数理アルゴリズムとシミュレーション　演習課題6　提出日：2020年11月26日
201811319 永崎遼太

課題2

(2-1)
まず、次のようなコードを作成した。

ーーーーー
```
A = zeros(100,10);

for i=1:100
    for j=1:10
        A(i,j)=abs(i-j+1);
    end
end

A_ = transpose(A);

x = ones(10,1);

maxit = 2000;
tol = 1e-05;

for it = 1:maxit
    r = 0;
    w = A_*A*x;
    x = w / norm(w);
    l1 = dot(x, A_*A*x);
    l2 = dot(x,x);
    lambda = l1/l2;
    r = norm(A_*A*x - lambda*x);
    if r < tol
        break;
    end
    T_(it)=r;
    t=it;
end

sqrt(lambda)
```
ーーーーー

これを実行すると、次のように最大特異値が得られる。

ーーーーー

```
ans =

   1.7192e+03
```

ーーーーー

また、次のようなコードを作成した。

ーーーーー

```
A = zeros(100,10);

for i=1:100
   for j=1:10
      A(i,j)=abs(i-j+1);
   end
end

A_ = transpose(A);

x = ones(10, 1);

maxit = 2000;
tol = 1e-05;

for it = 1:maxit
   r = 0;
   w = A_*A\x;
   x = w / norm(w);
   l1 = dot(x, x);
   l2 = dot(x,A_*A\x);
   lambda = l1/l2;
   r = norm(A_*A*x - lambda*x);
   if r < tol
      break;
   end
   T_(it)=r;
   t=it;
end

sqrt(lambda)
```

ーーーーー

これを実行すると、次のように最小特異値が得られる。

ーーーーー

ans =

    0.5144

ーーーーー

Aに対してsvd関数を実行した結果は次のようになる。

ーーーーー

ans =

1.0e+03 *

```
    1.7192
    0.0438
    0.0089
    0.0031
    0.0017
    0.0011
    0.0008
    0.0007
    0.0006
    0.0005
```

ーーーーー

この結果より、べき乗法、逆反復法を用いて求めた値は正しいことが確かめられた。

課題3

次のようなコードを作成した。

ーーーーー

```
load data.mat
X_ = mean(X)
Y_ = mean(Y)
x1 = zeros(200,1);
x2 = zeros(200,1);
for i = 1:200
    x1(i)=X(i)-X_;
end

for i = 1:200
    x2(i)=Y(i)-Y_;
end

A = [x1 x2];
A_ = transpose(A);

x = ones(2,1);

maxit = 2000;
tol = 1e-05;

for it = 1:maxit
    r = 0;
    w = A_*A*x;
    x = w / norm(w);
    l1 = dot(x, A_*A*x);
    l2 = dot(x,x);
    lambda = l1/l2;
    r = norm(A_*A*x - lambda*x);
    if r < tol
        break;
    end
    T_(it)=r;
    t=it;
end
t
x

slps = x(2)/x(1);
ints = -x(2)/x(1)*X_+Y_;
```

```
x = linspace(-8,10);
y = slps*x +ints;
plot(x,y,'r','LineWidth',3);

hold on
scatter(X,Y,'b');
hold off
```

ーーーーー

これを実行した結果、図1のようなグラフが描画される。

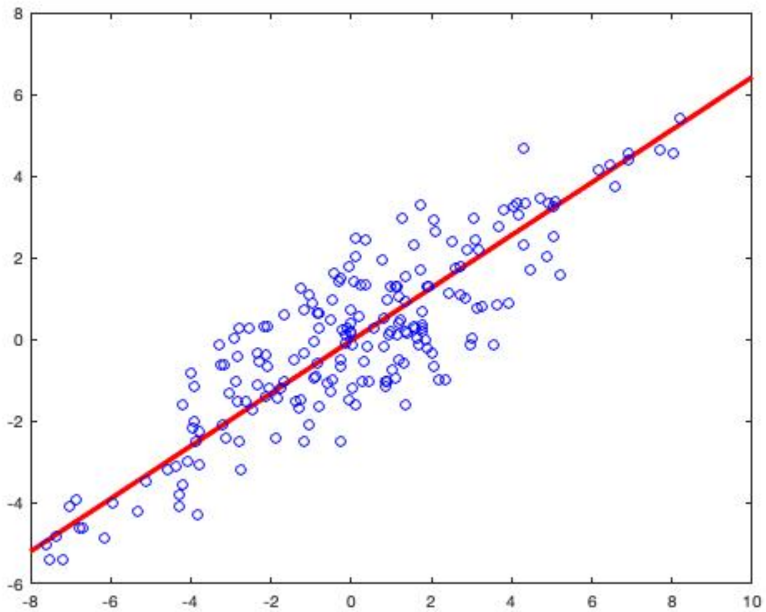

図1 主成分方向を示した直線を重ねたグラフ

課題4

(4-1)
次のようなコードを作成した。

―――――
```
rng(0);
U1 = rand(100,10);
V1 = rand(70, 10);

U2 = rand(100, 40);
V2 = rand(70, 40);

A = U1*V1' + 1.0E-5*U2*V2' + 1.0E-10*rand(100,70);


[U, S, V] = svd(A);
s = diag(S)
n = 70;
fro = zeros(70,1);

for i=1:n
   k = i;
   A_ = zeros(100,70);
   for j=1:k
      u_tmp = U(:,j);
      v_tmp = transpose(V(:,j));
      tmp = s(j)*u_tmp*v_tmp;
      A_ = A_ +tmp;
   end
   val = A - A_;
   fro(i) = norm(val, 'fro');
end

semilogy(fro)
```
―――――

これを実行した結果、図2のようなグラフが描画される。

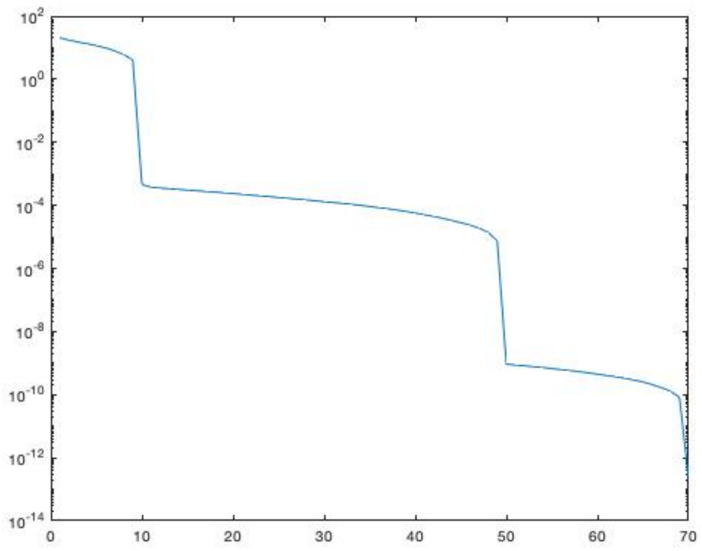

図2 kの値ごとのフロベニウスノルムに関する誤差

(4-2)
次のようなコードを作成した。

——————

```
rng(0);
U1 = rand(100,10);
V1 = rand(70, 10);

U2 = rand(100, 40);
V2 = rand(70, 40);

A = U1*V1' + 1.0E-5*U2*V2' + 1.0E-10*rand(100,70);


[U, S, V] = svd(A);
s = diag(S);
n = 70;
fro = zeros(70,1);

for i=1:n
   k = i;
   A_ = zeros(100,70);
   for j=1:k
      u_tmp = U(:,j);
```

```
        v_tmp = transpose(V(:,j));
        tmp = s(j)*u_tmp*v_tmp;
        A_ = A_ +tmp;
    end
    val = A - A_;
    fro(i) = norm(val, 'fro');
end

value = zeros(70,1);

for i=1:n-1
    k = i;
    l = k + 1;
    value_ = 0;
    for j=l:n
        tmp = s(j)*s(j);
        value_ = value_ + tmp;
    end
    value(i) = sqrt(value_);
end

t = [value fro]

semilogy(value)
```
―――――

これを実行すると、まず、(4-1）で求めたフロベニウスノルムに関する誤差を格納したベクトルと(4-2)で求めた値を格納したベクトルが連結されたベクトルが次のようにして出力される。

―――――

```
t =

   20.5575   20.5575
   17.3855   17.3855
   15.0081   15.0081
   13.2132   13.2132
   11.5450   11.5450
    9.7808    9.7808
    7.9254    7.9254
    6.0216    6.0216
    4.1010    4.1010
    0.0005    0.0005
    0.0004    0.0004
    0.0004    0.0004
    0.0003    0.0003
```

0.0003	0.0003
0.0003	0.0003
0.0003	0.0003
0.0003	0.0003
0.0003	0.0003
0.0003	0.0003
0.0002	0.0002
0.0002	0.0002
0.0002	0.0002
0.0002	0.0002
0.0002	0.0002
0.0002	0.0002
0.0002	0.0002
0.0002	0.0002
0.0001	0.0001
0.0001	0.0001
0.0001	0.0001
0.0001	0.0001
0.0001	0.0001
0.0001	0.0001
0.0001	0.0001
0.0001	0.0001
0.0001	0.0001
0.0001	0.0001
0.0001	0.0001
0.0001	0.0001
0.0001	0.0001
0.0001	0.0001
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000
0.0000	0.0000

```
   0.0000    0.0000
   0.0000    0.0000
   0.0000    0.0000
   0.0000    0.0000
   0.0000    0.0000
   0.0000    0.0000
        0    0.0000
```

ーーーーー

(4-2)で求まる値はkが1〜n-1の範囲であり、その値に関して(4-1）で求めたフロベニウスノルムに関する誤差と一致することが確認できる。
また、(4-2)で求めた値をグラフにした図3が出力される。

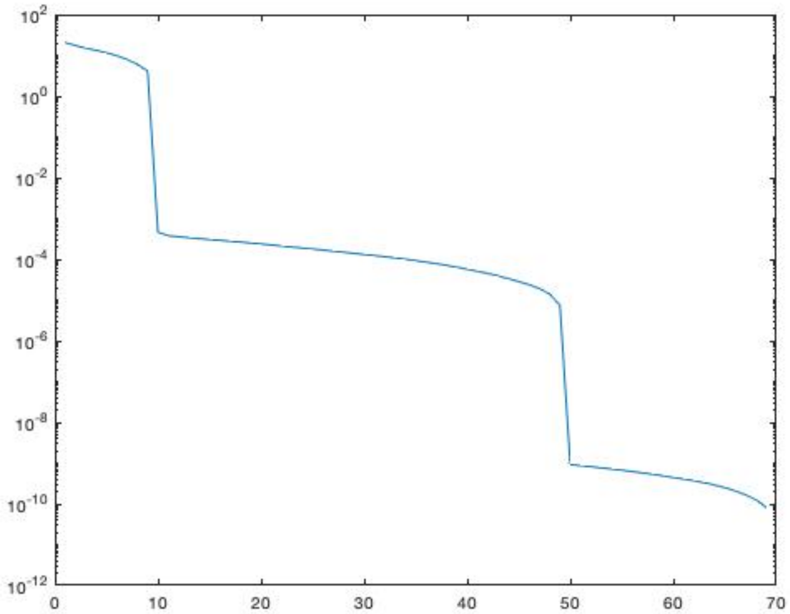

図3 （4-2)で求めた値のグラフ

これは、k=70以外の箇所は図2と一致していることがわかる。