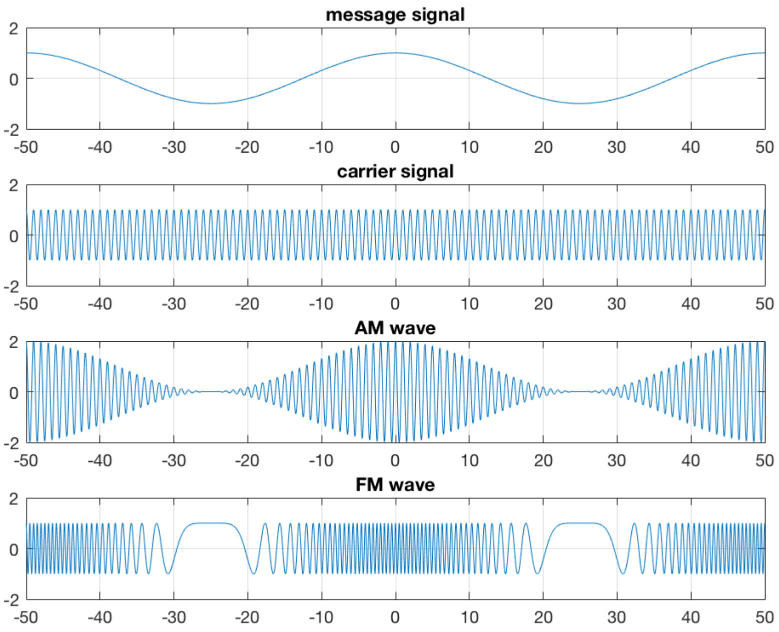
ENSC 180 - Assignment 8

Shengcong Zhou
Lab group 9
sza111@sfu.ca

1.

```
syms t

Ac=1; %carrier amplitude
fc=1; %carrier frequency
S=Ac*cos(2*pi*fc*t);%carrier signal shape

Am=1; %message amplitude
fm=0.02; %message frequency
%fm<<fc
M=Am*cos(2*pi*fm*t);%message signal shape

%AM modulation
S_AM=Ac*(1+M)*cos(2*pi*fc*t);
%FM modulation
kf=1; %frequency sensitivity
S_FM=Ac*cos(2*pi*fc*t + 2*pi*kf*Am*int(M,0,t));

TitleArray=["message signal","carrier signal","AM wave","FM wave"];
FunctionArray=[M,S,S_AM,S_FM];

for i=1:4
    subplot(4,1,i)
    fplot(FunctionArray(i))
    title(TitleArray(i))
    grid
    xlim([-50,50])
    ylim([-2,2])
end
```


message signal

carrier signal

AM wave

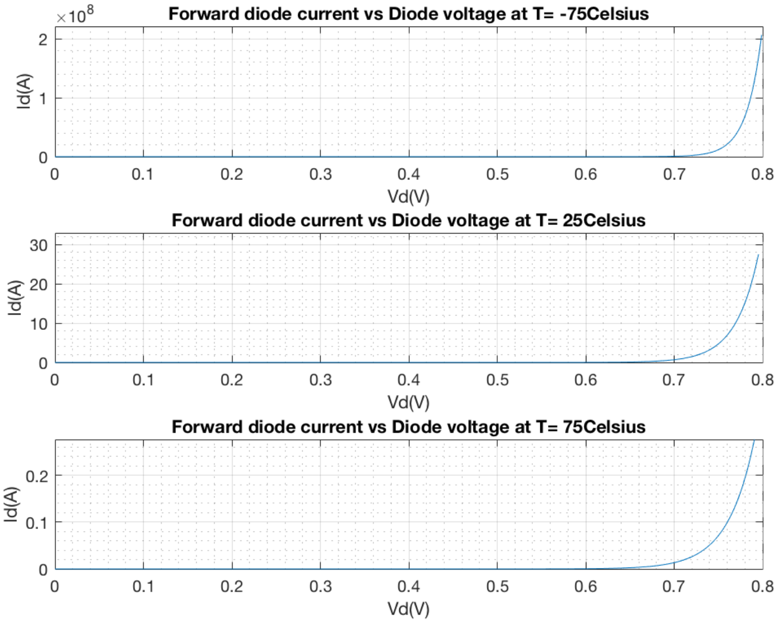
FM wave

As the amplitude of the message signal becomes more negative, the amplitude of AM wave decreases and the frequency of FM wave decreases.

As the amplitude of the message signal becomes more positive both AM amplitude and FM frequency increases.

2.

```
%plot three curves at three temps of voltage of a diode vs current of a diode
syms Vd

K = 1.38*10^-23; %in j/k
q = 1.6*10^-19; %in C
T=[-75 25 75]; %temperature in Kelvin
Is = 10^-12;%reverse current in A
n = 1;


for i=1:3
    Vt = K*(273+T(i))/q; %thermal voltage in V
    Id = Is*(exp(Vd/(n*Vt))-1);

    subplot(3,1,i)
    fplot(Id,[0,0.8])
    title(['Forward diode current vs Diode voltage at T= ',num2str(T(i)),'Celsius'])
    xlabel('Vd(V)')
    ylabel('Id(A)')
    grid
    grid minor
end
```

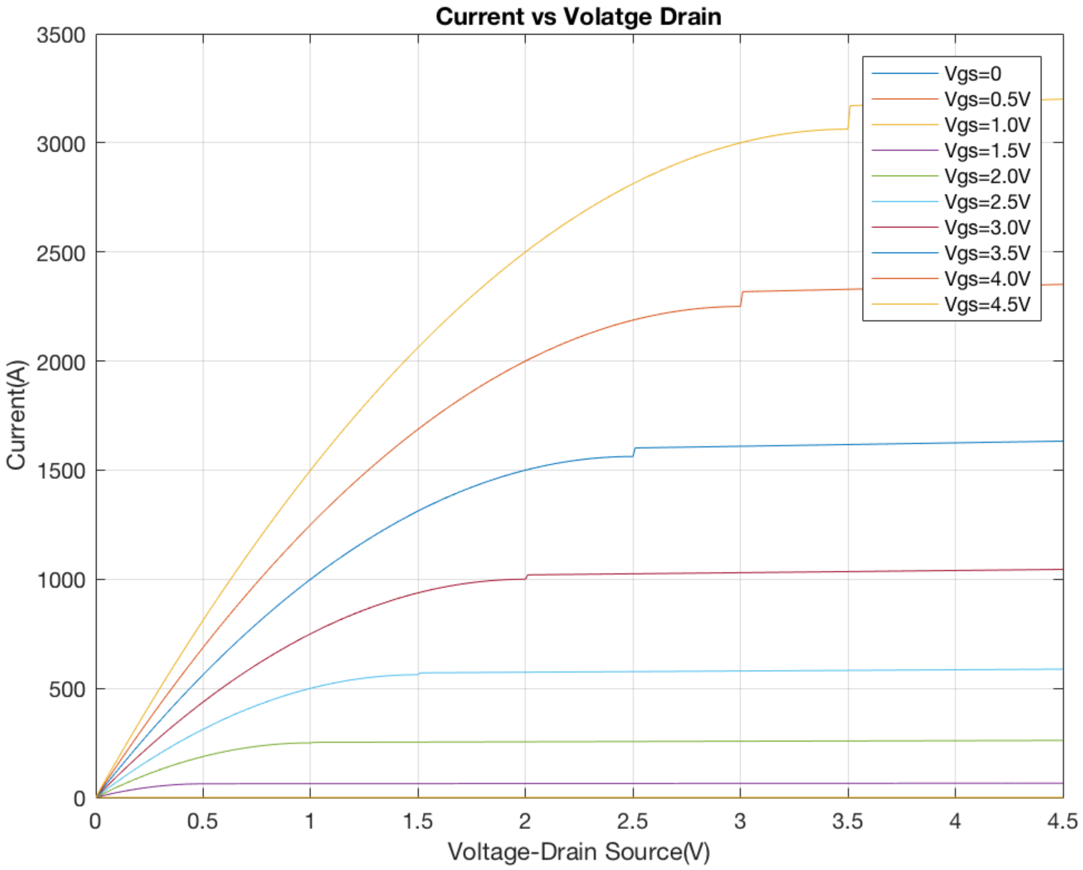


The current of diode at low temperature reaches much greater value than when it’s at higher temperature, this suggests that the resistance is much lower at low temperature at the same voltage. However this may not be true for all voltages since, current start rising at lower voltage for higher temperature.

3.

```
Kn = 100; %A/V^2
WL = 5; %unitless
Vt = 1; %volt
lambda = 0.01;

Vds = 0:0.01:5;
Vg = 0;

for i = 1:10
    ID = zeros;
    if i > 1
        Vg = Vg + 0.5;
    end
    for j = 1:501

        if Vg < Vt
            ID(j) = 0;
        else

            if Vds(j) <= Vg - Vt
                ID(j) = 0.5*Kn*(WL)*(2*(Vg-Vt)*Vds(j)-(Vds(j)^2));

            elseif Vds(j) > Vg - Vt
                ID(j) = 0.5*Kn*(WL)*((Vg-Vt)^2)*(1+lambda*Vds(j));
            end
        end
    end

    plot(Vds, ID)
    xlim([0,4.5])
    title('Linear: Current Volatge Drain')
    ylabel('Current A')
    xlabel('Voltage (Drain Source) V')
    grid
    grid minor

legend('Vgs=0','Vgs=0.5V','Vgs=1.0V','Vgs=1.5V','Vgs=2.0V','Vgs=2.5V','Vgs=3.0V','Vgs=3.5V'
,'Vgs=4.0V','Vgs=4.5V')
    hold on
end
grid
hold off
```

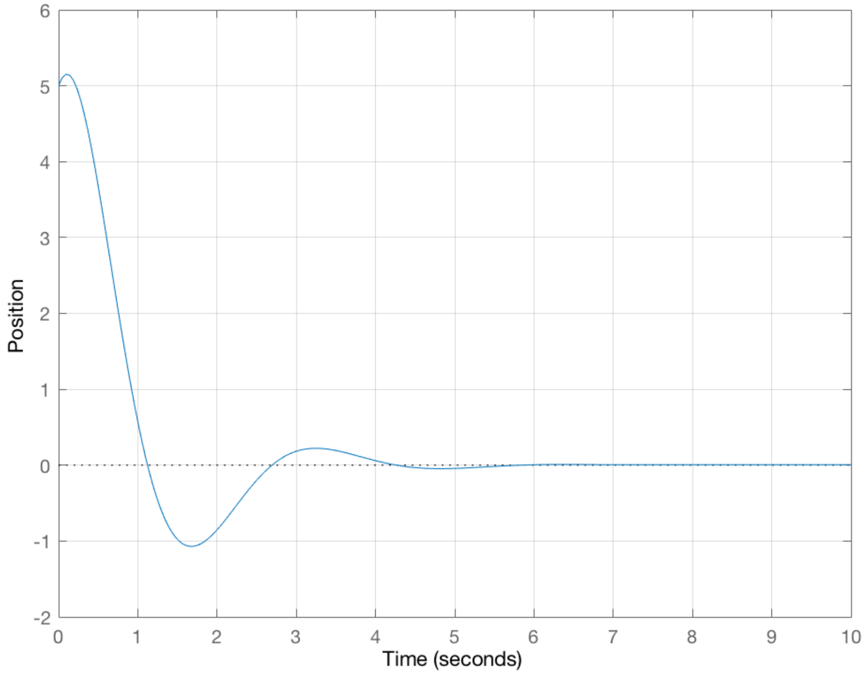
Because of the scaling some lines may not be easy to see



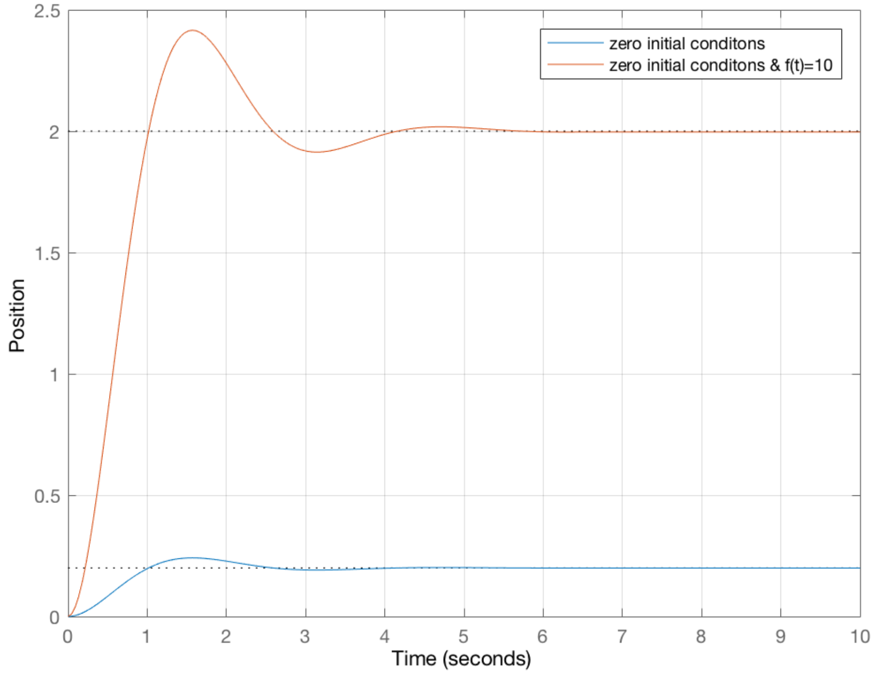
4.

```
A=[0 1;-5 -2];
B=[0;1];
C=[1 0];
D=0;

sys = ss(A,B,C,D);

%a
x0 = [5;3]; %initial position=5, initial velocity=3

figure
initial(sys,x0);
xlim([0 10])
title("Position of mass; x(0)=5, x'(0)=3")
xlabel('Time')
ylabel('Position')
grid

%b
opt = stepDataOptions('StepAmplitude',10);

figure
hold on
step(sys)
step(sys,opt)
xlim([0 10])
title('Position of mass; zero initial conditions & external excitation f(t)=10')
legend('zero initial conditons','zero initial conditons & f(t)=10’)
```

```
xlabel('Time')
ylabel('Position')
grid
hold off
```



Position of mass; x(0)=5, x'(0)=3

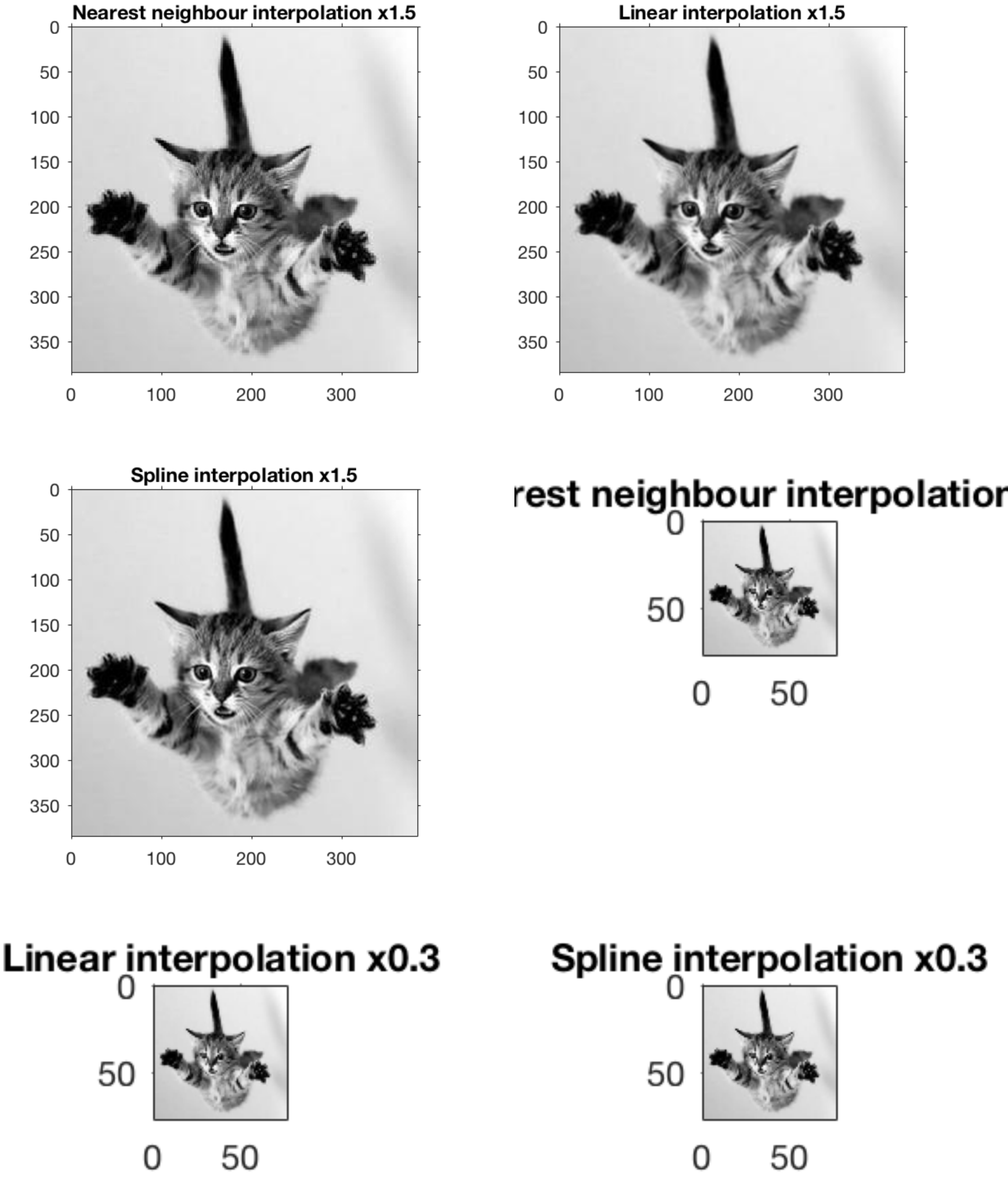


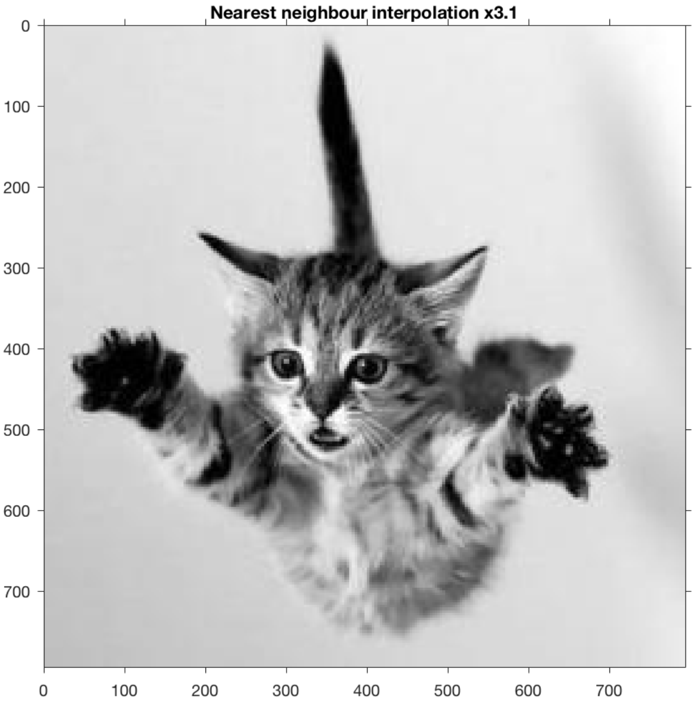
Position of mass; zero initial conditions & external excitation f(t)=10

5.

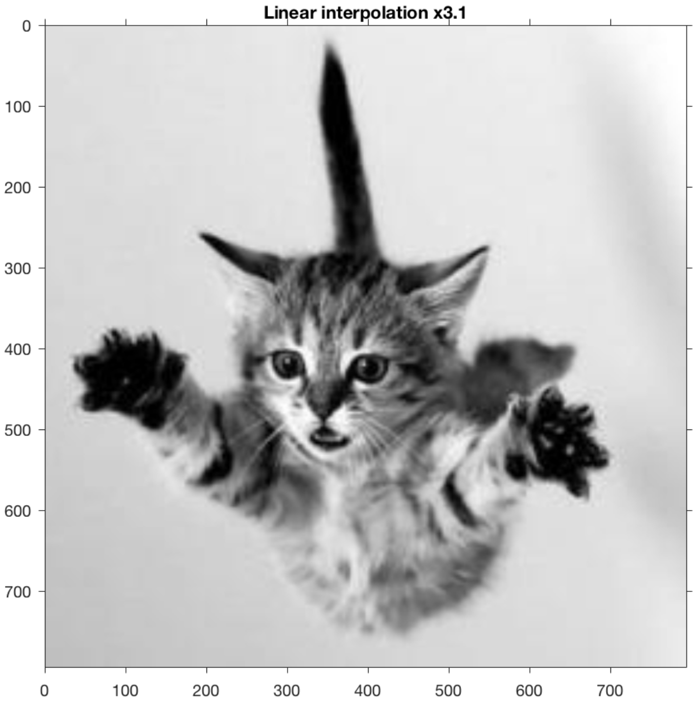
```
%function resize will take a file name of an image and a positive number
%as input arguments and scale the image by the number using nearest neighbour,
%bilinear, and bicubic interpolations
function resize(ImageName,Scalar)

%read specified image file and convert to double between 0 and 255
A=imread(ImageName);
A=double(A)/255;

% determine size of the image
[h1,w1]=size(A);

%determines the number of new rows and new columns
%where the minimum new image size is 1x1
h2=ceil(h1*Scalar);
w2=ceil(w1*Scalar);

%determine coordinates(indices) of original image pixels
[x,y]=meshgrid(1:h1,1:w1);

%divide the original image with the new image size
xx=linspace(1,h1,h2);
yy=linspace(1,w1,w2);

%determine coordinates of query points
[xnew,ynew]=meshgrid(xx,yy);

%nearest neighbour interpolation
B=interp2(x,y,A,xnew,ynew,'nearest');

%linear interpolation
C=interp2(x,y,A,xnew,ynew,'linear');

%Spline interpolation
D=interp2(x,y,A,xnew,ynew,'spline');

%plot
figure
imshow(B)
title(['Nearest neighbour interpolation x',num2str(Scalar)])
axis([0,w2,0,h2])
axis on
figure
imshow(C)
title(['Linear interpolation x',num2str(Scalar)])
axis([0,w2,0,h2])
axis on
figure
imshow(D)
title(['Spline interpolation x',num2str(Scalar)])
axis([0,w2,0,h2])
axis on
```

———————————————————————————————————————

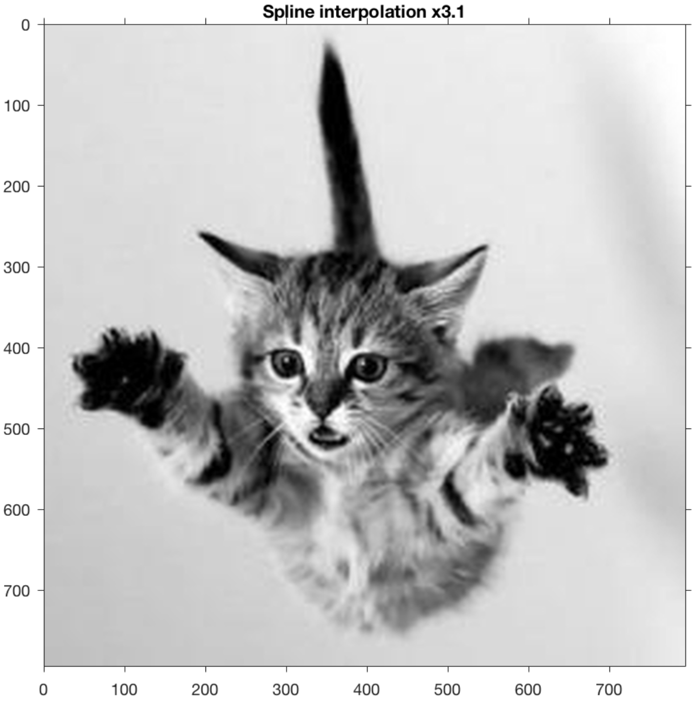
Nearest neighbour interpolation x1.5
0
50
100
150
200
250
300
350
0
100
200
300
Linear interpolation x1.5
0
50
100
150
200
250
300
350
0
100
200
300
Spline interpolation x1.5
0
50
100
150
200
250
300
350
0
100
200
300
rest neighbour interpolatior
0
50
0
50
Linear interpolation x0.3
0
50
0
50
Spline interpolation x0.3
0
50
0
50



Nearest neighbour interpolation x3.1



Linear interpolation x3.1



Spline interpolation x3.1