2010.03.18　2015.05.12 改訂

SU-FreeSBIE で実験データを解析する

第３回 C言語で作る超お手軽最小自乗フィットプログラム２：y 切片あり

摂南大学 理工学部 電気電子工学科　井上雅彦

参考：http://sprite.eng-scl.setsunan.ac.jp/sst_lab/2009/lms-3.html

前回はデータポイントを原点を通る直線 $y = ax$ にフィッティングするプログラムを作りました。C言語のソースファイルを図１に示します。今回はこれを改良して y 切片を持つ直線 $y = ax + b$ にフィッティングするプログラムを作ります。

表１のような測定結果をもとに図２のようなグラフを作成しました。各データ点と直線との残差を計算します。

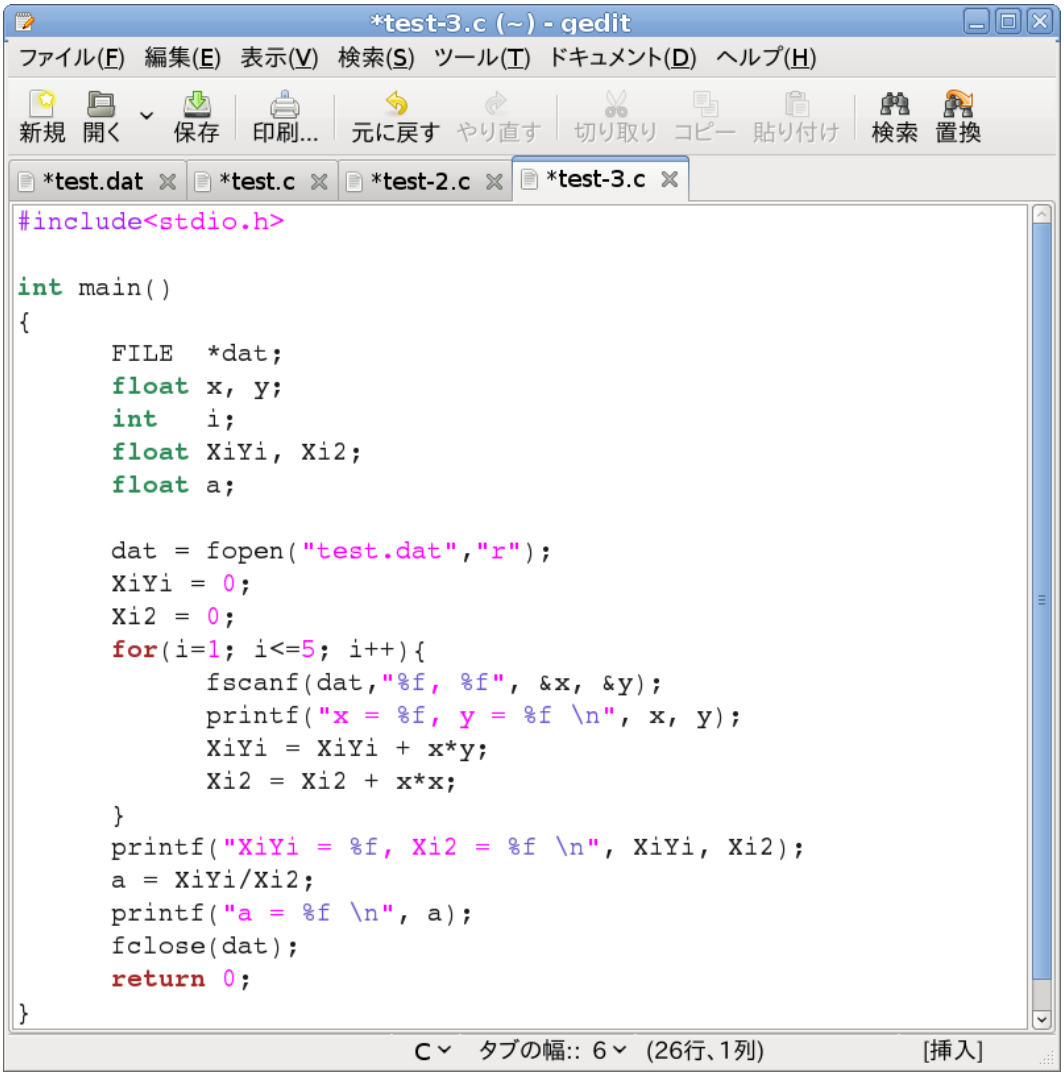

図１：前回作成した test-3.c

表 1 : 実験結果

(番号) i	(設定値) x_i	(測定値) y_i
1	2.0	20.1
2	4.0	29.3
3	6.0	33.0
4	8.0	45.0
5	10.0	47.0

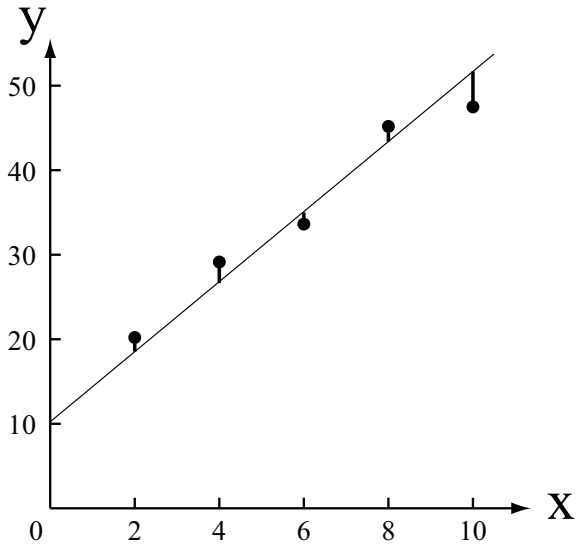

図 2 : y 切片を持つ直線 $y = ax + b$ に
フィッティングする

各設定値 x_i に対する理論値（直線上の値）は $ax_i + b$ なので，各データポイントにおける残差（理論値と測定値の差）は $ax_i + b - y_i$ となります。従って残差自乗和は，

$$\begin{aligned} S(a,b) &= \Sigma_{i=1}^{5}(ax_i + b - y_i)^2 \\ &= \Sigma_{i=1}^{5}(a^2x_i^2 + b^2 + y_i^2 + 2abx_i - 2by_i - 2ax_iy_i) \\ &= a^2\Sigma_{i=1}^{5}x_i^2 + \Sigma_{i=1}^{5}b^2 + \Sigma_{i=1}^{5}y_i^2 + 2ab\Sigma_{i=1}^{5}x_i - 2b\Sigma_{i=1}^{5}y_i - 2a\Sigma_{i=1}^{5}x_iy_i \end{aligned}$$

$S(a,b)$ は a についても b についても下に凸の二次関数で必ず最小値を持ちます。$S(a,b)$ を最小とする a,b を求めましょう。このとき a 方向でも b 方向でも接線の傾きは 0 となりますから，

$$\frac{\partial S(a,b)}{\partial a} = 2a\Sigma_{i=1}^{5}x_i^2 + 2b\Sigma_{i=1}^{5}x_i - 2\Sigma_{i=1}^{5}x_iy_i = 0 \qquad (1)$$

$$\frac{\partial S(a,b)}{\partial b} = 10b + 2a\Sigma_{i=1}^{5}x_i - 2\Sigma_{i=1}^{5}y_i = 0 \qquad (2)$$

a と b に関する連立一次方程式となります。この問題ではデータポイントの数は 5 個ですが，一般化のため n 個とすると，

$$2a\Sigma_{i=1}^{n}x_i^2 + 2b\Sigma_{i=1}^{n}x_i - 2\Sigma_{i=1}^{n}x_iy_i = 0 \qquad (3)$$

$$2nb + 2a\Sigma_{i=1}^{n}x_i - 2\Sigma_{i=1}^{n}y_i = 0 \qquad (4)$$

これを解いて，

$$a = \frac{n\Sigma_{i=1}^{n}x_iy_i - \Sigma_{i=1}^{n}x_i\Sigma_{i=1}^{n}y_i}{n\Sigma_{i=1}^{n}x_i^2 - (\Sigma_{i=1}^{n}x_i)^2} \qquad (5)$$

$$b = \frac{\Sigma_{i=1}^{n}x_i^2\Sigma_{i=1}^{n}y_i - \Sigma_{i=1}^{n}x_iy_i\Sigma_{i=1}^{n}x_i}{n\Sigma_{i=1}^{n}x_i^2 - (\Sigma_{i=1}^{n}x_i)^2} \qquad (6)$$

この（5）式と（6）を使って直線の傾き a と y 切片 b を求めるプログラムをステップバイステップで作りましょう。Windows をお使いの方は VMwarePlayer（フリーウエア）を，また MacOSX をお使いの方は VMwareFusion （有償）を使って SU-FreeSBIE を起動してください。

まず，データファイルを作ります。汎用テキストエディタ gEdit のアイコンをクリックして起動します。「ファイル」→「開く」で前回作成したデータファイル test. dat を開き，図 3 のように 1 行目と 2 行目を修正して「ファイル」→「別名で保存」で test-2.dat というファイル名で保存してください。

図 3 ：gEdit でデータファイルを作成する

いよいよ図 1 のソースファイルに改良を加えてゆきます。まず サメンション $\Sigma_{i=1}^{n} x_i$ と $\Sigma_{i=1}^{n} y_i i$ を表す float 型変数として Xi および Yi を定義します。これらの数値をまず 0 に初期化しておいて， for 文の中で x あるいは y を足してゆくことでサメンションを計算します。最後に計算結果を表示しましょう。またデータ数 n を表す変数として n を定義します。これは n = 5 に初期化しておきます。for 文の後で計算されたサメンションの結果を利用して a と b を計算し画面に表示します。データファイル名も test.dat から test-2.dat に変更します。図 4 に示すように test-3.c を修正し,「ファイル」→「別名で保存」を選び test-4.c という名前で保存してください。

続いて端末（terminal）を開き，コマンドラインから test-4.c をコンパイルして test-4 という実行ファイルを作り，実行します。画面に読み込んでだデータ，各サメンションの値，最後に a と b の値が表示されれば成功です。

SU-FreeSBIE% cc test-4.c -o test-4 （ENTER）

SU-FreeSBIE% ./test-4 （ENTER）
x = 2.000000, y = 20.100000
x = 4.000000, y = 29.299999
x = 6.000000, y = 33.000000
x = 8.000000, y = 45.000000
x = 10.000000, y = 47.000000
XiYi = 1185.400024, Xi2 = 220.000000, Xi = 30.000000, Yi = 174.399994
a = 3.475002, b = 14.029989

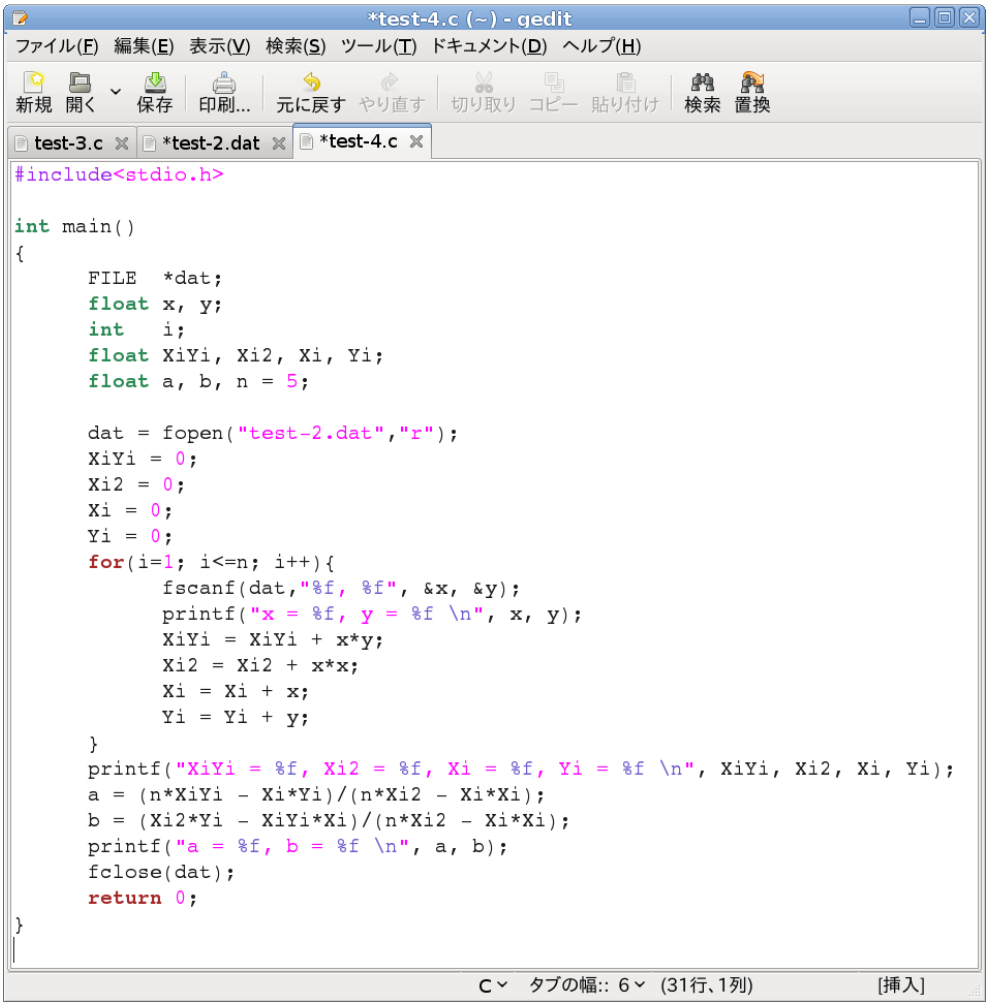

図４：test-4.c

練習問題

今回のプログラムではデータポイントの数が変更される度にプログラムを（ n の初期値を）書き換える必要がある。これをなくすためには for 文のところを次のように書き換えれば良い。 test-4.c の for 文のところを書き換えてプログラムが正常動作することを確認せよ。

```
n=0;
while(1){
        fscanf(dat,"%f, %f", &x, &y);
        if(feof(dat)){break;}
        printf("x = %f, y = %f ", x, y);
        XiYi = XiYi + x*y;
        Xi2 = Xi2 + x*x;
        Xi = Xi + x;
        Yi = Yi + y;
        n = n + 1;
}
```

以上