Abstract

This paper introduces a new method which employs the concept of "Orientation Vectors" to train a feed forward neural network. It is shown that this method is suitable for problems where large dimensions are involved and the clusters are characteristically sparse. For such cases, the new method is not NP hard as the problem size increases. We ‘derive’ the present technique by starting from Kolmogrov’s method and then relax some of the stringent conditions. It is shown that for most classification problems three layers are sufficient and the number of processing elements in the first layer depends on the number of clusters in the feature space. This paper explicitly demonstrates that for large dimension space as the number of clusters increase from N to N+dN the number of processing elements in the first layer only increases by d(logN), and as the number of classes increase, the processing elements increase only proportionately, thus demonstrating that the method is not NP hard with increase in problem size. Many examples have been explicitly solved and it has been demonstrated through them that the method of Orientation Vectors requires much less computational effort than Radial Basis Function methods and other techniques wherein distance computations are required, in fact the present
method increases logarithmically with problem size compared to the Radial Basis Function method and the other methods which depend on distance computations e.g. statistical methods where probabilistic distances are calculated. A practical method of applying the concept of Occum’s razor to choose between two architectures which solve the same classification problem has been described. The ramifications of the above findings on the field of Deep Learning have also been briefly investigated and we have found that it directly leads to the existence of certain types of NN architectures which can be used as a “mapping engine”, which has the property of “invertibility”, thus improving the prospect of their deployment for solving problems involving Deep Learning and hierarchical classification. The latter possibility has a lot of future scope in the areas of machine learning and cloud computing.

References

19. D. George and J.C. Hawkins: Trainable hierarchical memory system and method,
20. Corrinna Cortes and Vladimir Vapnik: Support-Vector Networks, Machine Learning, 20,
21. William B. Johnson and Joram Lindenstrauss: Extensions of Lipschitz mappings on to a
22. Sanjoy Dasgupta and Anupam Gupta: An Elementary Proof of a Theorem of Johnson
23. G.E. Hinton and R.R. Salkhutdinov: Reducing the Dimensionality with Neural Networks,v
24. Dasika Ratna Deepthi and K. Eswaran: A mirroring theorem and its application to a new
25. Dasika Ratna Deepthi and K. Eswaran: Pattern recognition and memory mapping using
2010.
27. R.P. Lippmann: An introduction to computing with neural nets, IEEE,ASSP magazine, pp
4-22 (1987)
28. Ralph P. Boland and Jorge Urrutia: Separating Collection of points in Euclidean Spaces,
29. K.Eswaran:A system and method of classification etc. Patents filed IPO No.(a)
1256/CHE July 2006 and (b) 2669/CHE June 2015
30. K.Eswaran: A non iterative method of separation of points by planes and its application,
31. K.Eswaran and C. Chaitanya: Cloud based unsupervised learning architecture, Recent
researches In AI and and Knowledge Engg. Data Bases, WSEAS Conf. at Cambridge Univ.

Index Terms
Computer Science
Artificial Intelligence

Keywords
Neural Networks, Neural Architecture, Deep Learning, Orientation Vector, Kolmogorov