1.

(a)

```
G =

     0     0     0     0
     1     0     1     0
     1     0     0     1
     0     0     1     0

c =

     2     0     2     1
```

```
A =

    0.0375    0.2500    0.0375    0.0375
    0.4625    0.2500    0.4625    0.0375
    0.4625    0.2500    0.0375    0.8875
    0.0375    0.2500    0.4625    0.0375
```

Script:

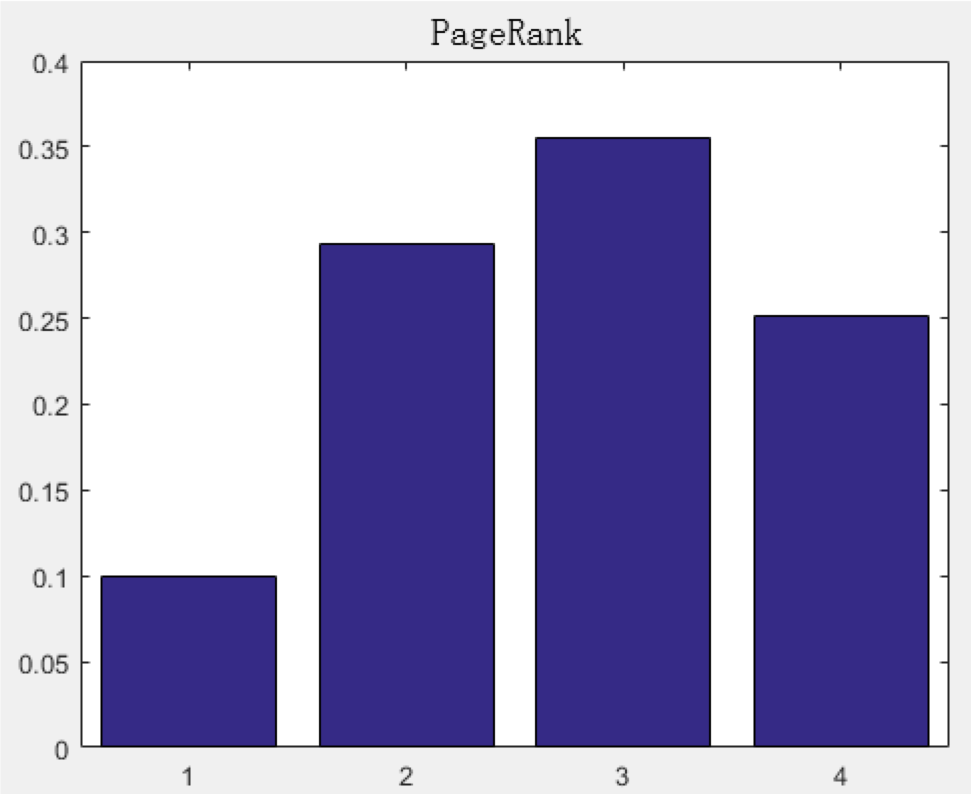
```
P = 0.85;
G = [0,0,0,0;1,0,1,0;1,0,0,1;0,0,1,0];

[n,~] = size(G);
c = sum(G);
k = find(c~=0);
z = 1/n*ones(1,n);
z(k) = (1-P)/n;
D = sparse(k,k,1./c(k),n,n);
e = ones(n,1);
A = P*G*D + e*z;
```

(b)I define the 'change' as the maximum of absolute value of every element's relative difference. And here I set the tolerance to be e-5.

And I get the final PageRank vector to be

```
x1 =

   0.099859622801701
   0.293458162655756
   0.355664613844579
   0.251017600697964
```

With the PageRank bar plot



Script:

```
x0=1/n*ones(n,1);
x1=A*x0;
tol=0.001;
%tolerence: the relative change of every element is smaller than tol
%here we don't make any approximation on A so x is always normalized
while ( max(abs((x1-x0)./x0)) > tol )
    x0=x1;
    x1=A*x0;
end
```

(c) I use partial pivoting Gauss Elimination to solve (I-pGD)x=e, I get

```
x =

    0.099859785917034
    0.293457816080161
    0.355664990937382
    0.251017407065422
```

As iteration method reaches the accuracy of e-5, it turns out the approximation method also has high accuracy as well.

Iteration	zx Approximation	difference
0.099859623	0.099859786	-1.63E-07
0.293458163	0.293457816	3.47E-07
0.355664614	0.355664991	-3.77E-07
0.251017601	0.251017407	1.94E-07

Script

```
I=speye(n,n);
AA=I-P*G*D;
x=GaussPivotScale(AA,e);
x=x/sum(x);
```

Gauss Elimination Code

```
function x = GaussPivotScale(A,b)
%
% GaussPivotScale uses Gauss elimination with partial
% pivoting and row scaling to solve the linear system
% Ax = b.
%
% x = GaussPivotScale(A,b) where
%
% A is the n-by-n coefficient matrix,
% b is the n-by-1 result vector,
%
% x is the n-by-1 solution vector.
n = length(b);
A = [A b]; % Augmented matrix
for k = 1:n-1,
M = max(abs(A(k:end, k:end-1)), [], 2);
% Find maximum for each row
a = abs(A(k:end, k)); % Find maximum for kth column
[~,I] = max(a./M);
% Find relative row with maximum ratio
I = I + k-1; % Adjust relative row to actual row
if I > k
    % Pivot rows
A([k I],:) = A([I k],:);
end
m = A(k+1:n, k)/A(k, k); % Construct multipliers
[Ak,M] = meshgrid(A(k,:), m); % Create mesh
A(k+1:n,:) = A(k+1:n,:) - Ak.*M;
end
Ab = A;
% Find the solution vector using back substitution
x = BackSub(Ab);
```

```
function x = BackSub(Ab)
%
% BackSub returns the solution vector of the upper
% triangular augmented matrix Ab using back substitution.
%
% x = BackSub(Ab) where
%
% Ab is the n-by-(n+1) augmented matrix,
%
% x is the n-by-1 solution vector.
n = size(Ab, 1);
for k = n:-1:1,
Ab(k,:) = Ab(k,:)./Ab(k, k); % Construct multipliers
Ab(1:k-1, n+1) = Ab(1:k-1, n+1)-Ab(1:k-1, k)*Ab(k, n+1);
% Adjust rows
end
x = Ab(:, end);
```

2. according to spirit of regression, here the residual is

$$\mathrm{R} = \sum_{\mathrm{i}} \frac{1}{\sigma_{\mathrm{i}}{}^{2}} (f_i - f(x_i, f_r, E_r, \Gamma))^2$$

$$\mathrm{f} = \frac{f_r}{(E - E_r)^2 + \frac{\Gamma^2}{4}}$$

I have to solve the equation

$$\frac{\partial \mathrm{R}}{\partial f_r} = 0, \qquad \frac{\partial \mathrm{R}}{\partial E_r} = 0, \quad \frac{\partial \mathrm{R}}{\partial \Gamma_r} = 0$$

Code:

```
%E in MeV, f, sigma in mb
E=0:25:200;
f=[10.6,16,45,83.5,52.8,19.9,10.8,8.25,4.7];
sigma=[9.34,17.9,41.5,85.5,51.5,21.5,10.8,6.29,4.14];
sigma2=sigma.*sigma;

X0 = [76950,75,3600]';
tol = 0.0001;

D = (E-X0(2)).^2+X0(3)/4;
f1 = sum((1./sigma2).*(X0(1)./D./D-f./D));
f2 = sum((1./sigma2).*(X0(1)./D./D./D-f./D./D).*(E-X0(2)));
f3 = sum((1./sigma2).*(X0(1)./D./D./D-f./D./D));
B = [f1;f2;f3];

f11 = sum((1./sigma2)./D./D);
f12 = sum((1./sigma2).*(4*X0(1).*(E-X0(2))./D./D./D - 2*f.*(E-
X0(2))./D./D));
f13 = sum((1./sigma2).*(f/4./D./D-1/2*X0(1)./D./D./D));
f21 = sum((1./sigma2)./D./D./D.*(E-X0(2)));
f22 = sum((1./sigma2).*(f./D./D-X0(1)./D./D./D + (E-
X0(2)).*(6*X0(1).*(E-X0(2))./D./D./D./D-4.*f.*(E-X0(2))./D./D./D)));
f23 = sum((1./sigma2).*(f/2./D./D./D-3*X0(1)/4./D./D./D./D).*(E-
X0(2)));
f31 = sum((1./sigma2)./D./D./D);
f32 = sum((1./sigma2).*(6*X0(1).*(E-X0(2))./D./D./D./D-4.*f.*(E-
X0(2))./D./D./D));
f33 = sum((1./sigma2).*(-3/4*X0(1)./D./D./D./D-f/2./D./D./D));
F = [f11,f12,f13;f21,f22,f23;f31,f32,f33];

dX = -F\B;
X1 = X0+dX;
n = 1;
while ( max(abs(dX./X0)) > tol )
    X0 = X1;
    D = (E-X0(2)).^2+X0(3)/4;
    f1 = sum((1./sigma2).*(X0(1)./D./D-f./D));
    f2 = sum((1./sigma2).*(X0(1)./D./D./D-f./D./D).*(E-X0(2)));
    f3 = sum((1./sigma2).*(X0(1)./D./D./D-f./D./D));
    B = [f1;f2;f3];

    f11 = sum((1./sigma2)./D./D);
    f12 = sum((1./sigma2).*(4*X0(1).*(E-X0(2))./D./D./D - 2*f.*(E-
X0(2))./D./D));
    f13 = sum((1./sigma2).*(f/4./D./D-1/2*X0(1)./D./D./D));
    f21 = sum((1./sigma2)./D./D./D.*(E-X0(2)));
    f22 = sum((1./sigma2).*(f./D./D-X0(1)./D./D./D + (E-
X0(2)).*(6*X0(1).*(E-X0(2))./D./D./D./D-4.*f.*(E-X0(2))./D./D./D)));
```

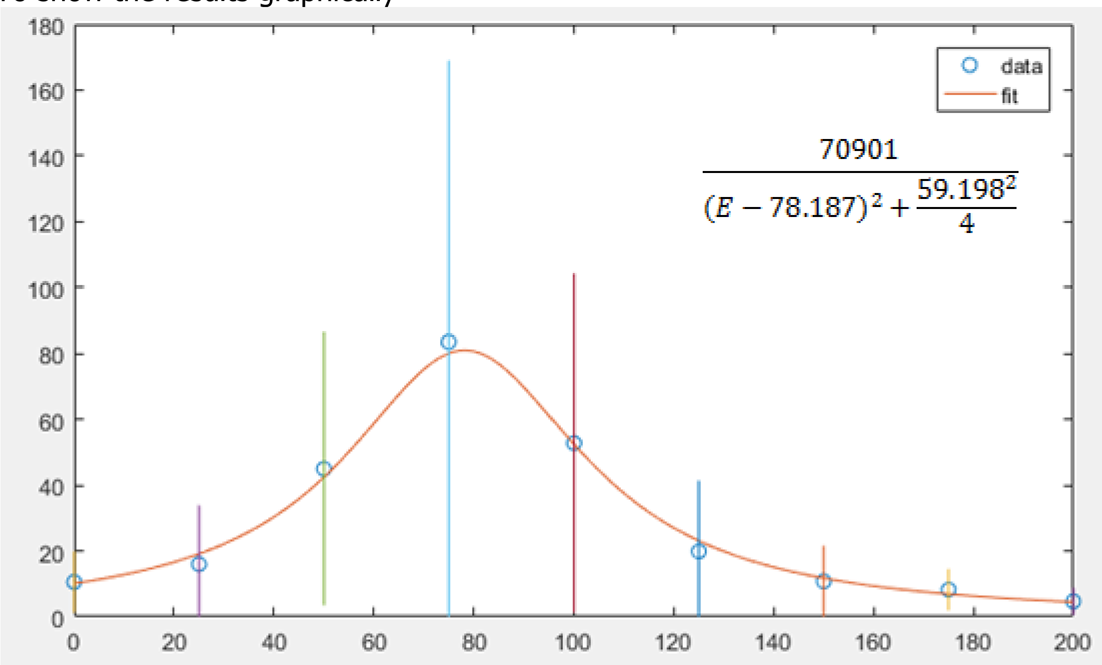
```
    f23 = sum((1./sigma2).*(f/2./D./D./D-3*X0(1)/4./D./D./D./D).*(E-
X0(2)));
    f31 = sum((1./sigma2)./D./D./D);
    f32 = sum((1./sigma2).*(6*X0(1).*(E-X0(2))./D./D./D./D-4.*f.*(E-
X0(2))./D./D./D));
    f33 = sum((1./sigma2).*(-3/4*X0(1)./D./D./D./D-f/2./D./D./D));
    F = [f11,f12,f13;f21,f22,f23;f31,f32,f33];

    dX = -F\B;
    X1 = X0+dX;
    n = 1+n;
end
```

here I set the termination condition of iteration to be the largest change(relative, absolute value) of $[f_r, E_r, \Gamma^2]$ smaller than tolerance, which I set as 0.0001.

and I set the initial value of $[f_r, E_r, \Gamma^2]$ to be [76950,75,3600] (which is determined from plot of data points)
after 35 iterations, $[f_r, E_r, \Gamma^2]$ meets the tolerance, and I get

f_r	70901
E_r	78.187
Γ	59.198

To show the results graphically



E_r means the energy when the cross section reaches its maximum.
Γ means the half peak breadth.
$\frac{4f_r}{\Gamma^2}$ is the maximum of f