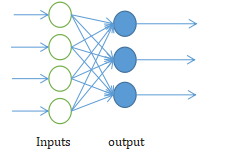
# CISC 452 assignment 3 report

Zili Luo 20001744

###### LVQ models

4-input unit fully connected to 3 output unit as the LVQ model for LVQ1 and 3 input for LVQ2

Inputs output

The output function for LVQ models are the eculidian\_distance between input and the codebook vector

the initial value for the LVQ1,LVQ2 and PCA Network are all set randomly between [0,1)

The learning rate for LVQ1 and LVQ2 is 0.05 and multiply by 0.98 for each epoch, as i found leaning rate that greater than 0.1 may sometimes cause the vector reach an infinite number

LVQ1

Codebook vector for LVQ1:

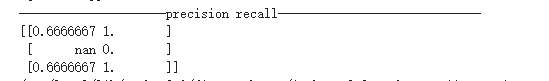
[[4.84663773 -0.0391240567 7.23872]

[3.8785758 0.227406427 3.0725193]

[0.367404133 -0.316297114 7.51224089]

[-0.194959894 0.412148952 3.35053611]]

LVQ1 precision recall on testing set:



The confusion matrix for LVQ1 list as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| act  prd | setosa | versicolor | virginica |
| setosa | 10 | 0 | 0 |
| versicolor | 5 | 0 | 5 |
| virginica | 0 | 0 | 10 |

LVQ2

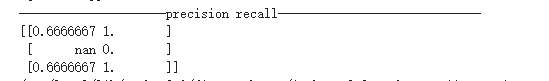
Codebook vector for LVQ2:

[[4.50956345 -1.4317615 1.97766984]

[2.57786727 -2.67181873 10.5428848]

[3.4242 -1.47468007 3.80100536]]

LVQ1 precision recall on testing set:



The confusion matrix for LVQ1 list as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| act  prd | setosa | versicolor | virginica |
| setosa | 10 | 0 | 0 |
| versicolor | 5 | 0 | 5 |
| virginica | 0 | 0 | 10 |

##### PCA net work

The PCA network have the same structure as the LVQ model:no hidden layers, 4-inputs to 3-outputs

The output function for PCA models is the sum of weight\*input



The learning rate for PCA network is 0.01, for the same reason too much learning rate may cause the weight reach infinite

The final weight for the PCA model:

[[0.235385492 0.323478431 0.860726058]

[0.904455125 0.206983894 -0.21307835]

[-0.353054464 0.803417802 -0.0301873628]

[0.0446025059 0.455216736 -0.461479098]]

###### Conclusion

The LVQ network performance on testing set are the same for both with/without the PCA layer

The overall accuracy is 66%, which is good.

During the the training, I found that the LVQ rarely predict the input data as class2(iris\_versicolor) on the initial codebook vector value between[0,1). So the vector for that class rarely changes, cause the poor performance with classify the class2 data in testing dataset.