Q. $J(\theta) = -l(\theta) + \lambda \|\theta\|^2$

$-l(\theta) = \sum \log(1 + \exp(-y_i \theta^T \tilde{x}_i))$

where $\theta \in \mathbb{R}^{d+1}$ & $\tilde{x}_i \in \mathbb{R}^{d+1}$

& $\|\theta\|^2 = \theta^T \theta$

$\therefore \nabla J(\theta) = \nabla(-l(\theta)) + \lambda \nabla(\theta^T \theta)$

$\nabla(-l(\theta)) = \sum \frac{\exp(-y_i \theta^T \tilde{x}_i)}{1 + \exp(-y_i \theta^T \tilde{x}_i)} \cdot -y_i \tilde{x}_i$ $\left\{ \begin{array}{l} \text{Note where} \\ \tilde{x}_i \in \mathbb{R}^{d+1} \end{array} \right\}$.

$\nabla(\theta^T \theta) = 2\theta$

Now

$\nabla J(\theta) = \sum \frac{1}{1 + \exp(-y_i \theta^T \tilde{x}_i)} \cdot (-y_i \tilde{x}_i) + 2\lambda\theta$

We know $\sigma = \frac{1}{1 + \exp(-y_i \theta^T \tilde{x}_i)}$

$\therefore 1 - \sigma = \frac{1}{1 + \exp(-y_i \theta^T \tilde{x}_i)} \cdot \exp(-y_i \theta^T \tilde{x}_i)$

$\therefore \nabla J(\theta) = \sum \cancel{\sigma} (1 - \sigma) \cdot (-y_i \tilde{x}_i) + 2\lambda\theta$

Now $\nabla^2 J(\theta) = \frac{\partial}{\partial \theta^T} \left[\sum (1 - \sigma)(-y_i \tilde{x}_i) + 2\lambda\theta \right]$

For $x^{(i)}$, we can write.

$\{\nabla^2 J(\theta)\}_{i^{th}} = \frac{\partial}{\partial \theta^T} \sum (1 - \sigma)(-y_i x_i^{(i)}) + (\nabla 2\lambda\theta)_{i^{th}\,\text{row}}$

$= \cancel{-\frac{1}{(1+}}$

$= + \frac{1}{(1 + \exp(-y_i \theta^T \tilde{x}_i))^2} \cdot \exp(-y_i \theta^T \tilde{x}_i) (y_i^2 \tilde{x}_i^T x_i^{(i)})$

$+ \nabla(2\lambda\theta)_{i\,\text{row}}$

$x_i = d \times 1 \times 2$

DATE

$$\left\{\nabla^2 J(\theta)\right\}_{i^{th}\,row} = \sum \sigma(1-\sigma)\, y_i^2\, \tilde{x}_i\, x_i^{(i)} + 2\lambda (\nabla\theta)_{i^{th}\,row}$$

$$H = \nabla^2 J(\theta) = \sum \sigma(1-\sigma)\, y_i^2\, \tilde{x}_i\, \tilde{x}_i^T + 2\lambda I$$

where $I \in (d+1) \times (d+1)$

& $\tilde{x}_i \in \mathbb{R}^{d+1}$

$\therefore \tilde{x}_i x_i^T \in (d \times 1) \times (1 \times (d+1))$

$\in (d+1) \times (d+1)$

For $J(\theta)$ to be convex, the hessian should be PSD

ie $a^T H a \geq 0 \quad \forall a \in \mathbb{R}^{d+1}$

or $LHS = a^T H a$

$= \sum \sigma(1-\sigma)\, y_i^2 . (a^T \tilde{x}_i) . (a^T \tilde{x}_i)^T$

$+ 2\lambda a^T a$

$$= \underbrace{\sum \sigma(1-\sigma)}_{\text{always +ve}}\, \underbrace{y_i^2}_{\text{+ve}}\, \underbrace{\|a^T \tilde{x}_i\|^2}_{\text{+ ve}} + 2\lambda \underbrace{\|a\|^2}_{\text{+ve.}}$$

$\therefore LHS$ will ≥ 0 where $\lambda \geq 0$

or H is PSD

$\|^{ly}$ H is PD when $\lambda > 0$

```
# -*- coding: utf-8 -*-
"""
Created on Wed Sep 28 18:36:08 2016

@author: Akash Rastogi
"""

import numpy as np
import scipy.io as sio
import matplotlib.pyplot as plt
import os

os.getcwd()
mnist_49_3000 = sio.loadmat('mnist_49_3000.mat')

x = mnist_49_3000['x']
y = mnist_49_3000['y']
d,n = x.shape
i = 2000 #Index of the image to be visualized
plt.imshow( np.reshape(x[:,i], (int(np.sqrt(d)),int(np.sqrt(d)))))
plt.show()

A = np.ones(n)
A = A[None,:]
xNew = np.vstack((A, x))

xTrain = xNew[:,:2000]
yTrain = y[:,:2000]
dTrain ,nTrain = xTrain.shape

#sigmoid as per our defination as 1/(1+exp(-(yi) * theta.transpose() * (xi))
def sigmoid(y, x, theta):
   aMat = (np.matrix(theta)) * np.matrix(x)
   aArray = aMat.A1
   b = aArray[None,:]
   c = np.exp(- y * b)
   oneArray = np.ones(x.shape[1])[None,:]
   sig = oneArray/(oneArray - c)
   return sig


k = sigmoid(yTrain, xTrain, theta)
def gradient(y, x, theta, lamda):
   var1 = (1-sigmoid(y, x, theta))
```

```
    var2 = var1 * (-y)
    var3 = var2 * x
    term1 = var3.sum(axis = 1)
    term2 = 2*lamda * np.ones(x.shape[0])
    grad =  term1 + term2
    return grad

def hessian(y, x, theta, lamda):
    hVar1 = np.matrix(xTrain) * np.matrix(np.transpose(xTrain))
    hVar2 = np.squeeze(np.asarray(hVar1))
    return hess

lamda = 10
theta = np.ones(d)
```