アルゴリズム入門　＃8

地引　昌弘

2024.11.28

はじめに

今回は、観測により得られた未知のデータに対して、数理的かつ合理的な因果関係のモデル (例えば微分方程式など) を見い出すことが難しい場合を対象に、数値解析を用いてその傾向などを分析する手法について取り上げます。

1　前回の演習問題の解説

1.1　演習 7-1 — 2 次元配列の生成

これは簡単にコードだけを示します:

```
!pip install ita              # Google Colaboratory へのログイン毎に 1 度実行して下さい。
import ita

def array2new1():             # (a)
  size_i = 5; size_j = 5
  a = ita.array.make2d(size_i, size_j)
  for i in range(size_i):
    for j in range(size_j):
      a[i][j] = j - i
  pprint.pprint(a, width=25)

def array2new2a():            # (b)
  size_i = 5; size_j = 5
  a = ita.array.make2d(size_i, size_j)
  for i in range(size_i):
    for j in range(size_j):
      a[i][j] = i**j
  pprint.pprint(a, width=25)

def array2new2b():            # (b)
  size_i = 5; size_j = 5
  a = [[i**j for j in range(size_j)] for i in range(size_i)]
  pprint.pprint(a, width=25)
```

(b) については、通常のプログラムに加え、教科書の 5.6 節「【発展】配列の様々な機能」で紹介されている**内包表記** (Comprehension) を用いたプログラムも作成してみました。内包表記は、

配列 = [式 for i in range(start, stop, step)]

という形式で使います。配列の各要素には、“式” の結果が代入されます。また、式の部分は、入れ子にした内包表記や枝分かれなどを記述することができます。但し、この例を見ても分かる通り、対象が 1 次元配列であれば、それなりに見通しの良いプログラムと言えますが、対象が 2 次元配列の場合 (要は入れ子になっている場合) や、式に枝分かれなどを入れる場合は、可読性が下がってしまいます。

```
def array2new3():             # (c)
  size_i = 5; size_j = 5
  a = ita.array.make2d(size_i, size_j)
  for i in range(size_i):
    for j in range(size_j):
      if i == j:
        a[i][j] = 1
      else:
        a[i][j] = 0
  pprint.pprint(a, width=25)

def array2new4():             # (d)
  size_i = 5; size_j = 5
  a = ita.array.make2d(size_i, size_j)
  for i in range(size_i):
    for j in range(size_j):
      if abs(i-2) == abs(j-2):
        a[i][j] = 1
      else:
        a[i][j] = 0
  pprint.pprint(a, width=25)
```

(d) については、第 7 回で紹介した絶対値を求める abs 関数を利用しました (これも簡単なクイズという感じですね)。

1.2 演習 7-2 — 行列の加算

次は行列の加算です。今回の加算では、渡される行列の行数 · 列数が事前に分からないので、調べる必要があります。さらに、加算する行列同士の行数 · 列数が合っていないと演算ができないため、これも調べる必要があります。これはコードを示します:

```
!pip install ita              # Google Colaboratory へのログイン毎に 1 度実行して下さい。
import ita
import pprint

def mtrx_add(x, y):
  r = []
  if (len(x) != len(y)): return False            # 行列同士の行数が異なる場合は演算不可
  for i in range(len(x)):
    r.insert(len(r), [])                         # 加算結果の 1 行分を用意
    if (len(x[i]) != len(y[i])): return False    # 同、列数が異なる場合は演算不可
    for j in range(len(x[i])):
      r[i].insert(len(r[i]), x[i][j] + y[i][j])  # 加算結果の 1 要素分を用意し、結果を格納
  return(r)
```

動作結果も載せておきます。

```
>>> x = [[1,2], [3,4]]
>>> y = [[10,20], [30,40]]
>>> r = mtrx_add(x, y)
>>> pprint.pprint(r, width=10)
[[11, 22],
 [33, 44]]
```

```
>>> z = [[1,2], [3,4], [5,6]]     # 加算対象行列の行数が異なる場合は演算不可を返す。
>>> r = mtrx_add(x, z)
>>> pprint.pprint(r, width=10)
False
>>> z = [[1,2,3], [4,5,6]]        # 同、列数が異なる場合は演算不可を返す。
>>> r = mtrx_add(x, z)
>>> pprint.pprint(r, width=10)
False
```

2 未知データの傾向分析

ここに未知のデータ集合 (数値の組の列) があるとして、そこから何かを見い出す場合を考えます。もちろん、複数の事項 (= データの種類)が関係していないと意味のある検討をすることはできません[1]。つまり、具体的に行なうことは、ある事項と別の事項との間にどんな関係があるかを調べることだと言えます。議論を簡潔にするため、取り敢えず関係する事項は二つ (X と Y) だとしましょう (つまり、データは、(X, Y) という組になっているとします)。X, Y については素性が分からないので、取り敢えず何らかの確率に従って生じていると考えることにします[2]。もちろん、X, Y の関係が十分に見えてくれば、微分方程式などによる数理モデルを考えることができますが、まだそこまでには至っていません。このような場合、最初の手掛かりとして、まずは X, Y の関係がどんな傾向を持っているかを調べることになります。以下では、未知のデータが隠し持っている傾向をどのように分析したらよいのか、その手法について考えて行きましょう。

2.1 相関係数

多くの人が最初に思い付く (と言うか、最も簡単な) 関係は、X が増えると Y はどうなるか、ではないでしょうか。つまり、観測されたデータの組 (X, Y) に対して、X が大きい場合は Y も大きいのか、あるいは小さいのかを見てみたいわけです。特に、この関係を数値で表わすことができれば、全く別のデータ集合と定量的な (客観的な事実に基く) 議論をするこができます。この指標を C としましょう。C に望まれる要件を直感的に整理すると、下記のようになるでしょうか (こんな感じだと関係が分かり易いですよね/ 以下、要件 $1 \sim 3$ と呼ぶことにします)。

1. X が大きくなると Y も大きくなる $\rightarrow C$ は正数 & その絶対値は関係の強さに対応
2. X が大きくなると Y は小さくなる $\rightarrow C$ は負数 & その絶対値は関係の強さに対応
3. X の大小と Y の大小に差がない $\rightarrow C$ は 0

以下では、この要件に合った C の定義を考えることにします。まず最初に決める必要があるのは、「大きい or 小さい」の取り扱いでしょう。数学的なイメージとしては、例えば、X が増えるにつれ… となるのですが、現在調べているデータは未知のデータなので、大きい順や小さい順に整理されて並んでいるとは限りません。よって、「大きい or 小さい」を判断できる何らかの基準を決める必要があります。大小を判断できる基準として最初に思い付くのは、平均ですよね (違う人もいるかも知れませんが、私はこれが自然かなと考えます)。そこで、X, Y の平均値を $\bar{X}, \bar{Y}$ で表わし、各 (X, Y) とその平均値との偏差 $(X - \bar{X}, Y - \bar{Y})$ を考えてみることにします。但し、まずは X, Y の関係を大まかに見たいのに、数値が二つ出て来るのでは、かえって混沌としてしまうので、両者を掛けることで一つにしてみましょう。即ち、各 (X, Y) に対して、$(X - \bar{X})(Y - \bar{Y})$ を見るわけです。この式は、その意味を考えてみると、$\bar{X}, \bar{Y}$ を基準値とし、X, Y がこの基準よりどれだけ $\pm$ 方向に離れているかを示す値になっていることが分かります。例えば、X がプラス方向に動いた時 $(X - \bar{X} > 0)$、Y もプラス方向に動けば $(Y - \bar{Y} > 0)$、偏差の積は正になります (要件 1 の場合)。また、X と Y の動く方法が逆の場合は、偏差の積は負になります (要件 2 の場合)。これより、各 (X, Y) に対する偏差の積を全て加えることで、要件 1 の組が多ければ偏差の積の和も大きな正数になり、要件 2 の組が多ければその逆になります。そして、もし要件 $1 \cdot 2$ の組が同じぐらい存在すれば、偏差の積の和は互いに相殺されて 0 前後になります。これは好都合ですね。偏差の積の和は指標 C の候補と言えます。しかし、このままでは、データの組が多いと偏差の積の和もそれに伴い大きくなり、別のデータ集合との議論に使えなくなってしまうため (データ数に差がある場合は公平でない)、データ数で割った平均値を用いることにしましょう。これを、平均を表わす記号 E を用いて、$E[(X - \bar{X})(Y - \bar{Y})]$ と表わすことにします。

[1] 例えば、温度を示す数値が並んでいたとして、それがいつ計測されたのか、どこで計測されたのか等の情報が全て欠落していれば、せいぜい頻度を数えるぐらいしかできないので、有益な検討があまりできないということです。

[2] これは、第 6 回でカオス理論の応用について説明した際にも出て来た考え方ですね。何か法則性があれば、それに基づいたモデルを作れますが、それが見当たらない場合は、無作為に発生していると扱わざるを得ないというわけです

ところで、未知のデータ集合 P, Q があり、そのどちらにも長さに関するデータが含まれていたとします。P にある長さのデータは m 単位で観測され、Q では cm 単位で観測されていた場合、互いの $E[(X-\bar{X})(Y-\bar{Y})]$ を比較すると、Q の方が 100 倍大きくなってしまいます。この問題は、$E[(X-\bar{X})(Y-\bar{Y})]$ を、これと “同じ次数” を持つ X, Y の式で割り算することにより、基本的に解決できます (これは、同じ単位で割り算することにより、単位に関係ない値に変えてしまおうという意味を持っています)。但し、どんな式で割り算してもよいわけではなく、少なくとも次の条件を満たしている必要があります。

・$E[(X-\bar{X})(Y-\bar{Y})]$ と次数は同じだが違う形の式

・X, Y の関係が示す傾向に合った式

ここで、以後の見通しを良くするために、データの総数を n として、$A_i = X_i - \bar{X}$, $B_i = Y_i - \bar{Y}$ と置くことにします。$E[(X-\bar{X})(Y-\bar{Y}])$ は下記のように表わされます。

$$E[(X-\bar{X})(Y-\bar{Y}]) = E[AB] = \frac{1}{n}\sum_{i=1}^{n} A_i B_i$$

さて、$\sum AB$ と同じ次数で形が違う式として、まず思い付くのは $\sum A \sum B$ です。しかし、n が増えるにつれ、$\sum A \sum B$ の方は項数が指数的に増えてしまうので、好ましくありません[3]。これは、$\sum A \sum B$ が多項式同士の掛け算になっているからで、$\sum AB$ の方もこれを用いた多項式同士の掛け算に変形すれば、うまく行くかも知れません。そこで、$\sum AB$ の代わりに $(\sum AB)^2$ とし、$\sum A \sum B$ との関係を調べてみます。都合が良いことに、両者の項数は合っています。次数については、4 次式と 2 次式になっているので、両者の次数を合わせるため、$\sum A \sum B$ の代わりに $\sum A^2 \sum B^2$ としてみます。$(\sum AB)^2$ と $\sum A^2 \sum B^2$ を比べると、両者の次数と項数がうまく合っているので、これは使えそうです。ところで、両者をよく見てみると、これらは、次に示す有名なシュワルツの不等式に出て来る項と同じ形ですね (覚えてる?)。

$$\left(\sum_{i=1}^{n} A_i B_i\right)^2 \le \left(\sum_{i=1}^{n} A_i^2\right)\left(\sum_{i=1}^{n} B_i^2\right)$$

よって、以下では、この不等式を利用しましょう。この式の両辺を n^2 で割ることにより、次の関係が導かれます。

$$\left(\frac{1}{n}\sum_{i=1}^{n} A_i B_i\right)^2 \le \frac{1}{n}\sum_{i=1}^{n} A_i^2 \cdot \frac{1}{n}\sum_{i=1}^{n} B_i^2 \text{ より、} E[AB]^2 \le E[A^2]\cdot E[B^2]$$

上右式の両辺を $E[A^2]\cdot E[B^2]$ で割って、平方根を取ると下記の関係式が得られます。

$$-1 \le \frac{E[AB]}{\sqrt{E[A^2]}\sqrt{E[B^2]}} \le 1 \tag{1}$$

次に、片方のデータだけを k 倍しても、式 (1) の値が変化しないことを確かめてみましょう (一見、強面に見えますが、各項が綺麗な形をしているので、そう難しく考えることはありません)。

$$\frac{E[AB]}{\sqrt{E[A^2]}\sqrt{E[B^2]}} = \left(\sum_{i=1}^{n} A_i B_i\right) \Big/ \left\{\left(\sum_{i=1}^{n} A_i^2\right)^{1/2}\left(\sum_{i=1}^{n} B_i^2\right)^{1/2}\right\}$$

$$= \left(\sum_{i=1}^{n} (X_i-\bar{X})(Y_i-\bar{Y})\right) \Big/ \left\{\left(\sum_{i=1}^{n} (X_i-\bar{X})^2\right)^{1/2}\left(\sum_{i=1}^{n} (Y_i-\bar{Y})^2\right)^{1/2}\right\}$$

上の式へ、X 代わりに kX' を代入してみます ($\bar{X}$ も k 倍されることに注意)。

$$\left(\sum_{i=1}^{n} (kX_i'-k\bar{X}')(Y_i-\bar{Y})\right) \Big/ \left\{\left(\sum_{i=1}^{n} (kX_i'-k\bar{X}')^2\right)^{1/2}\left(\sum_{i=1}^{n} (Y_i-\bar{Y})^2\right)^{1/2}\right\}$$

$$= k\left(\sum_{i=1}^{n} (X_i'-\bar{X}')(Y_i-\bar{Y})\right) \Big/ \left\{k\left(\sum_{i=1}^{n} (X_i'-\bar{X}')^2\right)^{1/2}\left(\sum_{i=1}^{n} (Y_i-\bar{Y})^2\right)^{1/2}\right\}$$

これより、分母/分子の k は約分できるので、これは k 倍する前の式 (1) と同じになります (k 倍する前の式とは、式 (1) に X' を代入した式のことです)。

[3]例えば、$\lim_{n\to\infty} (\sum AB)/(\sum A \sum B) \to 0$ となってしまう可能性がありますね。

以上より、未知のデータ集合 (X, Y) から得られる値 $E[(X-\bar{X})(Y-\bar{Y})]/(\sqrt{E[(X-\bar{X})^2]}\sqrt{E[(Y-\bar{Y})^2]})$ (但し、$\bar{X}$, $\bar{Y}$ は X, Y の平均) は、X, Y の傾向を表わす指標 C として利用できそうです。特に、この値は式 (1) より、観測した単位に関係なく、常に $-1 \leq C \leq 1$ が成り立つので、直感的にも分かり易い指標と言えます。この指標は**相関係数** (Correlation Coefficient) と呼ばれ、未知のデータ集合を取り扱う際の基本的な傾向分析手法に位置付けられています。

2.2 回帰分析の数値解法

相関係数を調べることで、未知のデータ集合が隠し持っている傾向を大雑把に把握することはできるようになりました。次は、もう少し詳細な関係を探る方法について考えてみましょう。つまり、観測されたデータの組 (X, Y) に対して、単に X が大きい場合は Y も大きいのか小さいのかではなく、「どのくらい」大きいのか小さいのかを (即ち定量的な関係を) 見てみたいわけです。未知のデータは、ある膨大なデータ集合 (これを**母集団** (Population) と呼ぶことにします) から観測されたデータなので、上で述べた定量的な関係が分かれば、まだ観測されていないデータの予測も可能になります。これまでは、未知のデータに関係している事項 (= データの種類)を二つに限定して話を進めて来ましたが、ここではより一般的な分析手法を検討したいので、関係している事項は 2 個以上とします (つまり、一回の観測で 2 種類以上のデータが記録される)。以下、未知のデータを観測されたデータと呼ぶことにします。

さて、これら複数個ある事項間の関係を求めるのですが、各事項間で同時に存在できる関係の総数は[4]、例えば次のように考えると膨大な種類となり (考え方も難しいですよね)、また分析結果が正しいかどうかの検証も手間が掛かります。

- 要素数 n の集合 G を考え、G からまず最初に i_1 個の要素を取り出した集合を $G^{(1)}_{i_1}$ とする。$G^{(1)}_{i_1}$ に含まれる要素間の関係について考えたいので、以下では $i_1 \geq 2$ とし、$G^{(1)}_{i_1}$ にある i_1 個の要素間には全て関係が存在すると定義する (以下同じ)。
- G から i_1 個の要素を取り出す組合わせは ${}_nC_{i_1}$ なので、G にある任意の i_1 個の要素間に存在する関係の総数も ${}_nC_{i_1}$ となる。
- 次に、集合 $G \setminus G^{(1)}_{i_1}$ から i_2 個の要素を取り出した集合を $G^{(2)}_{i_2}$ とし ("\" は集合の引き算を意味する)、同様に考えると (重複をなくしたいので、$i_1 \geq i_2$ とする)、G から任意の i_1 個の要素を取り出して残った要素のうち、任意の i_2 個の間に存在する関係の総数は ${}_{n-i_1}C_{i_2}$ となる。
- $G^{(1)}_{i_1}$ と $G^{(2)}_{i_2}$ の両方に含まれる要素はないので ($G^{(1)}_{i_1} \cap G^{(2)}_{i_2} = \phi$)、$G^{(1)}_{i_1}$ 内の関係と $G^{(2)}_{i_2}$ 内の関係は同時に存在できる。よって、G には ${}_nC_{i_1} \cdot {}_{n-i_1}C_{i_2}$ 通りの関係が存在する ($G^{(1)}_{i_1} \cap G^{(2)}_{i_2} = \phi$ より、$G^{(1)}_{i_1}$ の要素は $G^{(2)}_{i_2}$ の要素と関係がないことに注意)。
- 以下、これを繰り返して行くことにより、G に存在する関係の総数は、${}_nC_{i_1} \cdot {}_{n-i_1}C_{i_2} \cdots {}_{n-(i_1+\ldots+i_{m-1})}C_{i_m}$ となる。
- $i_1, i_2, \ldots, i_m$ については、$i_1 + i_2 + \ldots + i_m \leq n$, $i_1 \geq i_2 \geq \ldots \geq i_m \geq 2$ を満たす全ての組合わせが考えられるので[5]、この組み合わせ集合を I で表わすと、要素数 n の集合 G に存在する最終的な関係の総数は下記の通りとなる。

$$\sum_{i_1, i_2, \ldots, i_m \in I} {}_nC_{i_1} \cdot {}_{n-i_1}C_{i_2} \cdots {}_{n-(i_1+\ldots+i_{m-1})}C_{i_m} \tag{2}$$

そこで、通常は、一つの事項とそれ以外の事項との関係だけを考えることにします。関係する事項の数を $n+1$ 個とした場合、つまり、データが $(x_1, x_2, \ldots, x_n, y)$ という組になっている場合、$x_1, x_2, \ldots, x_n$ 間は互いに関係がないと仮定し (これらは互いに独立で、前節で述べた相関は 0)、y と $x_1, x_2, \ldots, x_n$ との関係だけを調べるわけです。以後、y を "目的変数" or "独立変数"、x_i を "説明変数" or "従属変数" と呼ぶことにします。さらに、今回は説明変数が目的変数に及ぼす定量的な影響を見たいので、相関係数を調べる場合と異なり、説明変数は確率的に発生する値ではないと考えます[6]。また、説明変数と目的変数の間に成り立つ定量的な関係 (要は数式) には、様々な可能性が存在しますが、これも上

[4]例えば、事項 x が五つある場合を考えます。もし、事項 x_1 と x_2 が関係するとした場合、これと同時に存在できる関係は、1) x_3 と x_4 が関係し、x_5 はどれとも関係がない、2) x_3 と x_5 が関係し、x_4 はどれとも関係がない、3) x_4 と x_5 が関係し、x_3 はどれとも関係がない、4) x_3, x_4, x_5 が関係する、5) x_3, x_4, x_5 はどれとも関係がないの計 5 種類です。これをあらゆる場合について、数え上げる必要があるというわけです

[5]例えば、$n = 8$ の場合における $(i_1, i_2, \ldots, i_m)$ の組合わせ集合は、{(8), (7), (6,2), (6), (5,3), (5,2), (5), (4,4), (4,3), (4,2,2), (4,2), (4), (3,3,2), (3,3), (3,2,2), (3,2), (3), (2,2,2,2), (2,2,2), (2,2), (2)} です。ここで、$(i_1, i_2, \ldots, i_m) = (3, 2)$ とは、$i_1 = 3$, $i_2 = 2$, それ以外の $i_j = 0$ です。これは、G 内に、互いに関係を持つ 3 要素の組 $G^{(1)}_3$、互いに関係を持つ 2 要素の組 $G^{(2)}_2$、互いに関係を持たない 3 要素の組 $G^{(0)}_3$ が存在することを意味します ($G^{(1)}_3$ の要素は、$G^{(1)}_3$ 内の要素とのみ関係を持ち、$G^{(2)}_2$, $G^{(0)}_3$ の要素とは関係を持ちません/他も同じ)。ややこしいですね。。

[6]その意味は、$x_1, x_2, \ldots, x_n$ が与えられた場合、y がどうなるかを見たいということです。別の言葉で言えば、$x_1, x_2, \ldots, x_n$ が、y にどのような因果を及ぼしているかを知りたいというわけです。

と同様、全ての場合を想定して分析するのは面倒なので、通常は線形の関係にあるとして (ある意味割り切って) 分析をします。以上より、説明変数と目的変数の間に成り立つ数式を以下ように仮定し、観測されたデータより各係数 a_k を数値的に求めようというわけです。

$$y = a_0 + a_1x_1 + a_2x_2 + \cdots + a_nx_n \tag{3}$$

このような分析手法を**線形回帰分析** (Liner Regression Analysis) と呼びます[7]。

次に、観測されたデータと式 (3) との関係について考えます。本来ならば、観測されたデータは式 (3) を満たしているはずですが、実際には誤差 ϵ を含んでいます。i 回目に観測されたデータを $(x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)}, y^{(i)})$ とすると、これらの関係は以下の式で表わされます。

$$y^{(i)} = a_0 + a_1x_1^{(i)} + a_2x_2^{(i)} + \cdots + a_nx_n^{(i)} + \epsilon^{(i)} \tag{4}$$

ここで、誤差 ϵ について少し考えてみると、説明変数と目的変数の間に式 (3) の関係が成り立ち、観測が正しく行なわれるならば、誤差も小さくなるはずと予想されます[8]。これより、各観測データとの乖離 (かいり) が一番小さくなるような式 (3) が、説明変数と目的変数の定量的な関係を一番精度良く表わしていると想定されます。よって今回は、式 (4) より、全 $\epsilon^{(i)}$ の二乗和を最小にする $a_0, a_1, a_2, \ldots, a_n$ を求めることにします[9]。これを**最小二乗法** Least Squares Method と呼びます。

まずは、式 (4) を以下のように変形し、

$$\epsilon^{(i)} = y^{(i)} - (a_0 - a_1x_1^{(i)} - a_2x_2^{(i)} + \cdots + a_nx_n^{(i)}) \tag{5}$$

各観測データ $(x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)}, y^{(i)})$ $(1 \leq i \leq m)$ を与えた際に、式 (5) 右辺の二乗和 (式 (6)) を最小にする係数 $a_0, a_1, a_2, \ldots, a_n$ を求めます。

$$\sum_{i=1}^{m} \left\{ y^{(i)} - (a_0 + a_1x_1^{(i)} + a_2x_2^{(i)} + \cdots + a_nx_n^{(i)}) \right\}^2 \tag{6}$$

式 (6) は一見、強面に見えますが、よく見てみると各 a についてはただの 2 次式になっているのが分かります。そこで、a_k に注目して展開してみると、下記のようになります。以下では、a_0 を他の項と整合させて式を簡略化するため、取り敢えず $a_0 = a_0x_0^{(i)}$ と置いています ($x_0^{(i)}$ は常に 1 です)。

$$\begin{aligned}
&\sum_{i=1}^{m} \left\{ \left(y^{(i)} - \sum_{j=0, j\neq k}^{n} a_jx_j^{(i)} \right) - a_kx_k^{(i)} \right\}^2 \\
&= \sum_{i=1}^{m} \left\{ (a_kx_k^{(i)})^2 - 2\left(y^{(i)} - \sum_{j=0, j\neq k}^{n} a_jx_j^{(i)} \right) a_kx_k^{(i)} + \left(y^{(i)} - \sum_{j=0, j\neq k}^{n} a_jx_j^{(i)} \right)^2 \right\} \\
&= \left\{ \sum_{i=1}^{m} \left(x_k^{(i)} \right)^2 \right\} a_k^2 - 2\left\{ \sum_{i=1}^{m} \left(y^{(i)} - \sum_{j=0, j\neq k}^{n} a_jx_j^{(i)} \right) x_k^{(i)} \right\} a_k + C
\end{aligned} \tag{7}$$

式 (7) にある a_k^2 の係数は必ず正数になるので、これは下に凸の 2 次曲線となり、1 階微分係数が 0 となる部分で最小値を取ります。よって、全 $\epsilon^{(i)}$ の二乗和を最小にする $a_0, a_1, a_2, \ldots, a_n$ は、下記の 1 次式を満たします。

$$\left\{ \sum_{i=1}^{m} \left(x_k^{(i)} \right)^2 \right\} a_k - \sum_{i=1}^{m} \left(y^{(i)} - \sum_{j=0, j\neq k}^{n} a_jx_j^{(i)} \right) x_k^{(i)} = 0 \tag{8}$$

後は、a_k 以外の各 a についても同様な 1 次式を作成し、これらを連立 1 次方程式として解くことにより、最終的に式 (3) の係数 $a_0, a_1, a_2, \ldots, a_n$ を求めます。とは言え、やはりこれでも計算は大変そうですよね。参考のため、$n = 1$ の時 (関

---

[7] “回帰” という単語は、少々特異な感じがしますね。統計学の世界では、ある時刻に母集団から適当な対象を選んで観測した値が、(何らかの偶然により本来の値に比べて) 偏っていたとしても、別の時刻に同じ対象を選んで観測すると、その値の平均は 1 回目より全体の平均値に近くなることが知られています (例えば、100m 走の時間を考えてみると、コンディションが良い時はベスト タイムが出ますが、常にコンディションが良いわけではなく、何度も計測すれば、その走者が持つ本来の値に近付いて行くというわけです)。これを “平均への回帰” と呼びますが、回帰分析も、観測を重ねることで真の関係に近付いて行くという意味で、これを由来としています。

[8] この場合、誤差は様々な要因が無作為に作用して発生することになります。無作為なので、誤差の発生は何らかの確率に従います。

[9] 直感的には、各観測値と式 (3) との平均距離 (の絶対値) が最小になる場合だと考えられるので、イメージは分かり易いですね。しかし、(教科書でも記述されていますが) このイメージだけでは、説明変数と目的変数の定量的な関係を一番精度良く表わせている理由として、少々不足しています。詳細な理由については付録 1 で説明しているので、是非読んでみて下さい。

係する事項が y と x_1 しかない時、つまり式 (3) が 2 次平面上の線形式になる時) の解を以下に示しておきます。これも一見、強面に見えますが、実は連立 2 元 1 次方程式を解くだけなので、このくらいであればそれほど面倒ではありません。

$$\begin{cases} \left\{\sum_{i=1}^{m}\left(x_0^{(i)}\right)^2\right\}a_0 + \sum_{i=1}^{m}\left(x_0^{(i)}x_1^{(i)}\right)a_1 &= \sum_{i=1}^{m}x_0^{(i)}y^{(i)} \\ \left\{\sum_{i=1}^{m}\left(x_1^{(i)}\right)^2\right\}a_1 + \sum_{i=1}^{m}\left(x_0^{(i)}x_1^{(i)}\right)a_0 &= \sum_{i=1}^{m}x_1^{(i)}y^{(i)} \end{cases} \tag{9}$$

より、

$$a_0 = \frac{\sum\left(x_1^{(i)}\right)^2\sum y^{(i)} - \sum x_1^{(i)}y^{(i)}\sum x_1^{(i)}}{m\sum\left(x_1^{(i)}\right)^2 - \left(\sum x_1^{(i)}\right)^2}$$

$$a_1 = \frac{m\sum x_1^{(i)}y^{(i)} - \sum x_1^{(i)}\sum y^{(i)}}{m\sum\left(x_1^{(i)}\right)^2 - \left(\sum x_1^{(i)}\right)^2}$$

となります。但し、$x_0^{(i)} = 1$ および $\left\{\sum(x_0^{(i)})^2\right\} = m$ に注意して下さい。関係する事項が三つ以上ある場合は、3 章で説明する連立 1 次方程式を数値的に解く方法が実用的です。

2.3 仮説検定

前節で説明した回帰分析は、観測された未知データの組 (X, Y) に対するシンプルな定量的関係を求めました。そこでは、観測誤差が正規分布に従うと仮定し、観測誤差に偏りがなく最も自然に表わせるような $a_0, a_1, a_2, \ldots, a_n$ を求めることが鍵でした (付録 1 参照)。但し、ここで検討したことは、あくまで観測誤差が最も自然に表わせることであり、(X, Y) の値が、真の分布に対してたまたま偏って観測されてしまったかどうかの判断は、また別の話になります。つまり、観測データの集計により得られた結果が、1) 偶然に観測されたもの、2) 観測対象が備える傾向のどちらであるかは、別途検証する必要があるわけです。これを**仮説検定** (Hypothesis Testing) と呼びます。

仮説検定は、主に次の手順からなります。

1. 「観測された結果は単なる偶然である[10]」と仮定する。この仮定を**帰無仮説** (Null Hypothesis) と呼ぶ。また、帰無仮説に対立する仮説を、**対立仮説** (Alternative Hypothesis) と呼ぶ。対立仮説は、帰無仮説が棄却 (or 否定) された場合に採択される[11]。

2. 帰無仮説を前提に、「観測された結果および、さらに極端な結果」が生じる確率を計算する[12]。この確率を **p 値** (有意確率 — P-Value) と呼ぶ。

3. p 値に対し、適当な基準による判定を行なう。この基準を**有意水準** (Significance Level) と呼ぶ。例えば、帰無仮説における p 値は 0.02 であり、通常は発生しないと想定される有意水準 0.05 より小さいため、今回観測された結果については帰無仮説を棄却して対立仮説を採用する、という感じ。

上の手順では、p 値の計算 (確率の計算) が鍵になります。真の p 値を得るには、観測対象が備える真の分布をもとに、今回観測された結果 (+さらに極端な結果) が生じる確率を計算することになります。しかしながら、一般に観測対象が備える真の分布は分かりません (と言うか、そもそもこの分布を知るために観測しているわけですよね)。そこで、窮余の策として、真の分布は正規分布であると想定し[13] (非常事態には非常手段ですよ)、今回観測されたデータが、(真の分布と想定した) 正規分布より偏っている確率を考えることにします。

まずは、観測データの平均値が、正規分布の平均値よりどのくらい離れているか、を考えることから始めてみましょう。これは、観測データの “不偏” 平均値 $\bar{x}$ と正規分布の平均値 μ に対して、$\bar{x} - \mu$ の確率分布を考えることにします (不

[10] つまり、上記 1) を仮定するわけです。こちらを仮定する理由は、観測対象が備える “真の” 傾向が分からない以上、正確な仮説検定を実施するには、考えられる全ての傾向を仮定しなければならず、それは現実的に不可能だからです。

[11] ここでの判断は、“帰無仮説を棄却” → “偶然ではなく、何か傾向があるらしい” → “現在判明している傾向は○○” → “では、それを採用” という流れです。

[12] 教科書の 7.1 節にある p 値の定義「その仮説が間違いであったとしても」とは、「何らかの傾向があると仮定したことが間違いであったとしても」=「観測された結果は、単なる偶然の結果であったとしても」という意味です。

[13] 付録 1 にも書きましたが、自然界や社会における様々な事象の発生分布は、正規分布に従うことが知られており、この仮定はそう的外れなものでもありません。付録 1 には、正規分布が汎用性を示す理由についても記載しています。

偏性の意味については付録 2 で説明しているので、是非読んでみて下さい)。この時、もし観測データの散らばり (分散) が大きいならば、その平均値 $\bar{x}$ も真の平均値 μ から離れている、即ちあまり正しい値ではない可能性が高くなります (分散が小さい場合は、その逆)。そこで、$\bar{x} - \mu$ を観測データの “不偏” 分散 s^2 で割ることにします。但し、観測したデータ数が多ければ、観測データの平均値も相応に正しい値へ近付くはずと考え、分散をそのまま使うのではなく、観測データ数 n で割ったものを使うことにします。つまり、最終的に下記の値の確率分布を考えるわけです (根号の使用は、今後の見通しを良くするため)。この値を **t 値** (T-Value) と呼びます:

$$\frac{\bar{x} - \mu}{\sqrt{s^2 \div n}} = \frac{\bar{x} - \mu}{s/\sqrt{n}} = \frac{\sqrt{n}\,(\bar{x} - \mu)}{s}$$

t 値が従う確率密度関数は解析的に求められており、これを **t 分布** (T-Distribution) と呼びます。t 分布から p 値を計算して仮説検定を行なうことを **t 検定** (T-Test) と呼びます。

仮説検定では一般に、母集団の確率分布が不明な場合 (かつ正規分布と想定しても、大きな乖離がない場合) は、上で説明した t 検定などを実施することになりますが[14]、母集団の確率分布が分かっている場合 (かつ簡単な計算で求められる場合) は、だいぶ見通しが良くなります。以下では、発生確率を簡単に求められる二項分布について考えてみましょう。二項分布に従う事象では、発生確率をどのように捉えるかが鍵となります。仮説検定では、帰無仮説が成り立つかどうか、即ち偶然かどうかを調べるので、何ら作為のない状態で発生する確率を考えます。教科書の 7 章で取り上げた新薬の効果に対する仮説検定では、これは新薬を処方しなくても快復する確率と考えられるので、二項分布に従い偶然に発生する確率は 9/15 となります[15]。

最後に参考のため、回帰分析に対する仮説検定について、少し触れておきます。仮説検定における帰無仮説は、観測された結果は単なる偶然であるという仮定です。よって、これを回帰分析に当てはめると、観測されたデータの組 (X, Y) には定量的な関係がない、つまり $a_0 = a_1 = a_2 = \ldots = a_n = 0$ を意味します。この場合、相関係数も 0 になります。通常、相関係数の計算はあまり難しくないため、回帰分析の仮説検定は相関係数を用いて判断することが多いです。

[14] ここでは、観測データと母集団との平均値の差に着目して t 検定を取り上げましたが、これ以外にも、着目するパラメータに応じて様々な検定方法が存在します (付録 2 参照)。

[15] とは言え、現実の世界では、例えば 9/15 の確率で画一的に分けられるというわけでもないため、このような事象に対する仮説検定でも、結局は平均が 9/15 となる “確率の確率分布” を考える場合が多いです。ややこしいですね。

3　連立 1 次方程式の数値的解法

以下では、連立 1 次方程式を数値的に計算するアルゴリズムを検討します。まずは改めて、数学的な計算手順を確認しておきましょう。次の 3 元連立 1 次方程式について、数学的な手順により解を求めてみます:

$$\begin{aligned} 2x_0 &+ 3x_1 &+ 2x_2 &= 1 \\ 2x_0 &+ 5x_1 &+ 4x_2 &= 4 \\ 4x_0 &+ 8x_1 &+ 8x_2 &= 7 \end{aligned}$$

最初は、1 番上の式を 2 番目、3 番目から引いて x_0 の**係数** (Coefficient) を消去します:

$$\begin{aligned} 2x_0 &+ 3x_1 &+ 2x_2 &= 1 \\ &\quad 2x_1 &+ 2x_2 &= 3 \\ &\quad 2x_1 &+ 4x_2 &= 5 \end{aligned}$$

次は、同様に 2 番目の式を 3 番目から引いて x_1 の係数を消去します:

$$\begin{aligned} 2x_0 &+ 3x_1 &+ 2x_2 &= 1 \\ &\quad 2x_1 &+ 2x_2 &= 3 \\ &&\quad 2x_2 &= 2 \end{aligned}$$

これより $x_2 = 1$ となるので、それを 2 番目、1 番目のに式に代入します。その結果、2 番目の式より $x_1 = 0.5$ が求まるので、それを 1 番目の式に代入します。以上より、最終的に全ての解が求まります:

$$\begin{aligned} x_0 &= -1.25 \\ x_1 &= 0.5 \\ x_2 &= 1 \end{aligned}$$

この手順を一般化して説明すると、まずは式同士を引き算することで順に変数の係数を消去していきます。これを**前進消去** (Forward Elimination) と呼びます。前進消去は、残った変数を並べた形が三角形になるまで行います。その後、三角形の頂点に位置する変数から、逆順に値を代入して行くことで、連立 1 次方程式を解くことができます。これを**後退代入** (Backward Substitution) と呼びます。これは、皆さんも良く知っている連立方程式の解法ですが、この手順は**ガウスの消去法** (Gaussian Elimination) と呼ばれています。

ガウスの消去法に基づく手順をプログラムで扱う場合、変数名は何でもよいので、係数だけを配列のデータとして与えることにします[16]。連立方程式は、式が上から順に並び (行)、変数が左から順に並んでいるので (列)、i 行目の式における j 番目の係数 (i 行 $\cdot$ j 列の係数) を 2 次元配列 `a[i][j]` に対応させます (座標系とは違うので注意しましょう)。この時、配列の添字番号は 0 から始まるため、一番上にある式は 0 番目の式 (or 0 行目の式)、一番左にある変数は x_0 として扱います。また、今回取り上げる連立方程式では、解が存在することを前提に、変数の数と式の数が同じであると想定します。よって、式の数は n 行ですが係数は $n+1$ 個あるので、`a` は `n` × `n+1` 配列です。以下、前進消去では、g 番目の式を用いて各式から変数 x_g を消して行くため `a[..][g]/a[g][g]` の扱いが、後退代入では、変数 x_g の値を各式へ反映して行くため `a[..][g]*a[g][n]` の扱いが、それぞれ鍵となります (添字番号に注目)。そして後退代入を終えた後、最終的な右辺の係数 `a[..][n]` が解 (各変数の値) となります。これらは混乱し易いので、以降のプログラムではしっかりと確認して下さい。

まずは下請けとして、配列 `a[h]`(これは h 行目の式を表わしている) の各要素から、配列 `a[g]` の対応する要素を `m` 倍して引く関数を用意しました:

```
def subvec(a_h, a_g, m):
  for i in range(len(a_h)):
    a_h[i] = a_h[i] - m*a_g[i]
```

配列 `a` は係数の入った 2 次元配列ですが、2 次元配列は 1 次元配列の各要素がさらに 1 次元配列となっているものなので、1 次元配列の引き算は、要素内にある 1 次元配列同士の引き算となります。これを用いると、ガウスの消去法は次のように書くことができます (各 `for` ループの値域 (`range` 関数の引数) に注意して下さい):

[16] 今回は、プログラムの引数として、2 次元配列を直接入力することにします。

```
def gauss1(a):

  # a[..] は一つの式を表わす。n は式の数 (= 2 次元配列の行数) を表わす。
  print("forward:"); print(a); n = len(a)

  # 上の式から順に変数 x_g を消して行く (前進消去)。
  for g in range(n-1):          # 一番上にある式 = 0 行目の式
    for h in range(g+1, n):     # g+1 行目以下の全式から変数 x_g を消す、という意味

      #「h 行目 - (a[h][g]/a[g][g])*g 行目」により、
      # h 行目の x_g と g 行目の x_g の係数を合わせて引き算を行ない、h 行目から x_g を消す。
      subvec(a[h], a[g], a[h][g]/a[g][g])
    print(a)

  # 下の式から逆順に変数 x_g の値を求めて行く (交代代入)。
  # x_g の解は、右辺の定数項 a[g][n] に入る。
  # 交代代入では、以降の計算に必要な係数しか修正していない点に留意
  print("backward:")
  for g in range(n-1, -1, -1):     # x_g を"全式"に代入、次に x_{g-1}を"全式"に代入、という意味

    # x_g の係数で定数項 (右辺) を割ることで、変数 x_g の値 (解) を求める (@1)。
    a[g][n] = a[g][n]/a[g][g]
    for h in range(g-1, -1, -1):  # g-1 行目以上の"全式"へ変数 x_g (だけを) を代入

      #「h 行目の定数項 - a[h][g]*変数 x_g の値」により、h 行目に変数 x_g を"代入した結果を反映"する。
      # ここでは代入結果の反映だけで、解が得られたわけではないことに留意 (各解は上記@1 の割り算で算出)。
      a[h][n] = a[h][n] - a[h][g]*a[g][n]
    print(a)

  # 最終的な解を表示する。
  for g in range(0, n):
    if g == 0: print("(", end="")    # 左端の"("を表示
    print(a[g][n], end="")           # 解を表示
    if g == n-1: print(")")          # 右端の")"を表示
    else: print(", ", end="")        # 解を区切る","を表示
```

各コードの意味と 9 ページにあるガウスの消去法による手順との対応は、コメントを参照して下さい。各行 [[..], [..], [..]] が計算途中の連立方程式一式に該当し、内部の [..] が 1 個の式に該当します。各行毎に、式 [..] の最後 (右端) にある定数項が方程式の解になっています。実行結果は下記のようになります (ここで用いた連立方程式は、9 ページと同じものです):

```
>>> gauss1([[2,3,2,1],[2,5,4,4],[4,8,8,7]])
forward:
[[2, 3, 2, 1], [2, 5, 4, 4], [4, 8, 8, 7]]
[[2, 3, 2, 1], [0.0, 2.0, 2.0, 3.0], [0.0, 2.0, 4.0, 5.0]]
[[2, 3, 2, 1], [0.0, 2.0, 2.0, 3.0], [0.0, 0.0, 2.0, 2.0]]
backward:
[[2, 3, 2, -1.0], [0.0, 2.0, 2.0, 1.0], [0.0, 0.0, 2.0, 1.0]]
[[2, 3, 2, -2.5], [0.0, 2.0, 2.0, 0.5], [0.0, 0.0, 2.0, 1.0]]
[[2, 3, 2, -1.25], [0.0, 2.0, 2.0, 0.5], [0.0, 0.0, 2.0, 1.0]]
(-1.25, 0.5, 1.0)
```

ところで、ガウスの消去法では、処理が前進消去と後退代入の二つに分かれていました。しかし、消去時に「自分より下の行を対象に消去する」代わりに「自分以外の全ての行を対象に消去する」ことで、いきなり解を求めることができます (図 1· 次ページ)。この解法は、**ガウス–ジョルダンの消去法** (Gauss–Jordan Elimination) と呼ばれています。

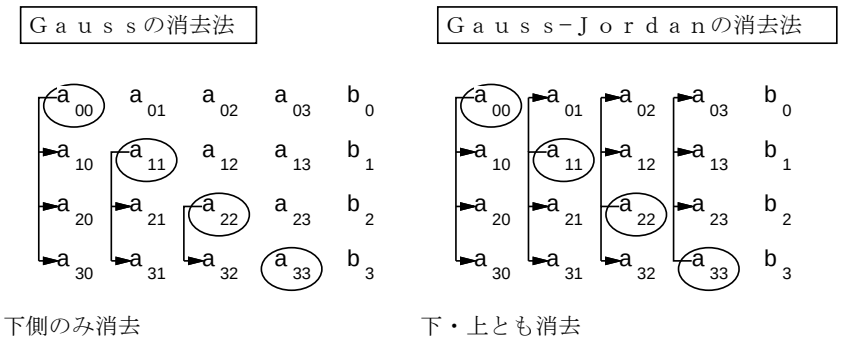

図 1: ガウス-ジョルダンの消去法 (矢印は引き算を意味)

具体的には、まず最初に各行から 0 行目を引き算して 0 行目以外の式から x_0 の項を消去し、次に (0 行目を含む) 各行から 1 行目を引き算して 1 行目以外の式から x_1 の項を消去し $\cdots$ と計算して行きます (つまり、i 行目の式から順次 $x_{j\,(\neq i)}$ の項が消えて行くわけです)。ガウス–ジョルダンの消去法によるプログラムはガウス–ジョルダンの消去法によるプログラムは、下記の通り (subvec 関数は、上と同じです):

```
def gauss2(a):

  # a[..] は一つの式を表わす。n は式の数 (= 2 次元配列の行数) を表わす。
  print("elimination:"); print(a)
  n = len(a)

  # 各式から変数 x_g を消して行く。
  # gauss1 関数では g+1 行目以下の式が対象だが、gauss2 関数では全式が対象となる点に注意
  for g in range(n):
    for h in range(n):

      # g 行目だけ変数 x_g を残す。
      if g != h:
        subvec(a[h], a[g], a[h][g]/a[g][g])
    print(a)

  # 各式から変数 x_g の値を求めて行く。
  print("division:")
  for g in range(n):    # g 行目の式には x_g しか残っていないので、代入 (反映) して行くという処理は不要

    # 変数 x_g の係数で定数項 (右辺) を割ることで、変数 x_g の値を求める。
    a[g][n] = a[g][n]/a[g][g]
    print(a)

  # 最終的な解を表示する。
  for g in range(0, n):
    if g == 0: print("(", end="")     # 左端の"("を表示
    print(a[g][n], end="")            # 解を表示
    if g == n-1: print(")")           # 右端の")"を表示
    else: print(", ", end="")         # 解を区切る","を表示
```

こちらの方がプログラムは簡潔になりますが、計算時間はやや多くなります[17]。

[17]計算量について学んでから、もう一度二つのプログラムを見比べて、両者の計算量を見積もってみて下さい。ヒントは “やや” です。

では、前ページのプログラムを用いて、今度は次の連立方程式を解いてみましょう:

$$\begin{aligned} 2x_0 &+ 3x_1 &+ 2x_2 &= 1 \\ 2x_0 &+ 3x_1 &+ 4x_2 &= 4 \\ 4x_0 &+ 8x_1 &+ 8x_2 &= 7 \end{aligned}$$

実行結果は、次のようになりました:

```
>>> gauss2([[2,3,2,1],[2,3,4,4],[4,8,8,7]])
elimination:
[[2, 3, 2, 1], [2, 3, 4, 4], [4, 8, 8, 7]]
[[2, 3, 2, 1], [0.0, 0.0, 2.0, 3.0], [0.0, 2.0, 4.0, 5.0]]
Traceback (most recent call last):
  File "<pyshell#25>", line 1, in <module>
    gauss2([[2,3,2,1],[2,3,4,4],[4,8,8,7]])
  File "gauss.py", line 164, in gauss2
    subvec(a[h], a[g], a[h][g]/a[g][g])
ZeroDivisionError: float division by zero
```

計算の途中で 0 による割り算をしたというエラーが表示されました。しかし、式を見れば分かりますが、この連立方程式が解けないということはなく、次のようにプログラムへ渡す式の順番を入れ換えれば、問題なく解くことができます:

```
>>> gauss2([[2,3,2,1],[4,8,8,7],[2,3,4,4]])
elimination:
[[2, 3, 2, 1], [4, 8, 8, 7], [2, 3, 4, 4]]
[[2, 3, 2, 1], [0.0, 2.0, 4.0, 5.0], [0.0, 0.0, 2.0, 3.0]]
[[2.0, 0.0, -4.0, -6.5], [0.0, 2.0, 4.0, 5.0], [0.0, 0.0, 2.0, 3.0]]
[[2.0, 0.0, 0.0, -0.5], [0.0, 2.0, 0.0, -1.0], [0.0, 0.0, 2.0, 3.0]]
division:
[[2.0, 0.0, 0.0, -0.25], [0.0, 2.0, 0.0, -1.0], [0.0, 0.0, 2.0, 3.0]]
[[2.0, 0.0, 0.0, -0.25], [0.0, 2.0, 0.0, -0.5], [0.0, 0.0, 2.0, 3.0]]
[[2.0, 0.0, 0.0, -0.25], [0.0, 2.0, 0.0, -0.5], [0.0, 0.0, 2.0, 1.5]]
(-0.25, -0.5, 1.5)
```

この原因は、どこにあるのでしょうか? 先ほどの連立方程式をよく見ると、2 番目の式から 1 番目の式を引いた際に、最初の二つの係数はどちらも 0 になってしまいます。その結果、2 周目 (x_1 以外の全変数項を消去) の処理で、x_0 の係数を消去するため x_1 の係数で割り算を行なうと、0 による割り算が生じてしまい、エラーになるというわけです。これより、次に消去しようとする係数が 0 だったら、別の行 (数式) と入れ換えてから消去を行なう必要があります。これを**ピボット選択** (Pivoting) と呼びます。連立方程式の解を求めるプログラムへピボット選択の機能を追加する場合は、以下の注意を払う必要があります。

- 係数が 0 に近い (絶対値の小さい) 値だと誤差が出易くなるので、ピボット選択では常に絶対値の大きい行を選ぶ必要があります。
- 但し、$2x + 3y = 1$ と $200x + 300y = 100$ とは同じ方程式なので、計算を始めるに際し、まずは各行の係数を絶対値の最大が 1 になるように調整してから、ピボット選択をする必要があります (例えば、前者は 3 で割り、後者は 300 で割る)[18]。

ところで、ピボット選択をしようとしても、全ての係数が 0 になってしまい、適切なピボットを選べない場合は存在するのでしょうか。これは、連立方程式が不定 · 不能の場合に起きます (つまり、解自体が存在しない場合です)。ガウス–ジョルダンの消去法にピボット選択の機能を追加したプログラムは、次のようになります:

[18] この操作を**スケーリング** (Scaling) と呼びます。

```
def selectpivot(a, g, n):
  max = abs(a[g][g])
  k = g

  # a[..][g] は、各式における変数 x_g の係数を表わす。
  for i in range(g+1, n):    # g+1 行目以下を対象とする理由は、解説を参照
    if abs(a[i][g]) > max:
      max = abs(a[i][g])
      k = i
  return(k)                  # 変数 x_g の係数が一番大きい式の番号を返す。


def swap(a, g, k):
  x = a[g]; a[g] = a[k]; a[k] = x


def gauss3(a):

  # a[..] は一つの式を表わす。n は式の数 (= 2 次元配列の行数) を表わす。
  print("elimination:"); print(a); n = len(a)

  idx = ita.array.make1d(n)
  for i in range(0, n):
    idx[i] = i

  # 各式から変数 x_g を消して行く (基本は、ガウス-ジョルダンの消去法と同じ)。
  for g in range(0, n):

    # 変数 x_g の係数 (の絶対値) が最大となる式の番号 k を探す。
    k = selectpivot(a, g, n)
    if abs(a[k][g]) < 0.00000001:    # 絶対値の最大が (ほぼ)0 ならば、不定/不能と判断する。
        return(None)

    # g 行目の式と k 行目の式を入れ替える。idx へ記録を残す (= idx を入れ替えておく) ことに注意
    swap(a, g, k); swap(idx, g, k)

    # g 行目だけ変数 x_g を残して他を消去する。
    for h in range(0, n):
      if g != h:
        subvec(a[h], a[g], a[h][g]/a[g][g])
    print(a, "/", idx)

  # 各式から変数 x_g の値を求めて行く。
  print("division:")
  for g in range(0, n):
    a[g][n] = a[g][n]/a[g][g]

  # 式の順番を元に戻す (idx[i] = i 番目の式が入れ替わった先が入っている)。
  # 式の順番を戻した後は、それに合わせて idx の記録も戻しておく。
  for i in range(0, n):
    if i != idx[i]:
      swap(a, i, idx[i]); swap(idx, i, idx[i])

  # 最終的な解を表示する。
  for g in range(0, n):
    if g == 0: print("(", end="")    # 左端の"("を表示
    print(a[g][n], end="")           # 解を表示
    if g == n-1: print(")")          # 右端の")"を表示
    else: print(", ", end="")        # 解を区切る","を表示

  return(None)
```

以前のプログラムでは、subvec 関数で変数を消去する際、左から順に消去していました。今回は、係数の絶対値が最大となる変数から順に消去していきます。このような変数を選ぶための selectpivot 関数を用意しました。この他に、配列の要素を入れ替える swap 関数も用意しました。selectpivot 関数では、“g 行目以下” の式から変数 x_g の係数が最大となる式を探すことに注意して下さい。その理由ですが、ガウス–ジョルダンの消去法では、変数 x_i の値を求めるために、i 行目の式から変数 x_i だけを残して他の変数を全て消去します (図 1 右 (11 ページ) を確認しましょう)。つまり、0 行目 $\sim$ $g-1$ 行目にある式は、変数 $x_0 \sim x_{g-1}$ を求めるための式として既に使われている(言い換えれば、$i\,(\leq g-1)$ 行目の式は x_i を求める式として位置付けられた) わけです。これより、変数 x_g の値を求めるために利用できる式は、g 行目以下の式になります。また、式の順序を入れ替えると、最後に解を出力する際にどの変数が何番目だったか分からなくなってしまいます。これを解消するため、初期値として番号 $0 \sim n-1$ を順に入れた配列 idx を用意し、式の交換時はこの配列も同じ交換をすることで、元の式が何番目になったかが分かるようにしてあります。結果を表示する際は、idx の記録より、元の並びに直してから表示しています。実行結果も示しておきます:

```
>>> gauss3([[2,3,2,1],[2,3,4,4],[4,8,8,7]])
elimination:
[[2, 3, 2, 1], [2, 3, 4, 4], [4, 8, 8, 7]]
[[4, 8, 8, 7], [0.0, -1.0, 0.0, 0.5], [0.0, -1.0, -2.0, -2.5]] / [2, 1, 0]
[[4.0, 0.0, 8.0, 11.0], [0.0, -1.0, 0.0, 0.5], [0.0, 0.0, -2.0, -3.0]] / [2, 1, 0]
[[4.0, 0.0, 0.0, -1.0], [0.0, -1.0, 0.0, 0.5], [0.0, 0.0, -2.0, -3.0]] / [2, 1, 0]
division:
(1.5, -0.5, -0.25)
```

“/” より右側は、idx 配列の内容 (つまり、何番目の式を何番目の式と入れ替えたか) を示しています。

演習 8-1 相関係数を計算するプログラム (次ページ) を打ち込み、評価用データに対する相関係数を計算せよ (評価用データは、Web ページより取得のこと)。

参考: このプログラムにある init_data 関数の作りは、ファイルからデータを読み込む様々な場面で応用が利くので、是非覚えておいて下さい。また、今回は演習として取り上げませんでしたが、線形回帰分析に出て来る $x_j^{(i)}$ $(1 \leq j \leq n)$ は、このプログラムでは dset[i][j-1] に該当します。$y^{(i)}$ は dset[i][n] に該当し、dset[i][0] は全て 1 です ($x_0^{(i)} = 1$)。よって、式 (8) や式 (9) (6 ページ) にある $\sum$ の計算も、for ループと dset を用いれば、このプログラム例のように実はそう手間を掛けずに計算できます。後は、連立方程式を解くだけですが、これも “ピボット選択を行なうガウス–ジョルダン消去法” のプログラムがあるので、それを利用できます。線形回帰分析は一見、大変そうに見えましたが、これまでに皆さんが作ったプログラムを利用することで、効率的に実現できます。プログラムの再利用は大変重要ですね。

演習 8-2 仮説検定に関連する以下のプログラムを作成し、実行せよ。

a. A 社が商品 a に対する広告の効果を調査するため、広告を流す前後で任意に抽出したユーザに対して知名度に関するアンケートを行なったところ、以下の結果を得た。広告に効果があったかどうかを検証したいので、p 値を計算しなさい。

	商品 a を知っている	商品 a を知らない
広告前	17 人	21 人
広告後	35 人	23 人

b. 検証精度を高めるため、広告後にアンケートを行なうユーザ数を 500 人に増やすこととした。この時、500 人のうち何人のユーザが商品 a を知っていれば、有意水準 0.05 で広告に効果があったと言えるかを調べなさい (広告前の結果については、上の表を利用すること)。

演習 8-3 ピボット選択を行なうガウス–ジョルダンの消去法により、連立一次方程式を解くプログラムを作成せよ。作成後は、様々な方程式を入力し、その動作を確認せよ。

```
import ita
import math


# 評価用データの読み込み
def init_data(fname, dset):

  # ファイルから 1 行読み込み、line に格納する。
  i = 0
  for line in open(fname, "r"):

    # 配列の要素を一つ増やす (取り敢えず初期値はどちらも 0 にしておく)。
    dset.insert(len(dset), [0, 0])

    # まずは、読み込んだ行を " " (空白文字) 毎に区切る。
    items = line.split(" ")

    # 区切られた部分には、まだ ",", "(", ")", "\n" (改行) が残っているので、
    # それらを削除した後、数値データとして配列へ格納する。
    dset[i][0] = int(items[0].strip("(),\n"))
    dset[i][1] = int(items[1].strip("(),\n"))
    i = i + 1


# 相関係数を計算 (使い方: calc_cc("評価用データファイル"))
def calc_cc(fname):

  # 評価用データを格納する 2 次元配列を定義する。
  # 注意: 2 次元配列は、"1 次元配列の各要素が 1 次元配列になっている" という構造なので、
  #       初期値は 1 次元配列として用意する。
  dset = []

  # 評価用データを 2 次元配列 dset に読み込む。
  init_data(fname, dset)

  # X の平均値 X', Y の平均値 Y' を計算する。
  sum_x = 0; sum_y = 0
  for i in range(len(dset)):
    sum_x = sum_x + dset[i][0]
    sum_y = sum_y + dset[i][1]
  ave_x = sum_x/len(dset); ave_y = sum_y/len(dset)

  # E[AB] を計算する。
  sum_ab = 0
  for i in range(len(dset)):
    sum_ab = sum_ab + (dset[i][0] - ave_x)*(dset[i][1] - ave_y)
  ave_ab = sum_ab/len(dset)

  # sqrt(E[A**2])*sqrt(E[B**2]) を計算する。
  sum_a2 = 0; sum_b2 = 0
  for i in range(len(dset)):
    sum_a2 = sum_a2 + (dset[i][0] - ave_x)**2
    sum_b2 = sum_b2 + (dset[i][1] - ave_y)**2
  sq_a2b2 = math.sqrt(sum_a2/len(dset))*math.sqrt(sum_b2/len(dset))

  # 相関係数を計算する。
  cc = ave_ab/sq_a2b2
  print("%.2f" % (cc))
```

付録 1: 線形回帰分析と誤差の二乗和との関係

線形回帰分析では、全観測誤差 $\epsilon^{(i)}$ の二乗和を最小にする係数 $a_0, a_1, a_2, \ldots, a_n$ を求めました。これは、各観測値と式 (3) (6 ページ) との平均距離 (の絶対値) が最小になる場合だと考えられるので、直感的には正しそうに見えますが、厳密な意味では説明できていません。脚注 8 (6 ページ) でも述べた通り、観測誤差 ϵ は何らかの確率に従って発生すると考えられます。これにより、次のような疑問が湧いて来ます。即ち、m 回の観測で得られた誤差 $\epsilon^{(1)}, \epsilon^{(2)}, \ldots, \epsilon^{(m)}$ は、(誤差の) 母集団の傾向を正しく反映したデータ集合になっているのか、という疑問です。別の言い方をすれば、何らかの偶然により、たまたま偏ったデータを観測してしまったのではないか、もしそうでないならば、それをどうやって説明するのか、というわけです (以後、母集団に対して偏っていないことを、**不偏性** (Unbiasedness) と呼ぶことにします/母集団と観測データとの深い関係については、付録 2 を是非読んでみて下さい)。偏ったデータを用いて計算された母集団のパラメータ (今回の例では、係数 $a_0, a_1, a_2, \ldots, a_n$) は、当然乖離が大きいですよね。もちろん、脚注 7 (6 ページ) で述べたように、観測回数を増やせば平均的なデータ集合に近付いて行くと予想されますが、(厳密な意味では) 母集団の総数は不明であるという前提のもと、何回観測すれば、実質的に偶然の偏りを排除できるのかは不明なのです。よって、これを少々厳密に議論するためには、“確率” について少し掘り下げて考えてみる必要があります。

確率変数・確率密度関数

まずは、確率に従って生じている対象を意味する名称を決めましょう。(使い古された例ですが) サイコロを振った時に出る目を Z とします。Z は $1 \sim 6$ の値を取りますが、各値は $\frac{1}{6}$ の確率で決まります。このように、“ある確率法則に従って値が決まるもの” を**確率変数** (Random Variable) と呼びます。以後は、確率変数を対象として議論を進めることになります。また、ある確率に従って発生する現象を、**事象**と呼ぶことにします。

サイコロの例では、確率や確率変数の意味が直感的に分かり易いのですが、今度は次のような場合を考えてみましょう。

「区間 [1, 10] より一つの数値 Z を取り出す時、それが 5 になる確率は?」

以下では、この確率を $P(Z)$ と表わすことにします。$P(Z)$ は、何となく 1/10 になりそうな気もするのですが、実は区間 [1, 10] には数が無限に存在することから、$Z = 5$ になる確率 $P(Z = 5)$ は 0 になってしまいます。少々不思議な感じもしますが、サイコロの例と上の例との違いを考えてみると、これは、Z が離散的に変化する値かあるいは連続的に変化する値か、だと言えます。これより、Z が連続的に変化する場合は、個々の値を対象にすると確率は全て 0 になってしまうので、個々の値ではなく区間で考えることにします。つまり、$Z = 5$ になる確率 $P(Z = 5)$ ではなく、$4 \leq Z \leq 5$ になる確率 $P(4 \leq Z \leq 5)$ を考えるわけです。これだと、$\frac{1}{9}$ だと言えますね。但し、一般には、区間 [1, 10] 内の各数値が等しい確率で選ばれるとは限りません。例えば、区間 [1, 4] は選ばれにくいとか、区間 (4, 10] は選ばれ易いとかです (単純な思考実験では、そんなことはないですが)。よって、各値 Z が選ばれる重み (or 割合) を $f(z)$ で表わし、$a \leq Z \leq b$ になる確率 $P(a \leq Z \leq b)$ を次のように定義します ($f(z)$ は、**確率密度関数** (probability density function — 略称: PDF) と呼ばれます):

$$P(a \leq Z \leq b) = \int_a^b f(z)dz \quad (\text{但し、} \int_{-\infty}^{+\infty} f(z)dz = 1)$$

ここで、未知のデータ X, Y が従う確率について考えてみましょう。未知データは、ある確率に従うと仮定しているので、確率変数として扱われます。この確率変数は、以下のような状況にあると考えられます。

- 未知のデータは、膨大な母集団から観測されたデータである。
- 母集団には、膨大なデータが様々な確率で存在している。

つまり、観測された確率変数 (=未知データ) は少数であるが (母集団の一部であるが)、本来は連続的に変化するとして扱うべきもの、というわけです (この考え方は、後で必要になります)。

正規分布

線形回帰分析の大方針は、観測誤差を最小にする係数を求めるというものでした (その具体的な計算方法として、二乗和の最小値を求めたわけです)。観測誤差も何らかの確率に従う確率変数だと想定されるので、ここでは、観測誤差の確率密度関数について検討してみましょう。

もちろん、観測誤差の散らばりには様々な原因が存在するので、その確率も一概には決められません。しかし、それでは話が進まないので、まずは手始めとして、恐らく観測誤差が備えているだろうと想定される特徴について考えてみることにします。経験的には、以下のような特徴を挙げることができますね。

1. 大きさの等しい正の誤差と負の誤差は、等しい確率で発生する。
2. 小さい誤差は、大きい誤差より発生し易い。
3. ある限界より大きな誤差は、ほぼ発生しない。

上記以外にも細かい特徴を挙げることはできるかも知れませんが、列挙すればするほど検討も複雑になる恐れがあるので、取り敢えず以下では、一般的かつ矛盾もなさそうなこの三つから確率密度関数を導いてみることにしましょう (以後、この三つを誤差が従う本質的特徴と呼ぶことにします)。

確率密度関数を f とします。ここで、誤差の大きさが $\epsilon \sim \epsilon + d\epsilon$ にある確率は、$d\epsilon$ が十分に小さい場合、確率密度関数 f を用いて下記のように表わされます (積分を微小矩形の面積として近似するわけです/数値積分と同じですね)。誤差の本質的特徴 $1\cdot 2$ より、$f(\epsilon)$ は $\epsilon \to 0$ の時に最大値を取ると想定されます。

$$\int_{\epsilon}^{\epsilon+d\epsilon} f(\epsilon)d\epsilon \approx f(\epsilon)d\epsilon$$

また、真の値を X として観測値を x とすると、$\epsilon = x - X$ より $d\epsilon = dx$ なので、$f(\epsilon)d\epsilon = f(x-X)dx$ と書き換えることができます。この時、真の値 X に対する n 回の観測結果が $x_1, x_2, \ldots, x_n$ になる確率、即ち各誤差が $\epsilon_1, \epsilon_2, \ldots, \epsilon_n$ になる条件付き確率は、下記の式で表わすことができます。

$$f(\epsilon_1)d\epsilon \cdot f(\epsilon_2)d\epsilon \cdots f(\epsilon_n)d\epsilon = f(x_1 - X)\cdot f(x_2 - X)\cdots f(x_n - X)\cdot (dx)^n \tag{10}$$

さて、ここからが考え方の難しいところですが、もし観測結果に不偏性がある (観測結果に偏りがない) とすれば、n 回の観測結果が $x_1, x_2, \ldots, x_n$ となる確率が一番高いはず (一番ありふれているはず) と予想することができます。では、真の値 X がどんな値であったら、n 回の観測結果が $x_1, x_2, \ldots, x_n$ となる確率は一番高くなるでしょうか。真の値 X については、誤差の本質的特徴 1 から、観測回数 n を増やすにつれ $X \approx (x_1 + x_2 + \ldots + x_n)/n = \bar{x}$ となりそうなことが予想できます。そこで、式 (10) は $X = \bar{x}$ で最大となると考え[19]、このような f を求めてみることにします ($(dx)^n$ は定数なので、簡略化のため以後の議論では扱わないことにします)。

まずは議論の見通しを良くするため、式 (10) を $P(X) = f(x_1 - X)\cdot f(x_2 - X)\cdots f(x_n - X)$ と置き換えましょう。次に、$P(X)$ は掛け算が多くて扱いにくいので、両辺の対数を取って足し算に変えてから、極大値を調べるために微分します[20]。合成関数の微分を複数回適用すると、$d\log f(x_i - X)/dX = -f'(x_i - X)/f(x_i - X)$ となるので、最終的に下記の関係が成り立ちます。

$$\frac{d\log P(X)}{dX} = \sum_{i=1}^{n}\frac{d\log f(x_i - X)}{dX} = -\sum_{i=1}^{n}\frac{f'(x_i - X)}{f(x_i - X)}, \quad \text{但し、} X = \bar{x} \text{ の時 } \frac{d\log P(X)}{dX} = 0$$

この式はまだ少し複雑なので、もう少し見通しが良くなるように、$f'/f = Q$, $x_i - \bar{x} = y_i$ と置き換えることにします。これにより、“$X = \bar{x}$ の時 $d\log P(x)/dX = 0$” の部分は、下記のように表現できます (負符号を消せることに注意)。

$$\sum_{i=1}^{n}\frac{f'(x_i - \bar{x})}{f(x_i - \bar{x})} = \sum_{i=1}^{n} Q(y_i) = 0 \tag{11}$$

またこの時、y_i については下記が成り立ちます。

$$\sum_{i=1}^{n} y_i = \sum_{i=1}^{n}(x_i - \bar{x}) = (x_1 + x_2 + \ldots + x_n) - n\cdot\{(x_1 + x_2 + \ldots + x_n)/n\} = 0 \tag{12}$$

以上より、式 (12) の関係にある n 個の変数 y_i が満たす式 (11) の Q が分かれば、観測誤差の確率密度関数 f を求めることができます。

---

[19] 真実は未知であり、本当のところは所詮分からないので、誤差が従う本質的特徴を仮定して、ここからどんな合理的な議論ができるかを見てみようというわけです (物理や化学の雰囲気がありますね)。

[20] 一見、対数関数は単調増加関数なので、$\log P(X)$ も単調増加する (つまり極値はない) ように見えます。しかし、これには $P(X)$ が単調増加するという思い込みが入っており、$P(X)$ に極値があれば $\log P(X)$ も極値を持つ可能性があります (例えば、$P(X)$ が 1 を極値とする場合を考えてみよう)。

一見、更に取り付く島もなくなったように見えますが、各関係式の意味を考えると、次のように展望が開けて来ます。式 (12) は、独立に動く y_j $(1 \leq j \leq n-1)$ によって、y_n が束縛されることを意味しています (例えば、3 次元空間を考えると、x, y に応じて z が決まるイメージ)。つまり、y_j と y_n は、$y_n = -y_1 - y_2 - \cdots - y_{n-1}$ という関数になっているわけです。そこで、この線形式を例えば y_1 で偏微分することにより、$\partial y_n / \partial y_1 = -1$ という関係が得られます。これをもとに、式 (11) を y_1 で偏微分してみると、下記の式が得られます。

$$\frac{\partial\{Q(y_1) + Q(y_2) + \cdots + Q(y_n)\}}{\partial y_1} = Q'(y_1) + Q'(y_n)\frac{\partial y_n}{\partial y_1} = Q'(y_1) - Q'(y_n) = 0$$

続いて、式 (11) を y_2 で偏微分すると $Q'(y_2) = Q'(y_n)$ が得られます。以下同様に、$y_3, \ldots, y_{n-1}$ で偏微分して行くと、最終的に $Q'(y_1) = Q'(y_2) = \cdots = Q'(y_{n-1})$ となります。前ページにあるように、y_j $(1 \leq j \leq n-1)$ は独立に動く変数なので、この関係は "$Q'(y) =$ 定数" を意味します。よって、"$Q(y) = ay + b$" と考えることができます。さらに、これを式 (11) に代入して式 (12) を利用すると、下記のように $b = 0$ が得られます ($\because n =$ 任意の自然数)。

$$\sum_{i=1}^{n} Q(y_i) = \sum_{i=1}^{n}(ay_i + b) = a\sum_{i=1}^{n} y_i + n \cdot b = a \cdot 0 + n \cdot b = 0$$

以上の結果を式 (11) に適用すると、($x_i - \bar{x} = y_i$ に注意)、観測誤差の確率密度関数 f について下記の微分方程式が得られます。驚くほど見通しが良くなりましたね。

$$\frac{f'(y)}{f(y)} = ay \tag{13}$$

最後に、$z = f(y)$ としてこの微分方程式を解いてみます。これまで取り上げてきたように、一般的に微分方程式の解を解析的に得ることは難しいですが、この微分方程式については、#4 4.1 節で例示した方法で下記のように解くことができます。

$z = f(y)$ より $\dfrac{dz}{dy} = f'(y)$ なので、式 (13) は $\dfrac{dz}{dy} \cdot \dfrac{1}{z} = ay$ と表わされる。

よって、dy を両辺に掛けることで $\dfrac{1}{z}\, dz = ay\, dy$ となる。

続いて両辺を積分すると、$\log|z| = \dfrac{ay^2}{2} + C$ を得る。

これより、$z = f(y) = \pm \exp\Big(\dfrac{ay^2}{2} + C\Big) = \pm \exp(C) \cdot \exp\Big(\dfrac{ay^2}{2}\Big) = D \cdot \exp\Big(\dfrac{ay^2}{2}\Big)$　　(但し、$\exp(S) = e^S$)

ついに形が見えました。観測誤差の確率密度関数 f は、e の指数関数だったんですね。後は、都合の良いように定数 a, D を決めるだけです。前ページにある誤差の特徴 3 より、$\lim\limits_{y \to \pm\infty} f(y) = 0$ なので (極限的に大きな誤差が生じることはないはず)、$a < 0$ となる必要があります。この条件のもと、"f の分散 $= \sigma^2$" および "$\int_{-\infty}^{\infty} f(y) = 1$" となるように a, D を決めます。この計算は少々複雑なので省略しますが [21]、その結果は $a = -1/\sigma^2, D = 1/\sqrt{2\pi\sigma^2}$ となります。

上で求めた微分方程式の解に a, D を代入すると、最終的に下記の式が得られます。これが、誤差の本質的特徴に従う確率変数の確率密度関数になります。

$$f(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{y^2}{2\sigma^2}\Big) \tag{14}$$

さて、ここで改めて誤差の本質的特徴について考えてみると、これらは経験的に矛盾がなく、また制約も少ない特徴であることが分かります。つまり、別の言い方をすれば、様々な一般的事象が従う特徴だと想定できるわけです。今回は誤差を対象にしたので、式 (14) は $y = 0$ で最大値を取りますが (見事に特徴 2 を満たしていますね)、これを一般的な事象 (改めて、この確率変数を x とします) に拡張すると、(前ページ同様) x に不偏性があるならば式 (14) は $x = \bar{x}$ (x の平均値) で最大値を取るはず、と読み替えることができます。よって、誤差の本質的特徴改め、普遍的な特徴に従う確率変数 x が従う確率密度関数は、式 (14) を x 軸方向に $\bar{x}$ $(= \mu)$ だけ平行移動した下記の式になると言えます。

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\} \tag{15}$$

[21] これまでに学んでいない新たな数学的知識は不要ですが、計算は少々面倒です。気になる人は、ガウス積分について調べてみて下さい。

式 (15) に従う普遍的な分布を**正規分布** (Normal Distribution) と呼びます。正規分布の導出において仮定した特徴は単純なものであり、自然界や社会における様々な事象の発生分布が、(統計的に) 正規分布に従うことが知られています。もちろん、正規分布以外の分布に従う場合も数多く存在しますが、無作為に発生すると考えられる事象については、これまで正規分布を仮定して議論を進めることが多く行われて来ました。特に、母集団がどんな分布であろうと、n 回の観測結果を全て加えた値の分布 (つまり、加算値のバラつき) が、正規分布に従うという中央極限定理は、様々な調査の統計的正当性を支えています。

ところで、上で説明した正規分布の導出方法は、19 世紀初頭にドイツの大数学者**カール・フリードリヒ・ガウス** (Carl Friedrich Gauß) が考案したものです。正規分布が従う式 (15) については、18 世紀前半に二項分布の極限として知られるようになりました。二項分布の確率変数 X が従う確率を s とし、Y が従う確率を $1-s$ とします。この分布における事象を n 回観測する時、X が k 回生じる (Y が $n-k$ 回生じる) 確率は、$P(X=k) = {}_nC_k s^k (1-s)^{n-k}$ で表わされます。ここで $n \to \infty$ とした場合、$\lim_{n\to\infty} \sum_{k=0}^{k=n} {}_nC_k s^k (1-s)^{n-k} = 1$ より、P が漸近する連続関数 p は確率密度関数と見なすことができます。そこで下記を満たすような $p(x)$ を求めてみると、(詳細な計算は省略しますが) 式 (15) と同じ形をしていたというわけです。

$$\int_{-\infty}^{+\infty} p(x)dx = \int_{-\infty}^{+\infty} \Big(\lim_{n\to\infty} {}_nC_x s^x (1-s)^{n-x} \Big) dx = 1$$

線形回帰分析の最尤推定

さて、観測誤差の確率分布が見えて来たので、ようやく線形回帰分析において、これらが母集団の傾向を正しく反映しているかどうかの議論ができるようになりました。各観測誤差 $\epsilon^{(i)}$ を確率変数とし、これが従う確率密度関数を $p(\epsilon)$ とします (形としては式 (15) に従いますが、以下の議論にあるように全く同じというわけではありません)。まず、m 回の観測によって得られた誤差が、$\epsilon^{(1)}$, $\epsilon^{(2)}$, …, $\epsilon^{(m)}$ になる確率を考えます。これは前節同様、次のような条件付き確率として考えることができます。

$$\int_{-\Delta}^{+\Delta} p(\epsilon^{(1)})dx \int_{-\Delta}^{+\Delta} p(\epsilon^{(2)})dx \cdots \int_{-\Delta}^{+\Delta} p(\epsilon^{(m)})dx$$

Δ は、各 $\epsilon^{(i)}$ 周辺の微細区間を表わしているので、この式も前節同様、以下のように近似できます。

$$p(\epsilon^{(1)})\, p(\epsilon^{(2)}) \cdots p(\epsilon^{(m)})\, d\epsilon\, d\epsilon \cdots d\epsilon$$

各 $\epsilon^{(i)}$ には式 (5) (6 ページ) の関係があるので、$p(\epsilon)$ は、観測されたデータ $x^{(i)} = (x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)}, y^{(i)})$ や、係数 $a_0, a_1, a_2, \ldots, a_n$ も関係しています。また、$d\epsilon$ は定数なのでこれを省略すると、下記に示す式 (16) の値が大きくなるほど、m 回の観測によって得られた誤差集合が、$\epsilon^{(1)}$, $\epsilon^{(2)}$, …, $\epsilon^{(m)}$ となる確率は大きい、つまり、不偏的なデータ集合であると言えるわけです (誤差の二乗和が云々と言うだけでなく、実はこの議論が必要でした)。

$$p(x^{(1)}, a_0, a_1, a_2, \ldots, a_n)\; p(x^{(2)}, a_0, a_1, a_2, \ldots, a_n) \cdots p(x^{(m)}, a_0, a_1, a_2, \ldots, a_n) \tag{16}$$

$x^{(1)}, x^{(2)}, \ldots, x^{(m)}$ は既に観測されたデータであり、これらもまた定数と見なすことができるので、残りは、式 (16) を最大化する $a_0, a_1, a_2, \ldots, a_n$ を求めればよいわけです。このような係数が、観測誤差を最も自然に表わしている (別の言い方をすれば、説明変数と目的変数の定量的な関係を一番精度良く表わしている) と言うことができます。これを、**最尤推定** (Maximum Likelihood Estimation — 略称: MLE) と呼びます。

いよいよ最後です。式 (16) にある各 p は、式 (15) において、x の代わりに式 (5) の右辺を代入したものです。これを最大化するには、このままでは計算が大変なので前節同様、対数を取って足し算に変えます。この時、式 (15) は難しそうな顔つきをしていますが、基本的には $e^{-\square^2}$ という形をしているので、実は対数が消えて全て簡潔な足し算に変わります。つまり、式 (16) は、“式 (5) 右辺の 2 次式の $\sum$” だけに変形できるわけです。実際には符号が負になるので (式 (15) の指数部分は $-\square^2$ ですね)、“式 (5) 右辺の 2 次式の $\sum$” を最小化する $a_0, a_1, a_2, \ldots, a_n$ を求めることになります。

以上より、線形回帰分析において、説明変数と目的変数の定量的な関係を一番精度良く表わす係数 $a_0, a_1, a_2, \ldots, a_n$ は、全観測誤差 $\epsilon^{(i)}$ の二乗和を最小にする $a_0, a_1, a_2, \ldots, a_n$ であることが不偏的に言えたわけです[22]。

[22] 今回は、観測誤差 ϵ が従う確率密度関数 $p(\epsilon)$ の掛け算である式 (16) が、対数を取ることで綺麗に分解できたため、これを最大化するパラメータ

付録 2: 母集団と観測データとの関係

付録 1 で説明した線形回帰分析の誤差を最小化する議論では、観測されたデータの誤差が母集団の傾向を正しく反映した集合になっているかどうかを慎重に扱いました。これは、母集団の分布に関連する未知のパラメータ θ を、観測データの分布から得られたパラメータ $\widehat{\theta}$ により**推定**したい (or 代用したい) からです。そこで、まずは観測データの不偏性をどのように扱ったら (or 定義したら) 良いかを考えてみましょう。n 個の観測データから推定したパラメータを $\widehat{\theta}_n$ とします。各 $\widehat{\theta}_1, \widehat{\theta}_2, \ldots, \widehat{\theta}_n$ は当然散らばりますが、観測データに不偏性があるとすれば、n を増やすにつれ、$\widehat{\theta}_n$ は母集団のパラメータ θ に近付いて行くと予想されます。そこで、"$E(\widehat{\theta}_n) = \theta$" となる θ_n を、母集団のパラメータを偏りなく推定できているパラメータとして、**不偏推定量**と呼ぶことにします。以下では、観測データの平均および分散を例として、これらが不偏推定量になっているかどうかを調べてみることにしましょう。ここでは、確率変数 X, Y の平均と分散に対する下記の関係式を利用しているので、注意して下さい:

$$E(aX + bY) = aE(X) + bE(Y)$$
$$V(aX + bY) = a^2V(X) + b^2V(Y)$$

不偏平均

母集団の平均値を μ とします。また、n 個の観測データを $X = \{X_1, X_2, \ldots, X_n\}$ とします。この時、観測データの平均値は $\frac{1}{n}\sum_{i=1}^{n} X_i$ となりますが、以下では、これが母集団における μ の不偏推定量になっているかどうかを確認してみます。観測データの平均の平均 (つまり $E(\widehat{\theta}_n)$) は、以下のようになります:

$$E\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) = E\left(\frac{1}{n}X_1 + \frac{1}{n}X_2 + \cdots + \frac{1}{n}X_n\right) = \frac{1}{n}E(X_1) + \frac{1}{n}E(X_2) + \cdots + \frac{1}{n}E(X_n) \tag{17}$$

上の式に出て来る $E(X_i)$ ですが、これは「i 番目に観測されたデータの平均値[23]」を意味しています。ここで、1 度の観測機会でデータを n 回観測し、複数の観測機会を設ける場合を考えます。各観測機会において i 番目に観測されたデータを集め、その平均を計算すると、観測機会を増やすにつれ、この平均値は母集団の平均値 μ に近付くと予想されます。つまり、$E(X_i) = \mu$ と考えられるわけです。よって、これを式 (17) に代入すると、下記のようになります:

$$E\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) = \frac{1}{n}E(X_1) + \frac{1}{n}E(X_2) + \cdots \frac{1}{n}E(X_n) = \frac{1}{n}\mu + \frac{1}{n}\mu + \cdots \frac{1}{n}\mu = \frac{1}{n} \cdot n\mu = \mu$$

以上より、観測データの平均値は、母集団の平均値に対する不偏推定量になっていることが分かりました。観測データの平均値を母集団の平均値として扱うことが、一番自然だというわけです。

因みに、本節の節題は "不偏平均" となっていますが、観測データの平均値は不偏推定量となっていることから、不偏推定量であることを示す不偏という用語を通常は付けません。統計分野では観測データではなく標本という用語を使うので、不偏平均の代わりに "標本平均" という用語を使います。つまり、標本平均と書かれていれば、暗黙に不偏平均を意味しているわけです。本資料では、理解し易さを考慮して、不偏平均という用語を使いました。

(今回の例では係数 a_j) を解析的に求めることができました。しかし、一般的には都合良く求めることができないため、数値的な計算を行なうことになります。数値的な計算では、例えば 4 次関数のように極値が複数ある場合、全ての極値を求めたかどうかの確証が得られません。これは、見逃した極値の方が最適 (つまり最大 · 最小) であるならば、それを求められないことを意味します。最尤推定の考え方自体は、どのような統計モデルにも適用できる普遍的な考え方ですが、数値的な計算により求めたパラメータが正しいとは限らない場合があるわけです。

[23] この意味がよく分からない方は、例えば次の事例を思い浮かべて下さい。サイコロを n 回振るという操作を t 組行なう時、各組で 3 回目に出る目の平均値を考えます。この時、1 が出るのか、5 が出るのかは分かりませんが、期待値は 3.5 と計算できます。つまり、平均値は 3.5 と考えるわけです。なお、念のためですが、期待値と平均値には次の関係があります。全体で n 回発生する事象において、値 x_i が a_i 回あったとします。この時、平均値は $\sum(x_i * a_i)/n$ と計算され、期待値は $\sum x_i * (a_i/n)$ と計算されますが、これは同値ですね。

不偏分散

今度は、観測データから得られた分散の不偏性について調べてみましょう。母集団の分散を σ^2 とし、観測データの平均を $\bar{X}$ とします (他の記号は前節と同じです)。この時、観測データの分散は $\frac{1}{n}\sum_{i=1}^{n}(X_i-\bar{X})^2$ となりますが、前節と同じく、これが母集団における σ^2 の不偏推定量になっているかどうかを確認してみます:

$$\begin{aligned}
E\Big(\frac{1}{n}\sum_{i=1}^{n}(X_i-\bar{X})^2\Big) &= E\Big(\frac{1}{n}\sum_{i=1}^{n}\big\{(X_i-\mu)-(\bar{X}-\mu)\big\}^2\Big)\\
&= E\Big(\frac{1}{n}\sum_{i=1}^{n}\big\{(X_i-\mu)^2-2(X_i-\mu)(\bar{X}-\mu)+(\bar{X}-\mu)^2\big\}\Big)\\
&= E\Big(\frac{1}{n}\sum_{i=1}^{n}(X_i-\mu)^2\Big)-2E\Big(\frac{1}{n}\sum_{i=1}^{n}(X_i-\mu)(\bar{X}-\mu)\Big)+E\Big(\frac{1}{n}\sum_{i=1}^{n}(\bar{X}-\mu)^2\Big) \qquad (13)
\end{aligned}$$

この式は少々複雑なので、1 項ずつ見てみることにします。

まずは第 1 項です。

$$\begin{aligned}
E\Big(\frac{1}{n}\sum_{i=1}^{n}(X_i-\mu)^2\Big) &= \frac{1}{n}\sum_{i=1}^{n}E\Big((X_i-\mu)^2\Big)\\
&= \frac{1}{n}\Big\{E\Big((X_1-\mu)^2\Big)+E\Big((X_2-\mu)^2\Big)+\cdots+E\Big((X_n-\mu)^2\Big)\Big\} \qquad (14)
\end{aligned}$$

上の式にある $E\big((X_i-\mu)^2\big)$ ですが、前節と同様、各観測 “機会” において i 番目に観測されたデータを集めて $(X_i-\mu)^2$ の平均値を計算すると、観測機会を増やすにつれ、この平均値は母集団の分散 σ^2 に近付いて行くと予想されます[24]。よって、$E\big((X_i-\mu)^2\big)=\sigma^2$ とし、これを式 (14) に代入することで、第 1 項は $(1/n)(\sigma^2+\cdots+\sigma^2)=\sigma^2$ となります。

続いて第 2 項です (計算を簡略化するため、以下では係数の 2 を省いています)。

$$\begin{aligned}
E\Big(\frac{1}{n}\sum_{i=1}^{n}(X_i-\mu)(\bar{X}-\mu)\Big) &= E\Big(\frac{1}{n}\Big\{(X_1-\mu)(\bar{X}-\mu)+(X_2-\mu)(\bar{X}-\mu)+\cdots+(X_n-\mu)(\bar{X}-\mu)\Big\}\Big)\\
&= E\Big(\frac{1}{n}(X_1+X_2+\cdots X_n-n\mu)(\bar{X}-\mu)\Big)\\
&= E\Big(\frac{1}{n}(n\bar{X}-n\mu)(\bar{X}-\mu)\Big)=E\Big((\bar{X}-\mu)(\bar{X}-\mu)\Big)=E\Big((\bar{X}-\mu)^2\Big)=V(\bar{X})
\end{aligned}$$

観測データの平均値 $\bar{X}$ に対する分散 $V(\bar{X})$ は、下記のようになります。

$$\begin{aligned}
V(\bar{X}) &= V\Big(\frac{1}{n}\sum_{i=1}^{n}X_i\Big)=V\Big(\frac{1}{n}X_1+\frac{1}{n}X_2+\cdots+\frac{1}{n}X_n\Big)=\frac{1}{n^2}V(X_1)+\frac{1}{n^2}V(X_2)+\cdots\frac{1}{n^2}V(X_n)\\
&= \frac{1}{n^2}\Big\{E\Big(\big(X_1-E(X_1)\big)^2\Big)+E\Big(\big(X_2-E(X_2)\big)^2\Big)+\cdots+E\Big(\big(X_n-E(X_n)\big)^2\Big)\Big\} \qquad (15)
\end{aligned}$$

上の式にある $E(X_i)$ は、前節の議論より $E(X_i)=\mu$ です。また、これを代入した $E\big((X_i-\mu)^2\big)$ は、上の議論により $E\big((X_i-\mu)^2\big)=\sigma^2$ となります。よって、これを式 (15) に代入することで、第 2 項は $(1/n^2)(\sigma^2+\cdots+\sigma^2)=\sigma^2/n$ となります。

最後は第 3 項です。

$$E\Big(\frac{1}{n}\sum_{i=1}^{n}(\bar{X}-\mu)^2\Big)=\frac{1}{n}\sum_{i=1}^{n}E\big((\bar{X}-\mu)^2\big)=\frac{1}{n}\Big\{E\big((\bar{X}-\mu)^2\big)+\cdots+E\big((\bar{X}-\mu)^2\big)\Big\}=E\big((\bar{X}-\mu)^2\big)=V(\bar{X})$$

第 3 項は、第 2 項と同じ値になりました (正確に言えば、計算を簡略化するため、第 2 項では係数の 2 を省いているのですが)。$V(\bar{X})$ は、上の議論より σ^2/n です。

---

[24] x_k を母集団のデータ, N を母集団のデータ数とすると、$\sigma^2=\frac{1}{N}\sum_{k=1}^{N}(x_k-\mu)^2$ です。

以上より、観測データの分散に対する不偏推定量は、$E\Big(\frac{1}{n}\sum_{i=1}^{n}(X_i-\bar{X})^2\Big)=\sigma^2-2\frac{1}{n}\sigma^2+\frac{1}{n}\sigma^2=\frac{n-1}{n}\sigma^2$ となります。この結果を見てみると、観測データの分散は、母集団の分散に対する不偏推定量になっていません。観測データの分散を $n/(n-1)$ 倍すると、母集団の分散に対する不偏推定量になるわけです。これは、観測データの分散を、前節で調べた平均と同じように母集団の分散と見なしてしまうと、母集団の分散を小さく見積もってしまうことを意味しています。よって、観測データをもとにした不偏分散を $\frac{n}{n-1}\cdot\frac{1}{n}\sum_{i=1}^{n}(X_i-\bar{X})^2=\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\bar{X})^2$ と定義します[25]。この定義は、

$$E\Big(\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\bar{X})^2\Big)=E\Big(\frac{n}{n}\cdot\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\bar{X})^2\Big)=\frac{n}{n-1}E\Big(\frac{1}{n}\sum_{i=1}^{n}(X_i-\bar{X})^2\Big)=\frac{n}{n-1}\cdot\frac{n-1}{n}\sigma^2=\sigma^2$$

より、母集団に対する不偏分散になっていることが確認できます。平均値に比べると、計算がだいぶ複雑でしたね。

推定量について

付録 1 では最尤推定量を、この付録では不偏推定量を取り上げました。両者にはどのような相違があるのか、推定量の意味について少し整理しておきます。

推定とは、母集団が持つパラメータを予想することですが、推定量に期待される特徴は一つ (例えば不偏性) だけではありません。不偏推定量 θ_n について言えば、$E(\widehat{\theta}_n)=\theta$ を満たすことで、母集団のパラメータを偏りなく推定できていると考えられます。しかし、ここでは θ_n の平均だけに着目しており、平均値としては θ になるものの、観測 “機会” 毎にバラバラの θ_n が得られるようでは、あまり好ましくありません。つまり、推定量自体の分散も小さくなるような推定方法が必要となります。推定量が持つこのような特徴は、**有効性** (Efficiency) と呼ばれており、この他にも様々な特徴が望まれます。

最尤推定は、母集団の確率分布あるいは確率密度関数の形が分かっている場合に、そのパラメータを求める方法として利用されます[26]。最尤推定量は、不偏推定量になっていないこともありますが、有効性を始めとした他の特徴を解析的に計算できる場合が多いため (加えて、これらが望ましい特徴を示すため)、推定量を求める代表的な方法の一つとなっています。

最後に、(これまで述べて来た推定量を求める方法とは別に) 推定の種類 (or 大枠) について少し触れておきます。本資料では全て、“最適と想定される母集団のパラメータ” を推定する方法を取り上げました。このような推定を**点推定** (Point Estimation) と呼びます。但し、上で述べた推定量の有効性が低い場合 (例えば、観測データの散らばりが大きいなど)、点推定により得られたパラメータは、母集団のパラメータから乖離している可能性が高くなります。そこで、“母集団のパラメータ θ は、確率 P で値 A ∼ B 間にある” といった、推定量に幅を持たせ、その信頼度も同時に推定する方法があります。このような推定を**区間推定** (Interval Estimation) と呼びます。

[25] 少々混乱しますが、n で割る分散は、確率分布 (or 確率密度関数) が明らかで解析的な計算をする際の分散、$n-1$ で割る分散は、未知な母集団に対する観測データをもとにした分散という関係にあります。また、統計分野では、観測データ数に対して自由度という用語を使います。これより、$n-1$ で割る分散は、自由度 $n-1$ の分散と呼ばれることがあります。

[26] 一般に、$E(\widehat{\theta}_n)=\theta$ を満たす任意のパラメータを求めることは難しいですが、最尤推定は、どのような確率分布あるいは確率密度関数であっても、その形が分かっていれば、原理的にパラメータを推定できる普遍的な方法です。