COMP/INDR 421/521 HW03: Multiclass Multilayer Perceptron

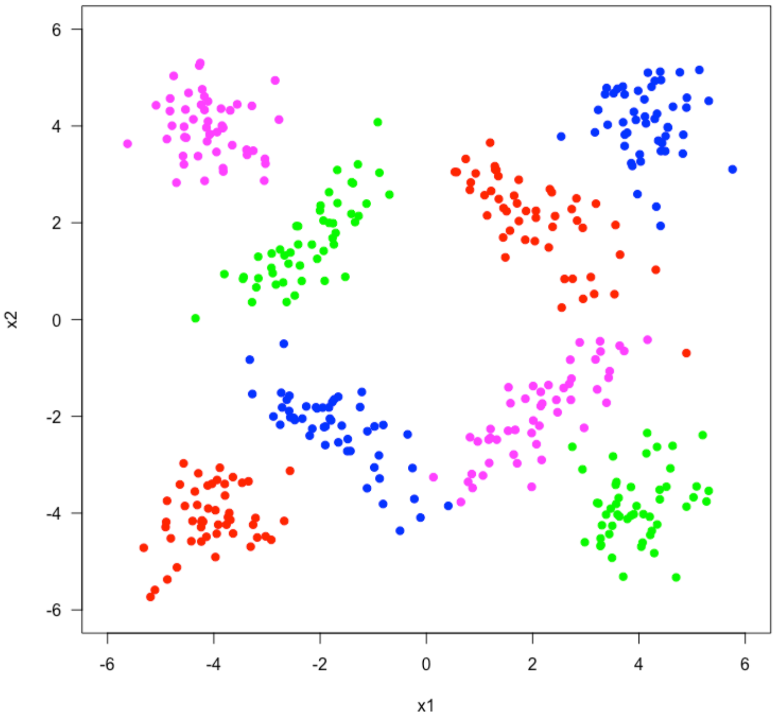
Deadline: November 3, 2017, 11:59 PM

In this homework, you will implement a multiclass multilayer perceptron algorithm in R, Matlab, or Python. Here are the steps you need to follow:

1. Generate random data points from eight bivariate Gaussian densities with the following parameters:

$$
\begin{aligned}
\mu_1 &= \begin{bmatrix} +2.0 \\ +2.0 \end{bmatrix}, & \Sigma_1 &= \begin{bmatrix} +0.8 & -0.6 \\ -0.6 & +0.8 \end{bmatrix}, & N_1 &= 50, & y_1 &= 1 \\
\mu_2 &= \begin{bmatrix} -4.0 \\ -4.0 \end{bmatrix}, & \Sigma_2 &= \begin{bmatrix} +0.4 & +0.0 \\ +0.0 & +0.4 \end{bmatrix}, & N_2 &= 50, & y_2 &= 1 \\
\mu_3 &= \begin{bmatrix} -2.0 \\ +2.0 \end{bmatrix}, & \Sigma_3 &= \begin{bmatrix} +0.8 & +0.6 \\ +0.6 & +0.8 \end{bmatrix}, & N_3 &= 50, & y_3 &= 2 \\
\mu_4 &= \begin{bmatrix} +4.0 \\ -4.0 \end{bmatrix}, & \Sigma_4 &= \begin{bmatrix} +0.4 & +0.0 \\ +0.0 & +0.4 \end{bmatrix}, & N_4 &= 50, & y_4 &= 2 \\
\mu_5 &= \begin{bmatrix} -2.0 \\ -2.0 \end{bmatrix}, & \Sigma_5 &= \begin{bmatrix} +0.8 & -0.6 \\ -0.6 & +0.8 \end{bmatrix}, & N_5 &= 50, & y_5 &= 3 \\
\mu_6 &= \begin{bmatrix} +4.0 \\ +4.0 \end{bmatrix}, & \Sigma_6 &= \begin{bmatrix} +0.4 & +0.0 \\ +0.0 & +0.4 \end{bmatrix}, & N_6 &= 50, & y_6 &= 3 \\
\mu_7 &= \begin{bmatrix} +2.0 \\ -2.0 \end{bmatrix}, & \Sigma_7 &= \begin{bmatrix} +0.8 & +0.6 \\ +0.6 & +0.8 \end{bmatrix}, & N_7 &= 50, & y_7 &= 4 \\
\mu_8 &= \begin{bmatrix} -4.0 \\ +4.0 \end{bmatrix}, & \Sigma_8 &= \begin{bmatrix} +0.4 & +0.0 \\ +0.0 & +0.4 \end{bmatrix}, & N_8 &= 50, & y_8 &= 4
\end{aligned}
$$

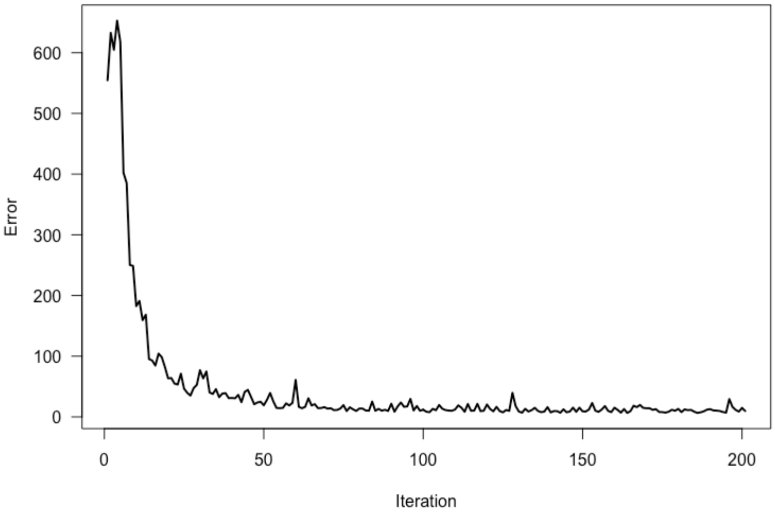
   Your data points should be similar to the following figure.



2. Learn a multilayer perceptron using the softmax function at the output layer for this multiclass classification problem. You can use the following learning parameters.

```
eta <- 0.1
epsilon <- 1e-3
H <- 20
max_iteration <- 200
```

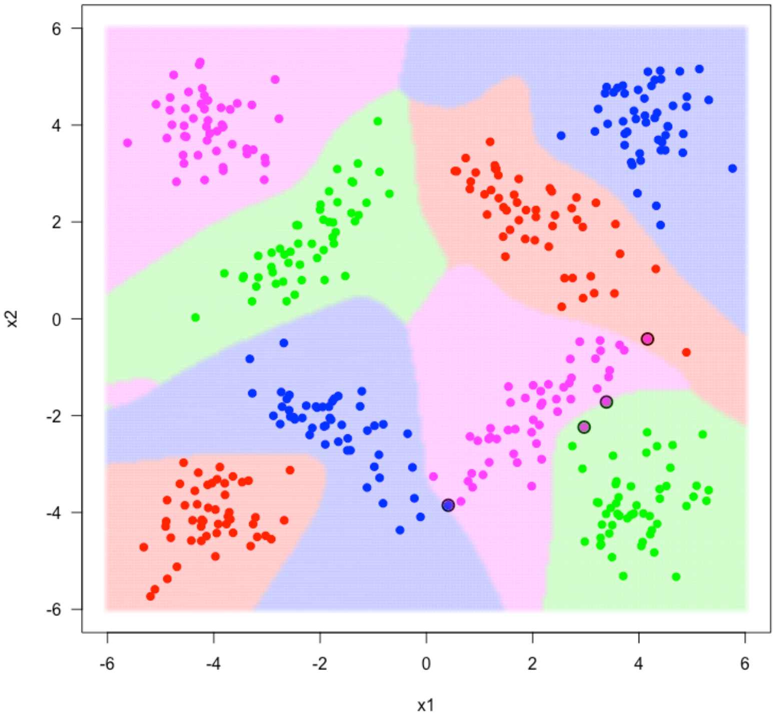
3. Draw the objective function values throughout the iterations. Your figure should be similar to the following figure.



4. Calculate the confusion matrix for the data points in your training set using the multilayer perceptron you will learn from the previous step. Your confusion matrix should be similar to the following matrix.

```
##            y_truth
## y_predicted   1   2   3   4
##           1 100   0   0   1
##           2   0 100   0   2
##           3   0   0  99   0
##           4   0   0   1  97
```

5. Draw your decision boundaries you will calculate using the multilayer perceptron from the previous step together with data points and clearly mark misclassified data points. Your figure should be similar to the following figure.



What to submit: You need to submit your source code in a single file (.R file if you are using R, .m file if you are using Matlab, or .py file if you are using Python) and a short report explaining your approach (.doc, .docx, or .pdf file). You will put these two files in a single zip file named as ***STUDENTID.zip***, where ***STUDENTID*** should be replaced with your 7-digit student number.

How to submit: E-mail the zip file you created to mehmetgonen@ku.edu.tr with the subject line ***Intro2MachineLearningHW03***. Please follow the exact style mentioned for the subject line and do not send a zip file named as ***STUDENTID.zip***. Submissions that do not follow these guidelines will not be graded.

Late submission policy: Late submissions will not be graded.

Cheating policy: Very similar submissions will not be graded.