Appendix

```
load ifk.mat

[m,n] = size(d);
a = 0; b = 1;

n = m;

sig  = 1e-4;

dx = (b-a)/m;
g = @(x,y) x*exp(-x*y);

x = [];
for j = 1:n
    x = [x  a + (dx/2) + (j-1)*dx]; %form x
end
y = x;

G = zeros(m,n);

for i = 1:m
    for j = 1:n
        G(i,j) = g(x(j),y(i)).*dx; %form G
    end
end

% Get the singular values for the system matrix
%[U,S,V] = svd(G);
```

No.1 Use the L-curve to find a value for .

```
[U,sm,X,V,W] = cgsvd(G,eye(n));

%regtools
[reg_corner,rho,eta,reg_param] = l_curve(U,sm,d,'Tikh');
alpha = reg_corner
```

No.2 (a) Identify a value for δ and justify your choice.

```
delta = sig*sqrt(m)
```

b) Use your value for α from 1. to calculate .

```
malpha = @(alpha) inv(G'*G + (alpha^2)*eye(n))*G'*d;

discrepancy = norm(G*malpha(alpha) - d,2)^2
```

c) Implement a nonlinear solver such as fsolve in MATLAB to find a regularization parameter using the discrepancy principle.

```
f = @(alpha) norm(G*malpha(alpha) - d,2)^2 - delta; %non linear function
alph = fsolve(f,alpha)
```

No.3 (a) Identify a value for δreg and justify your choice.

```
deltareg = 20
```

(b) Use your values for α from 1 and 2c to calculate. Discuss if these choices of α satisfy the regularized discrepancy you chose in 3a.

```
reg_d = @(alpha) (norm(G*malpha(alpha) - d)^2)/(sig^2) +
(alpha^2)*norm(malpha(alpha))^2;

regd_1 = reg_d(alpha)

regd_2c = reg_d(alph)
```

(c) Implement a nonlinear solver such as fsolve in MATLAB to find a regularization parameter using the regularized discrepancy principle.

```
freg = @(alpha) (norm(G*malpha(alpha) - d)^2)/(sig^2) +
(alpha^2)*norm(malpha(alpha))^2 - deltareg;

al = fsolve(freg,alph)
```

No.4 Plot model parameter estimates using Tikhonov regularization with all three values of α on the same graph. Relate the shape of the different graphs to the corresponding values for α.

```
alpha = [alpha alph al];

m1 = malpha(alpha(1));
m2 = malpha(alpha(2));
m3 = malpha(alpha(3));

figure(2)
plot(m1,'-*'); hold on
```

```
plot(m2,'-o'); hold on
plot(m3,'-^')
legend('m_{\alpha_{1}}','m_{\alpha_{2}}','m_{\alpha_{3}}')
xlabel('i'); ylabel('Parameter Estimates')
title('Parameter estimates')
```

No.5 (i) Plot the resolution matrices from each value of α

```
Gp = @(alpha) inv(G'*G + (alpha^2)*eye(n))*G';

figure(2)
clf
colormap('gray')
subplot(2,2,1)
imagesc((Gp(alpha(1)))*G)
set(colorbar,'Fontsize',18);

xlabel('j')
ylabel('i')
title('Model resolution matrix \alpha_{1}')

subplot(2,2,2)
imagesc((Gp(alpha(2)))*G)
set(colorbar,'Fontsize',18);

xlabel('j')
ylabel('i')
title('Model resolution matrix \alpha_{2}')
subplot(2,2,3)
imagesc((Gp(alpha(3)))*G)
set(colorbar,'Fontsize',18);

xlabel('j')
ylabel('i')
title('Model resolution matrix \alpha_{3}')
```

(ii) Plot the diagonal elements of the resolution matrices on the same graph.

```
figure(3)
plot(diag(Gp(alpha(1))*G),'.','MarkerSize',20); hold on
plot(diag(Gp(alpha(2))*G),'*','MarkerSize',20); hold on
plot(diag(Gp(alpha(3))*G),'o','MarkerSize',20);
legend('\alpha_{1}','\alpha_{2}','\alpha_{3}',Location='best')
xlabel('i'); ylabel('Diagonal elements of the resolution matrix')
title('Diagonal elements of the resolution matrix')
```