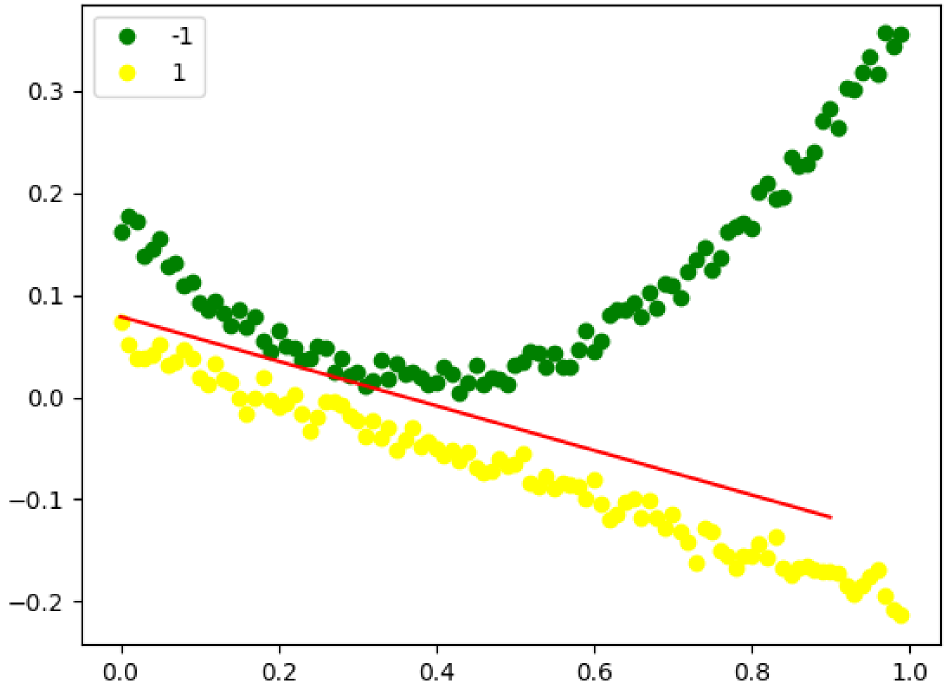
Problem 1

As we can see, the classification is good.

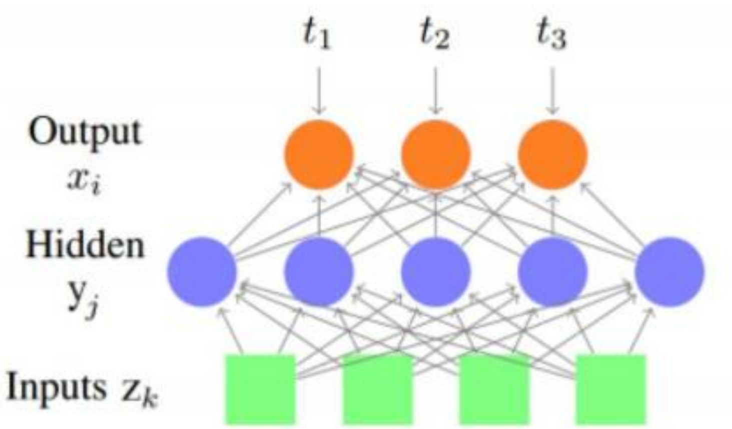


The python code is:

```
import numpy as np
import matplotlib.pyplot as plt
def load_data():
    import scipy.io
    Dataset_MATLAB = scipy.io.loadmat('data3.mat')
    Data = Dataset_MATLAB['data']
    X_Data = [Data[i][:-1] for i in range(len(Data))]
    Y_Data = [Data[i][-1] for i in range(len(Data))]
    return X_Data,Y_Data

def perceptron(X, Y, eta):
    w=np.ones((X.shape[1],1))
    X=np.matrix(X)
    Y=np.matrix(Y).T
    b=1
    round = 0
    all_correct=False
    while True:
```

```
            Errors = 0
            for i in range(X.shape[0]):
                XX = X[i]
                YY = Y[i]
                if      ((w.T*XX.T+b).getA()[0][0]      <0      and      YY.getA()[0][0]      >0)      or
((w.T*XX.T+b).getA()[0][0] >0 and YY.getA()[0][0] <0):
                    w+=eta*(XX.T*YY)
                    b+=eta*YY
                    Errors += 1
            if Errors == 0:
                break
            round += 1
            print(round, Errors)
        return w, b

def draw(w, b, X, Y):
        x_points = np.arange(0, 1, 0.1)
        y_points = np.array([-(w[0]*i+b)/w[1] for i in x_points]).flatten()
        x_0 = []
        x_1 = []
        for i in range(X.shape[0]):
            if Y[i] == -1:
                x_0.append(X[i])
            else:
                x_1.append(X[i])
        x_0 = np.array(x_0)
        x_1 = np.array(x_1)
        plt.plot(x_0[:, 0], x_0[:, 1], 'o', color='green', label='-1')
        plt.plot(x_1[:, 0], x_1[:, 1], 'o', color='yellow', label='1')
        plt.plot(x_points, y_points, '-', color='red')
        plt.legend()
        plt.show()

if __name__ == "__main__":
        X_Data, Y_Data = load_data()
        X = np.array(X_Data)
        Y = np.array(Y_Data)
        w,b=perceptron(X, Y, 0.01)
        draw(w, b, X, Y)
```

Problem 2



(a)

$$E = -\sum_i (t_i \log(x_i) + (1-t_i)\log(1-x_i))$$

$$x_i = \frac{1}{1+e^{-s_i}}$$

$$\frac{\partial E}{\partial \omega_{ji}} = \frac{1}{N}\sum_i \left(\frac{\partial E_i^n}{\partial x_i^n}\right)\left(\frac{\partial x_i^n}{\partial \omega_{ji}}\right) = \frac{1}{N}\sum_i \left(\frac{\partial E_i^n}{\partial x_i^n}\right)\left(\frac{\partial x_i^n}{\partial s_i^n}\right)\left(\frac{\partial s_i^n}{\partial \omega_{ji}}\right)$$

$$\frac{\partial E_i}{\partial x_i} = \frac{\partial(-(t_i \log(x_i) + (1-t_i)\log(1-x_i)))}{\partial x_i} = -\frac{t_i}{x_i} - ((1-t_i)\cdot\frac{1}{1-x_i}\cdot(-1)) = -\frac{t_i}{x_i} + \frac{1-t_i}{1-x_i}$$

$$\frac{\partial x_i}{\partial s_i} = \frac{\partial \frac{1}{1+e^{-s_i}}}{\partial s_i} = \frac{\partial \frac{e^{s_i}}{1+e^{s_i}}}{\partial s_i} = \frac{(e^{s_i})'\cdot(1+e^{s_i}) - e^{s_i}\cdot(1+e^{s_i})'}{(1+e^{s_i})^2} = \frac{e^{s_i}}{(1+e^{s_i})^2} = \frac{e^{s_i}}{1+e^{s_i}}\cdot\frac{1}{1+e^{s_i}} = x_i\cdot(1-x_i)$$

$$s_i = \sum_j y_j \omega_{ji}$$

$$\frac{\partial s_i}{\partial \omega_{ji}} = y_i$$

$$\frac{\partial E_i}{\partial \omega_{ji}} = \frac{\partial E_i}{\partial x_i}\cdot\frac{\partial x_i}{\partial s_i}\cdot\frac{\partial s_i}{\partial \omega_{ji}} = (-\frac{t_i}{x_i} + \frac{1-t_i}{1-x_i})\cdot x_i\cdot(1-x_i)\cdot y_i = (x_i - t_i)\cdot y_i$$

$$y_j = \frac{1}{1+e^{-p_j}},\ p_j = \sum_k z_k m_{kj}$$

$$\frac{\partial E}{\partial m_{kj}} = -\sum_i (\frac{t_i}{x_i} - \frac{1-t_i}{1-x_i})\cdot\frac{\partial x_i}{\partial s_i}\cdot\frac{\partial s_i}{\partial y_j}\cdot\frac{\partial y_j}{\partial p_j}\cdot\frac{\partial p_j}{\partial m_{kj}} = -\sum_i (x_i - t_i)\cdot y_j\cdot(1-y_j)\cdot\omega_{ji}\cdot z_k$$

(b)

$$E = -\sum_i t_i \log(x_i)$$

$$x_i = \frac{e^{s_i}}{\sum_c e^c}$$

$$y_j = \frac{1}{1+e^{-p_j}}$$

$$s_i = \sum_j y_j \omega_{ji}$$

$$p_j = \sum_k z_k m_{kj}$$

$$\frac{\partial x_i}{\partial s_i} = x_i(1-x_i) \quad i = j$$

$$\frac{\partial x_i}{\partial s_j} = -x_i x_j \qquad i \neq j$$

$$\frac{\partial s_i}{\partial \omega_{ji}} = y_j$$

$$\frac{\partial E}{\partial \omega_{ji}} = -\sum_{n\neq i} \frac{t_n}{x_n} \cdot \frac{\partial x_n}{\partial s_i} \cdot \frac{\partial s_i}{\partial \omega_{ji}} - \frac{t_i}{x_i} \cdot \frac{\partial x_i}{\partial s_i} \cdot \frac{\partial s_i}{\partial \omega_{ji}}$$

$$= -\sum_{n\neq i} \frac{t_n}{x_n} \cdot (-x_n x_i) \cdot y_j - \frac{t_i}{x_i} \cdot x_i(1-x_i) \cdot y_j$$

$$=(\sum_{n\neq i} t_n x_i - t_i(1-x_i)) \cdot y_j$$

$$\frac{\partial E}{\partial m_{kj}} = -\sum_n \frac{\partial E}{\partial x_n} \cdot \sum_t \frac{\partial x_n}{\partial s_t} \cdot \frac{\partial s_t}{\partial y_j} \cdot \frac{\partial y_j}{\partial p_j} \cdot \frac{\partial p_j}{\partial m_{kj}}$$

$$= \sum_n \frac{t_n}{x_n} \sum_t \frac{\partial x_n}{\partial s_t} \cdot \frac{\partial s_t}{\partial y_j} \cdot y_j(1-y_j) z_k$$

$$= \sum_n \frac{t_n}{x_n} \cdot (x_n(1-x_n)\omega_{jn} + \sum_{t\neq n}(-x_n x_t)\omega_{jt}) \cdot y_j(1-y_j) z_k$$

$$= \sum_n t_n \cdot ((1-x_n)\omega_{jn} - \sum_{t\neq n} x_t \omega_{jt}) \cdot y_j(1-y_j) z_k$$

Problem 3

$$H = -\sum_{k=1}^{N} p_k \log p_k$$

$$Set\ \ f(p) = \sum_{k=1}^{N} p_k \log p_k,\ \ where\ \ G(p) = \sum_{k=1}^{N} p_k - 1 = 0$$

$$L(p) = f(p) - \lambda G(p) = \sum_{k=1}^{N} p_k \log p_k - \lambda(\sum_{k=1}^{N} p_k - 1)$$

$$\begin{cases} \dfrac{\partial L(p)}{\partial p} \Rightarrow \quad \log p_k + 1 - \lambda = 0 \\ G(p) = 0 \Rightarrow \quad \sum_{k=1}^{N} p_k = 1 \end{cases}$$

$$\text{So we can get} \begin{cases} \lambda = 1 - \log N \qquad p_k = \dfrac{1}{N} \\ L(p_0) = f(p_0) = \log N \end{cases}$$

Problem 4

The VC dimension of axis-aligned square is 3. We can see that when there are only three points, whether all three points are collinear or not, there will always be an axis-aligned square that can meet the condition. But If there are four points, the situation shown in the figure will appear, and no square meets the condition. Therefore the VC dimension of axis-aligned square is 3.