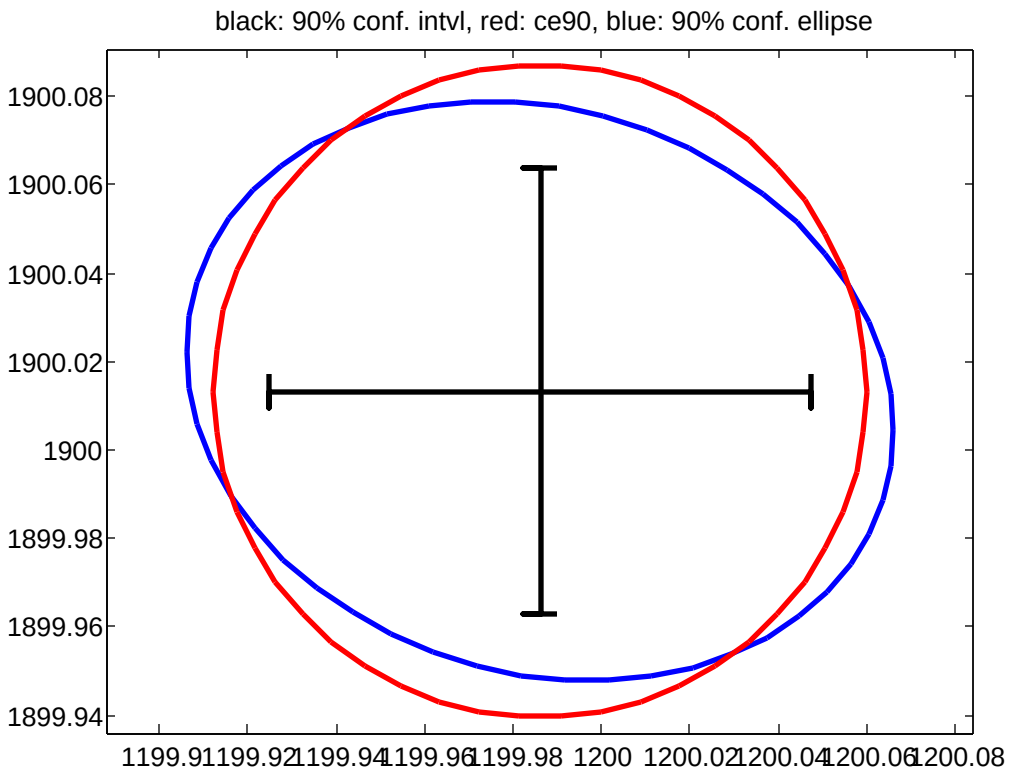
```
hw4
n =
     5
n0 =
     2
r =
     3
u =
     2
del =
          -9.8368251307025
          10.0482504354788
del =
        -0.177097051401564
       -0.0349143872193216
del =
     1.87388068704826e-005
    -1.92792887907436e-005
del =
    -6.13288526961868e-010
     1.85485433542491e-010
ans =
                       300                       900
                      1100                      1200
                      1600                       600
          1199.98609655609          1900.01331676916
s02 =
     1
v =
       -0.0369977920770414
         0.037998044653633
        -0.016132399637412
    -1.64328321596962e-005
    -2.43082499424549e-007
vsec =
       -0.0369977920770414
         0.037998044653633
        -0.016132399637412
         -3.38951494151079
       -0.0501393646458645
chitest =
          1.34409018860078
cv1 =
         0.215795282623898
cv2 =
          9.34840360449614
Sdd =
      0.00139000651118658     -0.000160938786719665
     -0.000160938786719665      0.000936322745366826
we pass 2-sided global test @ alpha=.05
z =
          1.64485362695147
x_half_width =
         0.0613247341412164
y_half_width =
         0.0503315196934075
x conf interval
xci =
          1199.92477182195          1200.04742129023
y conf interval
yci =
          1899.96298524946          1900.06364828885
lambda1 =
       0.00144129864797386
lambda2 =
      0.000885030608579551
V1 & length
ans =
         -0.95278140585244
          0.303657031306454
maj_length =
         0.0814703968491265
V2 & length
ans =
         -0.303657031306454
          -0.95278140585244
min_length =
         0.0638413390548588
ce90 =
         0.0738199263509244
theta =
          2.83306406597378
diary off
```


black: 90% conf. intvl, red: ce90, blue: 90% conf. ellipse
1900.08
1900.06
1900.04
1900.02
1900
1899.98
1899.96
1899.94
1199.91
1199.92
1199.94
1199.96
1199.98
1200
1200.02
1200.04
1200.06
1200.08

```
% hw4.m  4-nov-09
% solve data hw#4

X=[300;1100;1600;1210];
Y=[900;1200;600;1890];
d1=1345.40;
d2=707.08;
d3=1360.18;
th1d=33+51/60+30.6/3600;
th2d=25+13/60+55.1/3600;
degrad=180/pi;
th1=th1d/degrad;
th2=th2d/degrad;
sa=(10/3600)/degrad;
l=[d1;d2;d3;th1;th2];
n=5
n0=2
r=3
u=n0
B=zeros(n,u);
f=zeros(n,1);

w=[1/(0.05^2);
   1/(0.05^2);
   1/(0.05^2);
   1/(sa^2);
   1/(sa^2)];
W=diag(w);
iter=1;
keep_going=1;
while((keep_going==1) && (iter<10))
  res=distance2d(d1,4,1,X,Y);
  B(1,:)=res(1,3:4);
  f(1)=-res(2);
  res=distance2d(d2,4,2,X,Y);
  B(2,:)=res(1,3:4);
  f(2)=-res(2);
  res=distance2d(d3,4,3,X,Y);
  B(3,:)=res(1,3:4);
  f(3)=-res(2);
  res=angle2d(th1,4,2,1,X,Y);
  B(4,:)=res(1,3:4);
  f(4)=-res(2);
  res=angle2d(th2,4,3,2,X,Y);
  B(5,:)=res(1,3:4);
  f(5)=-res(2);
  N=B'*W*B;
  t=B'*W*f;
  del=inv(N)*t
  X(4)=X(4)+del(1);
  Y(4)=Y(4)+del(2);
  iter=iter+1;
  if(all(abs(del) < 1.0e-08))
    keep_going=0;
    end
  end

[X Y]

% post adjustment statistics

s02=1.0
v=f-B*del
vsec=v;
vsec(4)=v(4)*degrad*3600;
vsec(5)=v(5)*degrad*3600;
vsec
chitest=v'*W*v/s02
cv1=chi2inv(0.025,3)
cv2=chi2inv(0.975,3)
Sdd=inv(N)

% 90% c.i. x,y; 90% conf. ell. x,y; 90% conf. circ.

if((chitest > cv1) && (chitest < cv2))
  disp('we pass 2-sided global test @ alpha=.05');
  % do the error prop (P=0.90, alpha=0.10)
  % c.i. x-hat +/- z*sigma
  P=0.9;
  alpha=0.10;
  z=norminv(1-alpha/2,0,1)
  x_half_width=z*sqrt(Sdd(1,1))
  y_half_width=z*sqrt(Sdd(2,2))
```

```
  disp('x conf interval');
  xci=[X(4)-x_half_width  X(4)+x_half_width]
  disp('y conf interval');
  yci=[Y(4)-y_half_width  Y(4)+y_half_width]
  % now the conf ellipse
  [V,D]=eig(Sdd);
  if(D(2,2) > D(1,1))
    D=[D(2,2) 0; 0 D(1,1)];
    V=[V(:,2) V(:,1)];
    end
  lambda1=D(1,1)
  lambda2=D(2,2)
  disp('V1 & length');
  V(:,1)
  scale=chi2inv(P,2);
  maj_length=sqrt(lambda1*scale)
  disp('V2 & length');
  V(:,2)
  min_length=sqrt(lambda2*scale)
  % now conf circle
  ce90=cep2(P,Sdd)
  % ok now pictures
  theta=atan2(V(2,1),V(1,1))
  draw_ell3(X(4),Y(4),maj_length,min_length,theta,'b-',2);
  draw_cir3(X(4),Y(4),ce90,50,'r-',2);
  axis equal
  d=0.05*ce90;
  xc=X(4);
  yc=Y(4);
  xh=x_half_width;
  yh=y_half_width;
  px=[xc-xh; xc-xh; xc-xh; xc+xh; xc+xh; xc+xh];
  py=[yc+d; yc-d; yc; yc; yc-d; yc+d];
  plot(px,py,'k-','linewidth',2);
  py=[yc-yh; yc-yh; yc-yh; yc+yh; yc+yh; yc+yh];
  px=[xc+d; xc-d; xc; xc; xc-d; xc+d];
  plot(px,py,'k-','linewidth',2);
  shrink5;
  title('black: 90% conf. intvl, red: ce90, blue: 90% conf. ellipse');
else
  disp('we do not pass the 2-sided global test @ alpha=.05');
  end
```

```
% distance2d.m  30-oct-08
% compute distance condition equation
% returned vector is  [dc Fd dFdxi dFdyi dFdxj dFdyj]
function result=distance2d(d,i,j,X,Y)
xi=X(i);
yi=Y(i);
xj=X(j);
yj=Y(j);
Dij=sqrt((xj-xi)^2 + (yj-yi)^2);
dFdxi=(xj-xi)/Dij;
dFdyi=(yj-yi)/Dij;
dFdxj=-dFdxi;
dFdyj=-dFdyi;
F=d - Dij;
result=[Dij F dFdxi dFdyi dFdxj dFdyj];
```

note that the returned array has an extra element at the beginning, different from that specified in the instructions - it was added to return the value of the observation as computed from the current coordinates - this is useful for simulation and generation of synthetic data.

```
% angle2d.m  3-nov-08
function result = angle2d(a,i,j,k,X,Y)
xi=X(i);
yi=Y(i);
xj=X(j);
yj=Y(j);
xk=X(k);
yk=Y(k);
Dij_sq=(xj-xi)^2 + (yj-yi)^2;
Dik_sq=(xk-xi)^2 + (yk-yi)^2;
dFdxi= (yk-yi)/Dik_sq - (yj-yi)/Dij_sq;
dFdyi=-(xk-xi)/Dik_sq + (xj-xi)/Dij_sq;
dFdxj=(yj-yi)/Dij_sq;
dFdyj=-(xj-xi)/Dij_sq;
dFdxk=-(yk-yi)/Dik_sq;
dFdyk=(xk-xi)/Dik_sq;
ac=atan2(xk-xi,yk-yi) - atan2(xj-xi,yj-yi);
if(ac < 0)
  ac=ac + 2*pi;
  end

% ac
% degrad=180/pi;
% ac*degrad

Fa=a - ac;
result=[ac Fa dFdxi dFdyi dFdxj dFdyj dFdxk dFdyk];
```

see comments about distance2d.m - same comments apply here.