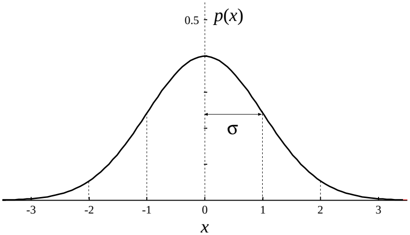


$$p(x;0,1) = \frac{1}{\sqrt{2\pi}}\mathrm{e}^{-\frac{x^2}{2}}$$

$x_i, i = 1, \cdots, 1000$ のうち約 68.26% が $-1 < x_i < 1$ に含まれている．その根拠：

$$\int_{-1}^{1} p(x)dx = 0.6826$$

正規分布，ガウス分布



$$\int_{-2}^{2} p(x)dx = 0.9544, \quad \int_{-3}^{3} p(x)dx = 0.9974$$

基本知識（確率・統計の復習）

- 平均 μ，分散 σ^2，標準偏差 σ
- 確率分布： 一様分布，正規分布
- 擬似乱数の生成
- 最尤推定
- 同時確率，条件付き確率
- マルコフ的情報源
- ベイズの公式，事前確率・事後確率
- 事後確率最大化

- 動的計画法（第８章）

ss002.py

```
import random

T = 200;
Sigma = 0.7
random.seed(20131107)

for i in range(T):
    print random.gauss(0,Sigma)
```

- 平均 $\mu = 0$, 標準偏差 $\sigma = 0.7$ の正規分布にしたがうデータを 200 個生成
- 正規分布（＝ Gauss 分布）とは？

擬似乱数

- おなじない：import random という行が必要
- 一様分布

```
for i in range(100):
    print random.randint (2,9)
# 2 から 9 までの整数が等確率で出力される
```

- 正規分布（ガウス分布）

```
for i in range(100):
    print random.gauss(72.0, 5.0)
```

\# 平均 μ=72，標準偏差 σ=5 の正規分布にしたがうデータが出力される.

擬似乱数

- おなじない：`import random` という行が必要
- 一様分布

```
for i in range(100):
    print random.randint (2,9)
# 2 から 9 までの整数が等確率で出力される
```

- 正規分布（ガウス分布）

```
for i in range(100):
    print random.gauss(72.0, 5.0)
```

\# 平均 μ=72，標準偏差 σ=5 の正規分布にしたがうデータが出力される.

- 標準偏差 σ の意味？！

擬似乱数

- おなじない：`import random` という行が必要
- 一様分布

```
for i in range(100):
    print random.randint (2,9)
# 2 から 9 までの整数が等確率で出力される
```

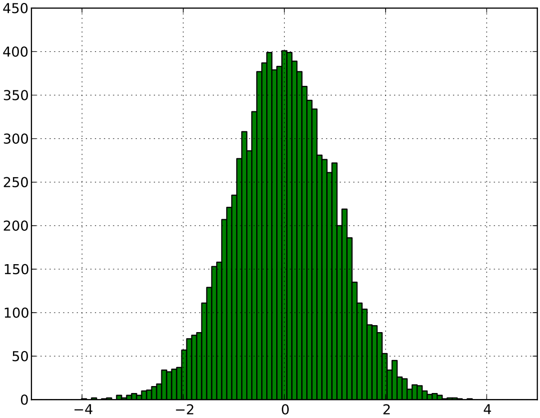
- 正規分布（ガウス分布）

```
for i in range(100):
    print random.gauss(72.0, 5.0)
```

\# 平均 μ=72，標準偏差 σ=5 の正規分布にしたがうデータが出力される.

- 標準偏差 σ の意味？！
- `random.seed(20131107)` とは？

正規分布（ガウス分布） ss105.py



- $\mathcal{N}(\mu, \sigma^2) = \mathcal{N}(0, 1)$ にしたがう１万のデータ
- $[-1 : 1]$ にあるデータは何%？　$[-3 : 3]$ は？

最尤推定

- 確率分布の形を仮定： $x \sim p(x; \boldsymbol{\theta})$,
- 目的：パラメータ $\boldsymbol{\theta}$ の値を知りたい！
  例：
  $$p(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \mathrm{e}^{-\frac{(x-\mu)^2}{2\sigma^2}}$$
- 与えられているデータ：$\boldsymbol{x} = (x_1, x_2, \cdots, x_N)$
- アイデア： データ $\boldsymbol{x}$ を固定して
  $$l(\boldsymbol{\theta}) = f(\boldsymbol{x}, \boldsymbol{\theta}) = \prod_{i=1}^{N} p(x_i; \boldsymbol{\theta})$$
  を最大にする $\boldsymbol{\theta}$ の値を真の $\boldsymbol{\theta}$ の推定値にしよう！
  $$\hat{\boldsymbol{\theta}} = \operatorname*{argmax}_{\boldsymbol{\theta}} l(\boldsymbol{\theta})$$
- 用語： 尤度（ゆうど）関数 $l(\boldsymbol{\theta})$, 最尤推定量 $\hat{\boldsymbol{\theta}}$

対数尤度

- アイデア： データ $\boldsymbol{x}$ を固定して

$$l(\boldsymbol{\theta}) = f(\boldsymbol{x}, \boldsymbol{\theta}) = \prod_{i=1}^{N} p(x_i; \boldsymbol{\theta})$$

  を最大にする $\boldsymbol{\theta}$ の値を真の $\boldsymbol{\theta}$ の推定値にしよう！

$$\hat{\boldsymbol{\theta}} = \operatorname*{argmax}_{\boldsymbol{\theta}} l(\boldsymbol{\theta})$$

- 対数関数は単調増加関数． $L(\boldsymbol{\theta}) = \log l(\theta)$ として

$$\hat{\boldsymbol{\theta}} = \operatorname*{argmax}_{\boldsymbol{\theta}} L(\boldsymbol{\theta})$$

  を求めても同じ．

- 用語： 対数尤度 $L(\boldsymbol{\theta})$， 最尤推定量 $\hat{\boldsymbol{\theta}}$

問題：最尤推定 ***

- n 個のデータ $x_1, x_2, \cdots, x_n$ を観測した.
- このデータは，互いに独立であると仮定する.
- 正規分布 $\mathcal{N}(\mu, 1)$ から生成されたと仮定する.

$$x_i \sim \mathcal{N}(\mu, 1), \ i = 1, \cdots, n$$

- このとき，μ の最尤推定量 $\hat{\mu}$ を求めよ.

問題：最尤推定 ***

・n 個のデータ $x_1, x_2, \cdots, x_n$ を観測した.
・このデータは，互いに独立であると仮定する.
・指数分布

$$p(x;\lambda) = \frac{1}{\lambda}\mathrm{e}^{-\frac{x}{\lambda}}$$

から生成されたと仮定する（$x \geqq 0$）.

・このとき，μ の最尤推定量 $\hat{\mu}$ を求めよ.

Python チュートリアル

開発環境：PyCharm

- PyCharm で Python プログラミング
  - 起動．立ち上げる
  - File → New → Python File →ファイル名入力→ OK
  - Run → Run （実行したいファイル名を選ぶ）
  - 終了 PyCharm → Quit PyCharm
- 念の為，コマンドプロンプトから ipython を起動しておく．

ss001.py

https://github.com/date333cs/Start_Python

```
1 T = 30;
2 for i in range(T):
3     print "iter␣=␣", i
```

- 3 行目，tab で挿入したスペースは重要
- range とは？

  ↓ コマンドプロンプト から ipython

```
>>> range(10)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

>>> help(range)
```

おまじない

- 日本語文字の取り扱い
  - ファイルの先頭におまじないをつける
    # -*- coding: utf-8 -*-
- from math import *
  log とか sin が使える． 例： log(2), cos(pi)
- # で始まる行はコメント（例外あり）

変数の取り扱い

- 代入

```
a = 3
b = b + 1
c += 1  # c=c+1 と同じ
```

- 配列（のようなもの．リスト型という）

```
a = [3,5,6,1]
a.append(4)   # a=[3,5,6,1,4]
  # ipython で a. の後，tab で候補を表示
print 7 in a  # False ← 7 はあるか？
print a

b = [1.0]*3      # b=[1.0, 1.0, 1.0] と同じ
b[2] = 5.0       # b=[1.0, 1.0, 5.0] となる
```

制御構文

- ループ（繰り返し）

```
for i in range(10):
    print i
    # 0 から 9 までが出力される
    # インデント（空白）が大事
```

- 条件分岐： if then else

```
if a == 0 :    # 条件の時は = が2つ必要!!
    print a
elif a == 3:
    print a*10
else:
    print 5
```

Python チュートリアル
【終わり】

終