Adaptive Computation & Machine Learning
Assignment #2
Due: March 21, 2018, 20h00

Question 1 K-Nearest Neighbour Classification {25 Marks}

Consider the following training and testing data sets where x_is are features and ys are class labels. Soft copies of these data sets (`knn_training_data.txt` and `knn_testing_data.txt` respectively for training and testing data sets) can be found in the attachment.

Training Data Set

Record #	x_1	x_2	x_3	y
1	3.12	-3.14	1.93	0
2	0.93	9.43	-0.66	0
3	1.36	4.24	2.13	0
4	4.00	0.46	1.36	0
5	5.20	3.51	-1.25	0
6	4.41	-0.07	2.04	0
7	-1.58	10.84	2.55	0
8	3.18	7.23	-1.07	0
9	4.64	-3.01	2.68	0
10	2.45	2.97	0.20	0
11	3.90	1.73	1.60	0
12	3.41	5.40	-2.52	0
13	-2.65	10.14	-1.33	0
14	2.69	6.80	-0.40	0
15	2.32	-2.63	3.15	0
16	2.81	11.30	-3.34	0
17	0.50	10.22	-1.10	0
18	1.10	7.44	0.41	0
19	3.09	-2.92	2.22	0
20	0.50	5.53	1.78	0
21	-0.27	3.27	-3.56	1
22	-1.86	-0.85	2.54	1
23	0.64	3.76	-4.47	1
24	-1.59	-2.16	1.71	1
25	-2.07	0.17	-1.01	1
26	-6.42	9.53	0.02	1
27	-1.24	-1.72	-0.53	1
28	0.89	2.26	-3.10	1
29	-2.13	1.57	-0.08	1
30	-4.21	4.73	-0.49	1
31	-4.23	4.92	-0.49	1
32	-0.37	2.89	-3.65	1
33	-6.52	9.60	-0.25	1
34	-0.71	-4.44	2.22	1
35	0.76	0.29	-1.65	1
36	-4.02	-8.30	12.56	1
37	-3.02	-0.67	2.71	1
38	-1.82	-9.04	9.02	1
39	-0.90	2.03	-2.37	1
40	-1.56	-2.21	4.26	1

Testing Data Set

Record #	x_1	x_2	x_3	y
1	2.39	9.80	-3.04	0
2	-2.70	12.20	0.68	0
3	-1.62	0.81	8.16	0
4	0.52	-3.26	3.09	0
5	3.82	10.93	-4.01	0
6	2.99	7.20	-1.22	0
7	4.65	-4.83	3.46	0
8	-1.25	10.20	2.18	0
9	4.22	9.46	-4.93	0
10	-1.42	11.65	-0.06	0
11	3.03	2.29	2.11	0
12	1.16	6.40	1.55	0
13	1.10	0.64	5.99	0
14	-2.25	11.53	2.59	0
15	3.78	-2.88	2.26	0
16	0.27	9.22	-3.29	0
17	2.82	2.86	0.20	0
18	5.81	5.01	-2.24	0
19	3.26	-4.46	3.80	0
20	3.69	-3.95	4.26	0
21	-2.26	-4.74	6.35	1
22	-2.69	-0.10	0.62	1
23	-0.91	-7.90	6.78	1
24	-2.08	1.28	-0.64	1
25	0.14	-1.09	1.00	1
26	-4.14	2.77	0.68	1
27	1.45	3.61	-4.06	1
28	-0.82	3.38	-3.67	1
29	-4.08	2.92	0.87	1
30	-2.07	1.05	-0.46	1
31	-0.30	2.48	-2.95	1
32	-4.15	7.12	-0.08	1
33	0.22	-0.52	-0.40	1
34	-1.75	-5.82	5.87	1
35	1.16	3.91	-4.55	1
36	-2.70	1.63	0.84	1
37	0.90	3.37	-4.50	1
38	1.64	4.25	-4.90	1
39	-3.58	0.46	2.35	1
40	0.66	-0.05	-0.19	1

Table 1: Training and Testing Data Sets

For the following questions on K-Nearest Neighbor (K-NN) classifier, complete the tables to show your calculations in terms of closest record number(s) from training data set, corresponding distance(s) and label(s). The final output of the K-NN will be entered in Prediction section. The records should be sorted in ascending order of their distances from the query data point. Use Euclidean distance in search for nearest neighbors. Examples for the predictions of 1-NN and 3-NN for $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2.99 \\ 7.20 \\ -1.22 \end{bmatrix}$ are given below. Use two decimals in your answers.

1-NN

Record #	Distance from the record	Label of the record
8	0.24	0
Prediction		0

3-NN

Record #	Distance from the record	Label of the record
8	0.24	0
14	0.96	0
12	2.26	0
Prediction		0

Q.1.1{5 Marks} What is the prediction of the 1-NN classifier for $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4.65 \\ -4.83 \\ 3.46 \end{bmatrix}$?

Record #	Distance from the record	Label of the record
Prediction		

Q.1.2{5 Marks} What is the prediction of the 3-NN classifier for $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1.45 \\ 3.61 \\ -4.06 \end{bmatrix}$?

Record #	Distance from the record	Label of the record
Prediction		

Q.1.3{5 Marks} What is the prediction of the 5-NN classifier for $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -0.30 \\ 2.48 \\ -2.95 \end{bmatrix}$?

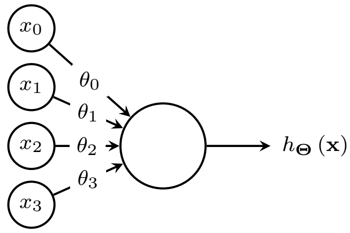
Record #	Distance from the record	Label of the record
Prediction		

Q.1.4{10 Marks} Using the above training and testing data sets given in Table 1, compute training and test error rates for 5-NN, 7-NN, 9-NN, 11-NN and 13-NN classifiers to complete the following table where train and test error rates for 1-NN and 3-NN classifiers are provided for you.

K-NN	Training Error Rate	Test Error Rate
1-NN	0.00	0.05
3-NN	0.00	0.07
5-NN		
7-NN		
9-NN		
11-NN		
13-NN		

Question 2 Logistic Regression {40 Marks}

Consider the following logistic regression model where $x_0 = 1.0$ and x_1, x_2 and x_3 are input features as defined in Table 1.



Let $\mathbf{x} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} \in \mathbb{R}^4$ be the input feature vector, $\mathbf{\Theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} \in \mathbb{R}^4$ be the parameter vector, and

$$h_{\mathbf{\Theta}}(\mathbf{x}) = \frac{1}{1 + e^{-\beta \mathbf{\Theta}^T \mathbf{x}}} = \frac{1}{1 + e^{-\beta(\theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3)}}$$

be the output of the logistic regression model where $\beta \in \mathbb{R}$ is an arbitrary real number.

Q.2.1{5 Marks} What is the range of $h_{\mathbf{\Theta}}(\mathbf{x})$? For what value(s) of β this classifier cannot learn from an arbitrary data set?

Q.2.2{20 Marks} For a given training set $\left\{\left(\mathbf{x}^{(n)}, y^{(n)}\right); n = 1, \ldots, N\right\}$, $\mathbf{x}^{(n)} = \begin{bmatrix} x_0^{(n)} \\ x_1^{(n)} \\ x_2^{(n)} \\ x_3^{(n)} \end{bmatrix} \in \mathbb{R}^4$, $y^{(n)} \in \{0, 1\}$, and cost function $J(\mathbf{\Theta}) = \frac{1}{4N} \sum_{n=1}^{N} \left(h_{\mathbf{\Theta}}\left(\mathbf{x}^{(n)}\right) - y^{(n)}\right)^4$, derive closed form parameter update rules for θ_k using batch gradient descent algorithm, i.e.,

$$\theta_k \leftarrow \theta_k - \alpha \frac{\partial J(\mathbf{\Theta})}{\partial \theta_k}, \forall k = \{0, \ldots, 3\}.$$

Q.2.3{15 Marks} Suppose that $\mathbf{\Theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} 0.63155343 \\ -0.69559537 \\ -0.37831336 \\ -0.4340919 \end{bmatrix}$ is obtained after running batch gradient descent learning algorithm for $\beta = 1$. The output of logistic regression model for an input $\mathbf{x}^{(n)}$ is mapped to a binary output (1 or 0) according to the following equation

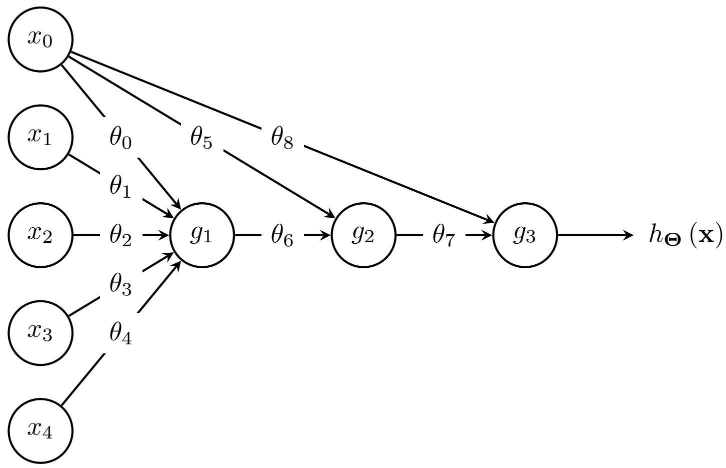
$$\begin{array}{ll} 1 & \text{for} \quad h_{\mathbf{\Theta}}\left(\mathbf{x}^{(n)}\right) > 0.5; \\ 0 & \text{for} \quad h_{\mathbf{\Theta}}\left(\mathbf{x}^{(n)}\right) \leq 0.5. \end{array}$$

Given the above information and the testing data in Table 1, complete the following table.

Testing record #	Classifier Output $h_{\mathbf{\Theta}}\left(\mathbf{x}^{(n)}\right)$	Final Output
5	0.01	0
10	0.06	0
15	0.13	0
20		
25		
30		
35		
40		

Question 3 Artificial Neural Networks {35 Marks}

Consider the following 4-layer (including input layer) ANN model where $\mathbf{x} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \in \mathbb{R}^5$ is input vector, θ_js are network parameters, g_ks are neuron activation functions, and $h_{\mathbf{\Theta}}(\mathbf{x})$ is the network output (or hypothesis).



Let $g_1(z) = \frac{1}{1+e^{-z}}$ and $g_2(z) = g_3(z) = z$ for an input $z \in \mathbb{R}$.
Q.3.1{10 Marks} What is the output $h_{\mathbf{\Theta}}(\mathbf{x})$ of the ANN in terms of x_is, θ_js and g_ks?

Q.3.2{25 Marks} The parameters of $h_{\mathbf{\Theta}}(\mathbf{x})$ are to be learned using a training set $\left\{\left(\mathbf{x}^{(n)}, y^{(n)}\right)\right\}_{n=1}^{N}$ of size N, where each training sample $\left(\mathbf{x}^{(n)}, y^{(n)}\right)$ is a tuple of a feature vector $\mathbf{x}^{(n)}$ with a label $y^{(n)}$ of positive class ($y^{(n)} = 1$) or negative class ($y^{(n)} = 0$). The parameters are iteratively learned by minimizing a cost function ($J(\mathbf{\Theta})$) between the real labels $y^{(n)}$s and the predicted labels $h_{\mathbf{\Theta}}\left(\mathbf{x}^{(n)}\right)$s resulted from the hypothesis evaluation, i.e.,

$$J(\mathbf{\Theta}) = \frac{1}{2N}\sum_{n=1}^{N}\left(h_{\mathbf{\Theta}}\left(\mathbf{x}^{(n)}\right) - y^{(n)}\right)^2 = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{2}\left(h_{\mathbf{\Theta}}\left(\mathbf{x}^{(n)}\right) - y^{(n)}\right)^2 = \frac{1}{N}\sum_{n=1}^{N} J(\mathbf{\Theta})^{(n)}.$$

As a minimizer of the cost function, gradient descent algorithm with a learning rate $\alpha \in (0, 1]$ can be employed to obtain iterative minimizers according to

$$\theta_j \leftarrow \theta_j - \alpha\frac{\partial J(\mathbf{\Theta})}{\partial \theta_j}.$$

Derive closed form equations of iterative minimizers only for θ_8, θ_7, θ_6 and θ_5s.