**Homework 03: Multilayer Perceptron for Multiclass Discrimination**

I wrote Multilayer Perceptron for Multiclass Discrimination using R programming language by following the steps below.

1. I read file **“h3\_images.csv”** using the **images\_data\_read\_file** variable.
2. I read file **“h3\_labels.csv”** using the **label\_data\_read\_file** variable.
3. I split the data into two as training and test data.
4. I read file **“initial\_V.csv”** using the **initial\_V\_read\_file**.
5. I read file **“initial\_W.csv”** using the **initial\_W\_read\_file**.
6. I split the data into two as training and test data.
7. I using softmax function, sigmoid function and safelog function.
8. Y label value is vector[1] converted vector[1,5]. I using below;

For example code: one-of-K-encoding

Y\_train\_truth <- matrix (0, N, K)

Y\_train\_truth[cbind(1:N,train\_label)]<-1

Y\_test\_truth <- matrix (0, N, K)

Y\_test\_truth[cbind(1:N,test\_label)]<-1

1. Hidden nodes calcuator (Z = X \* W). I using sigmoid function.

cbind(1,Z) : 1 value is a constant

For example code:

Z <- sigmoid (cbind(1,Z) %\*% W)

1. Output nodes calculator (Z = Z \* V). I using sofmax function.

cbind(1,Z) : 1 value is a constant

For example code:

Z <- softmax (cbind(1,Z) %\*% W)

1. Every each iteration sofmax and sigmoid function using initial\_V, initial\_W and objective\_values changed.
2. The learning time will be up to the number of iterations or the epsilon value.

Epsilon <- 0.0001 or max\_iteration <- 500

1. The confusion matrix was calculated at the end of learning period.

Şeyhmus Aydoğdu