

HW 2

Xinle Pang

Sep.23 2018

Question 1:

$D=(0.3333, 0.3333)$

Question 2:

The convergence process is listed as follows, while the elapsed time is 0.011359 seconds.

iter 1: $p(1) = 1.000000$, $p(2) = 1.000000$, $\text{norm}(f(x)) = 0.25222947$

iter 2: $p(1) = 1.595733$, $p(2) = 1.595733$, $\text{norm}(f(x)) = 0.00131883$

iter 3: $p(1) = 1.598864$, $p(2) = 1.598864$, $\text{norm}(f(x)) = 0.00003186$

iter 4: $p(1) = 1.598942$, $p(2) = 1.598942$, $\text{norm}(f(x)) = 0.00000001$

Question 3:

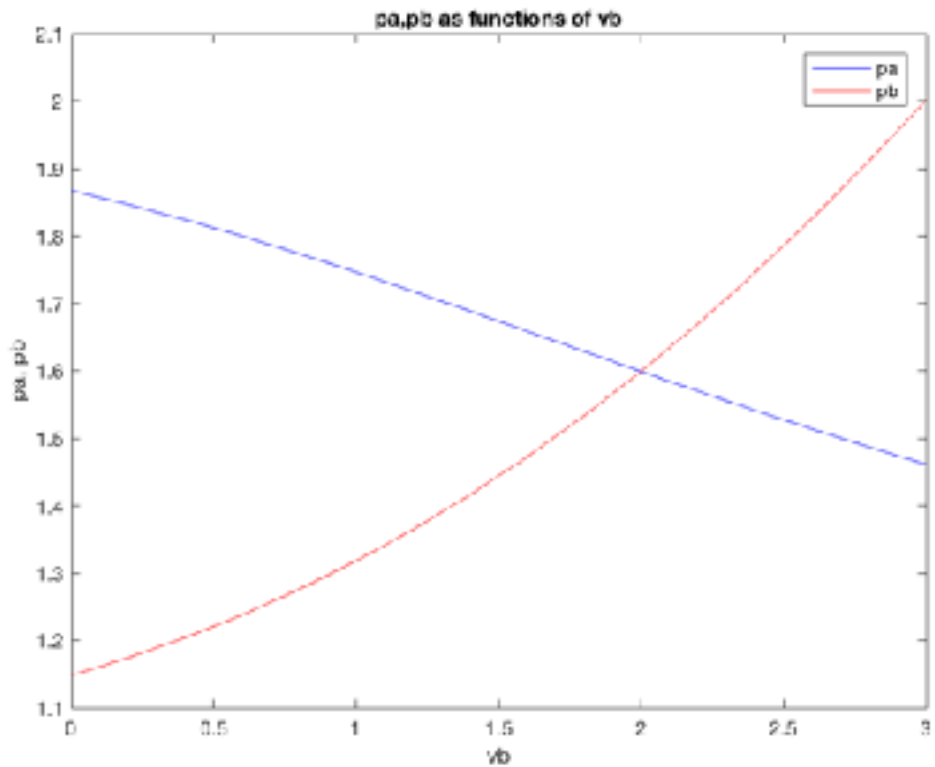
The elapsed time is 0.008439 seconds.

Question 4:

The elapsed time is 0.011076 seconds.

Questions 5:

I used Broyden's method, and the result is the following.



Code:

%HW2 Code

%% Question 1

```
clear;
v=[2,2]; p=[2,2];
[fval,D]=bertrand2(v,p);
display(D);
```

%% Question 2 Broyden's Method

```
clear;
p=[1; 1]; %initial guess
v=[2;2];
fVal = bertrand2(v,p); %initial value of FOC
f=@(p) bertrand2(v,p);
iJac = inv(myJac(f, p)); %initial Jacobian of FOC
```

%Broyden iterations:

```
tic
```

```

maxit = 100;
tol = 1e-6;
for iter = 1:maxit
    fnorm = norm(fVal);
    fprintf('iter %d: p(1) = %f, p(2) = %f, norm(f(x)) = %.8f\n', iter, p(1), p(2),
norm(fVal));
    if norm(fVal) < tol
        break
    end
    d = - (iJac * fVal);
    p = p+d;
    fOld = fVal;
    fVal = bertrand(p);
    u = iJac*(fVal - fOld);
    iJac = iJac + ( (d - u) * (d'*iJac) )/ (d'*u);
end
toc

```

%% Question 3 Secant Method

```

clear;
v=[2;2];

f1= @(p1,p2) p1*(1+exp(v(2)-p2))/(1+exp(v(1)-p1)+exp(v(2)-p2))-1;
f2= @(p1,p2) p2*(1+exp(v(1)-p1))/(1+exp(v(1)-p1)+exp(v(2)-p2))-1;

```

%Assign initial values

```

pa=1.2; paold=1;
pb=1.2; pbold=1;

```

% Secant iterations:

```

tol = 1e-8;
maxit = 100;

```

```

tic

```

```

for iter =1:maxit

```

```

    f1Val=f1(pa,pb); f2Val=f2(pa,pb);

```

```

    if abs(max(f1Val,f2Val)) < tol
        break
    end

```

else

%given pb, update the guess for pa using FOC1

f=@(p1) f1(p1, pb);

fold=f(paold);

fVal = f(pa);

paNew = pa - ((pa - paold) / (fVal - fold)) * fVal;

paold = pa;

pa = paNew;

%given updated pa, update pb using FOC2

f=@(p2) f2(pa, p2);

fold=f(pbold);

fVal = f(pb);

pbNew = pb - ((pb - pbold) / (fVal - fold)) * fVal;

pbold = pb;

pb = pbNew;

end

end

toc

%% Question 4

clear;

%initial price

p=[1; 1];

v=[2,2];

fVal = bertrand2(v,p);

%Iterations by rule: $p' = 1/(1-D(p))$

tic

ee=ones(length(p),1);

maxit = 100;

tol = 1e-6;

for iter = 1:maxit

fnorm = norm(fVal);

fprintf('iter %d: p(1) = %f, p(2) = %f, norm(f(x)) = %.8f\n', iter, p(1), p(2), norm(fVal));

if norm(fVal) < tol

break

```

    end
    [fval,D]=bertrand2(v,p);
    p = ee./(ee-D);
    fVal = bertrand2(v,p);
end
toc

```

%% Question 5

```

va=2;
vb=0:0.2:3;
pa=ones(length(vb),1); pb=ones(length(vb),1);

```

```

maxit = 100;
tol = 1e-6;

```

```

for i=1:length(vb)
    v=[va;vb(i)];

```

```

    %initial guess

```

```

    p=[1; 1];
    fVal = bertrand2(v,p);
    f=@(p) bertrand2(v,p);
    iJac = inv(myJac(f, p));

```

```

    %Broyden iterations:

```

```

    for iter = 1:maxit
        fnorm = norm(fVal);
        if norm(fVal) < tol
            break
        end
        d = - (iJac * fVal);
        p = p+d;
        fOld = fVal;
        fVal = bertrand2(v,p);
        u = iJac*(fVal - fOld);
        iJac = iJac + ( (d - u) * (d'*iJac) )/ (d'*u);
    end

```

```
pa(i)=p(1); pb(i)=p(2);
```

```
end
```

```
figure
```

```
plot(vb, pa, 'b', vb, pb, 'r');
```

```
title('pa,pb as functions of vb');
```

```
xlabel('vb'); ylabel('pa, pb');
```

```
legend('pa', 'pb');
```