CS/ECE/ME 532
Homework 7

1. **Training neural networks**. Below is some simple matlab code for training a neual network with one hidden layer. (You may translate this to python if you wish.)

```
p = 2;
n = 1e4;
% generate training data
X = rand(n,p)-.5;
Y1 = sum(X.^2,2)>.1;
Y2 = 5*X(:,1).^3 > X(:,2);
Y = [Y1 Y2];
figure(1);clf;
subplot(121);
scatter(X(:,1),X(:,2),20,Y1,'filled');
title('training data, label 1');
axis image;colorbar;colormap jet; set(gca,'fontsize',18)
subplot(122);
scatter(X(:,1),X(:,2),20,Y2,'filled');
title('training data, label 2');
axis image;colorbar;colormap jet; set(gca,'fontsize',18)


%%
Xb = [ones(n,1) X];
q = size(Y,2);
M = 2;
V = randn(M+1,q);
W = randn(p+1,M);
alpha = .1;

for epoch = 1:10;
   ind = randperm(n);
   for i = ind;
      % forward prop
      H = logsig([1 Xb(i,:)*W]); % 1 x M+1
      Yhat = logsig(H*V); % 1 x q

      % backprop
      delta = (Yhat-Y(i,:)).*Yhat.*(1-Yhat); % 1 x q
      Vnew = V-alpha*H'*delta;

      gamma = (delta*V(2:end,:)').*H(2:end).*(1-H(2:end)); % 1 x M
```

```
      Wnew = W-alpha*Xb(i,:)'*gamma;

      V = Vnew;
      W = Wnew;

   end
   epoch
end
%%
% final predicted labels
H = logsig([ones(n,1) Xb*W]); % n x M+1
Yhat = logsig(H*V); % n x q

figure(2);clf;
subplot(121);scatter(X(:,1),X(:,2),20,Yhat(:,1),'filled');
title('learned labels, label 1');
axis image;colorbar;colormap jet; set(gca,'fontsize',18)
subplot(122);scatter(X(:,1),X(:,2),20,Yhat(:,2),'filled');
title('learned labels, label 2');
axis image;colorbar;colormap jet; set(gca,'fontsize',18)


figure(3);clf;
subplot(121);scatter(X(:,1),X(:,2),20,1*(Yhat(:,1)>.5),'filled');
title('thresholded learned labels, label 1');
axis image;colorbar;colormap jet; set(gca,'fontsize',18)
subplot(122);scatter(X(:,1),X(:,2),20,1*(Yhat(:,2)>.5),'filled');
title('thresholded learned labels, label 2');
axis image;colorbar;colormap jet; set(gca,'fontsize',18)
```

a) Run the code. How does it perform? Are the learned labels close to the original lables?

b) Why do we use `Xb` instead of `X` ? What if we use `X` instead?

c) Explain the use of the "2"s in the expression for gamma.

d) Try increasing the number of epochs to 100. What effect does this have?

e) Try increasing the number of hidden nodes to 3; what happens? What happens if you use 4 hidden nodes? Can you explain *why* four hidden nodes performs so much differently from two?