ME598/494 Homework 3

1. (40 Points, problem from Professor M. Kokkolaras, McGill University) Vapor-liquid equilibria data are correlated using two adjustable parameters A_{12} and A_{21} per binary mixture. For low pressures, the equilibrium relation can be formulated as:

$$\begin{aligned} p &= x_1 \exp\left(A_{12}\left(\frac{A_{21}x_2}{A_{12}x_1 + A_{21}x_2}\right)^2\right) p_1^{sat} \\ &+ x_2 \exp\left(A_{21}\left(\frac{A_{12}x_1}{A_{12}x_1 + A_{21}x_2}\right)^2\right) p_2^{sat}. \end{aligned} \tag{1}$$

   Here the saturation pressures are given by the Antoine equation

$$\log_{10}(p^{sat}) = a_1 - \frac{a_2}{T + a_3}, \tag{2}$$

   where $T = 20(°\text{C})$ and $a_{1,2,3}$ for a water - 1,4 dioxane system is given below.

	a_1	a_2	a_3
Water	8.07131	1730.63	233.426
1,4 dioxane	7.43155	1554.679	240.337

   The following table lists the measured data. Recall that in a binary system $x_1 + x_2 = 1$.

x_1	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
p	28.1	34.4	36.7	36.9	36.8	36.7	36.5	35.4	32.9	27.7	17.5

   Estimate A_{12} and A_{21} using data from Table 1: (1) Formulate the least square problem; (2) solve using your own gradient descent or Newton's implementation; (3) solve using Matlab functions "lsqnonlin" or "lsqcurvefit".

2. (30 Points) Download the data homework3data.mat. The data contains a set of topologically optimal brackets. Each row of X represents a bracket structure and y the angle of the point load. See figure. Build a predictive model using y as the input and X the output through the Matlab *Neural Net Fitting* App under *Math, Statistics, and Optimization* section of the App tab. How will you tell if your predictions are good?

   Notes: When you use the Nerual Net Fitting app, make sure to check the *matrix rows* box because in our data, each rows is a data point, i.e., we have 100 data points.

3. (20 Points, Optional for MAE494) Logistic regression is commonly used to approximate systems with binary outputs, e.g., the result of an election, the outcome of a drug, the

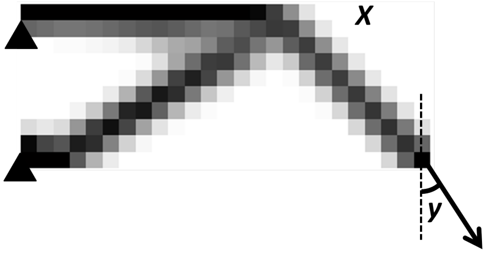

Figure 1: Problem 2 data illustration: Each row of X contains a flattened matrix that represents a topology; each element of y represents the angle of the fixed point load. The illustrated topology has the minimal compliance (most stiffness) for the given load under a fixed volume constraint.

purchase of a product, or many other discrete decision making of human beings. The mathematical form of a logistic regression model is as follows:

$$p(y; \mathbf{x}, \theta) = \frac{1}{1 + \exp(-y\theta^T \mathbf{x})}, \tag{3}$$

where y is the output that takes either 1 or -1, $\mathbf{x}$ are the input variables (covariates), θ are the unknown parameters. Given a dataset $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$, the likelihood of θ can be written as

$$L(\theta; D) = \prod_{i=1}^{N} \frac{1}{1 + \exp(-y_i \theta^T \mathbf{x}_i)}. \tag{4}$$

Is the maximum likelihood estimate of θ unique?

Hint: To obtain the maximum likelihood estimate of θ, one needs to minimize the **negative log**-likelihood function. To show that the solution is unique, you need to show that the negative log-likelihood function has a positive definite Hessian.

4. (10 Points) Please go through this tutorial on creating a metamodel using a deep convolutional neural network to predict Young's modulus of sandstone structures. There is an instruction on installing Keras and Tensorflow. Please attach your results.

   Note: To open *jupyter notebook*, go to the command line, change directory to the downloaded folder, and type in "jupyter notebook". This should open the notebook. Then run each block of the code to get the results.

5. (20 Points, bonus) For the predictive model built in Problem 2, compute the compliance of all predicted topologies under the corresponding loads. Compare these compliance values with the ground truth (i.e., using X). What do you observe? Explain the causes of the issues you identify.

   The compliance c of a topology x given load angle y can be calculated using the following Matlab code (from the 88-line topopt code.):

```matlab
nelx = 25;
nely = 10;
penal = 3;
%% Material properties
E0 = 1;
Emin = 1e-9;
nu = 0.3;
%% PREPARE FINITE ELEMENT ANALYSIS
A11 = [12  3 -6 -3;  3 12  3  0; -6  3 12 -3; -3  0 -3 12];
A12 = [-6 -3  0  3; -3 -6 -3 -6;  0 -3 -6  3;  3 -6  3 -6];
B11 = [-4  3 -2  9;  3 -4 -9  4; -2 -9 -4 -3;  9  4 -3 -4];
B12 = [ 2 -3  4 -9; -3  2  9 -2;  4  9  2  3; -9 -2  3  2];
KE = 1/(1-nu^2)/24*([A11 A12;A12' A11]+nu*[B11 B12;B12' B11]);
nodenrs = reshape(1:(1+nelx)*(1+nely),1+nely,1+nelx);
edofVec = reshape(2*nodenrs(1:end-1,1:end-1)+1,nelx*nely,1);
edofMat = repmat(edofVec,1,8)+repmat([0 1 2*nely+[2 3 0 1] -2 ...
    -1],nelx*nely,1);
iK = reshape(kron(edofMat,ones(8,1))',64*nelx*nely,1);
jK = reshape(kron(edofMat,ones(1,8))',64*nelx*nely,1);
% DEFINE LOADS AND SUPPORTS
F = zeros(2*(nely+1)*(nelx+1),1);
F(2*(nely+1)*(nelx+1)-1,1) = sin(y/180*pi);
F(2*(nely+1)*(nelx+1),1) = cos(y/180*pi);
U = zeros(2*(nely+1)*(nelx+1),1);
fixeddofs = union([1:2:2*(nely+1)],[2*(nelx+1)*(nely+1)]);
alldofs = [1:2*(nely+1)*(nelx+1)];
freedofs = setdiff(alldofs,fixeddofs);
sK = reshape(KE(:)*(Emin+x(:)'.^penal*(E0-Emin)),64*nelx*nely,1);
K = sparse(iK,jK,sK); K = (K+K')/2;
U(freedofs) = K(freedofs,freedofs)\F(freedofs);
ce = reshape(sum((U(edofMat)*KE).*U(edofMat),2),nely,nelx);
c = sum(sum((Emin+x.^penal*(E0-Emin)).*ce));
```

You can visualize x by the following code: `image(reshape(x(:),10,25)*255);`.