Implementation of Radial Basis Function Classifiers

Majid Nasiri Manjili
Shahid Rajaee teacher training University, Tehran, Iran
majid.nasiri@srttu.edu

Abstract

In this report radial basis function classifier is implemented on different datasets (synthetic 2d gaussian, iris and satimage). We have tested classification of these datasets by choosing different number of Hidden Layers (clusters) to analyze effect of this parameter on convergence of error rate. This report is based on Machine Learning course.

Keywords: classification, K-means, RBF

1 Introduction

Classification is task of assigning a class to unknown samples, this is a vital part of every machine learning problem. In fact better classification results more accuracy in following parts of algorithm and finally higher performance. This is the reason why the improvement of this classifiers has remained an active research topic in past years.

Historically radial basis functions were introduced for the purpose of extract function interpolation (Powell. 1987) [1]. A radial basis function network is an artificial neural network that uses radial basis functions as activation functions. The output of the network is a linear combination of radial basis functions of the inputs and neuron parameters. Radial basis function networks have many uses, including function approximation, time series prediction , classification, and system control [2]. Radial basis function (RBF) networks typically have three layers: an input layer, a hidden layer with a non-linear RBF activation function and a linear output layer (Figure 1). In otherwise we can separate RBF architecture as unsupervised and supervised sections as figure 2.

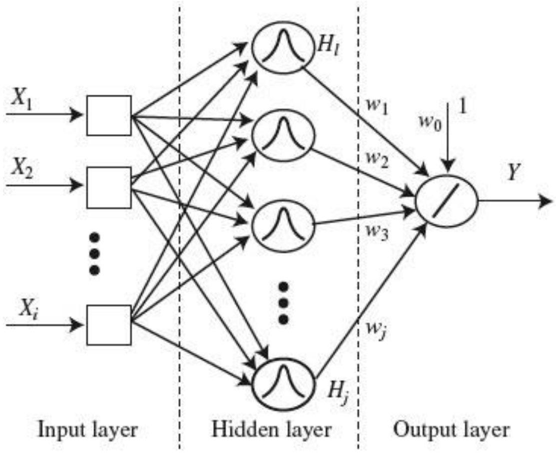

Figure 1: Radial Basis Function Architecture

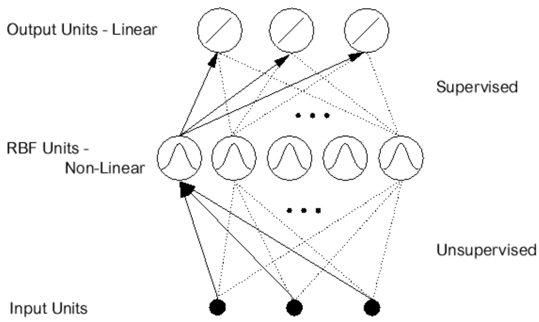

Figure 2: Supervised and unsupervised section of RBF

In this report we have implemented RBF classifier

In the following parts brief information about datasets is noted (2), and after developing technical details about implementation process and reporting results of classification on different datasets (3,4), finally there will be conclusion about results.

2 Datasets

Three different datasets have been used to evaluate classification ability of RBF classifier. The first one is a two class named positive class and negative class by Gaussian distribution (Figure 1).

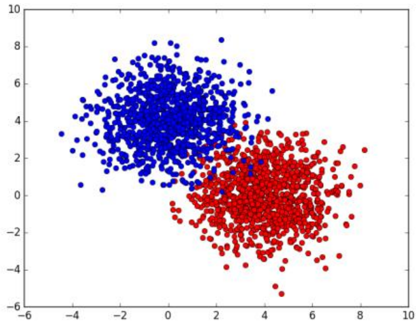

Figure 3: synthetic 2D Gaussian distribution

second one is iris datasets that consists of 3 different types of irises' (Setosa, Versicolour, and Virginica) petal and sepal length, stored in a 150x4 array, The rows being the samples and the columns being: Sepal Length, Sepal Width, Petal Length and Petal Width [1]. The third dataset is satimage, this dataset consists of the multi-spectral values of pixels in 3x3 neighborhoods in a satellite image, and the classification associated with the central pixel in each neighborhood. In this dataset there are 6 classes and 6435 samples (4435 training set and 2000 test set) with 36 attributes for every samples.

3 Implementation

RBF networks are typically trained by a two-step algorithm. In the first step, the center vectors of the RBF functions in the hidden layer are chosen. This step can be performed in several ways; centers can be randomly sampled from some set of examples, or they can be determined using k-means clustering. Note that this step is unsupervised. A third Backpropagation step can be performed to fine-tune all of the RBF net's parameters [3]. The second step simply fits a linear model with coefficients W_i to the hidden layer's outputs with respect to some objective function. A common objective function, at least for regression/function estimation, is the least squares function.

In the model that we have implemented, we have used k-means clustering algorithm to obtain center and spread of non-linear activation functions, so we used center of clusters as center activation functions and also variances of each cluster for spread of activation functions.

4 Results

4.1 Synthetic 2D gaussian

We have tested our RBF model to classify synthetic 2D gaussian by changing number of neurons in Hidden Layers by 2, 5 and 10 the corresponding results are showing in figure 4, 5 and 6. In these figures for each K, we tested multiple times as shown with different colors in figures.

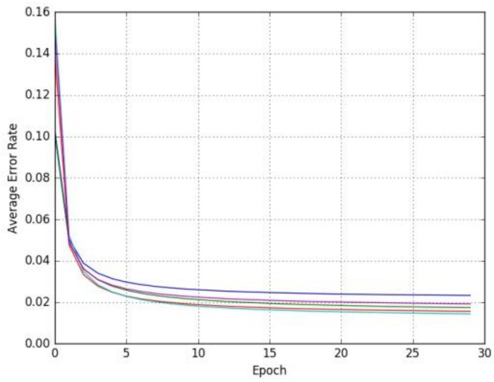

Figure 4: Classification of synthetic 2D gaussian by K=2

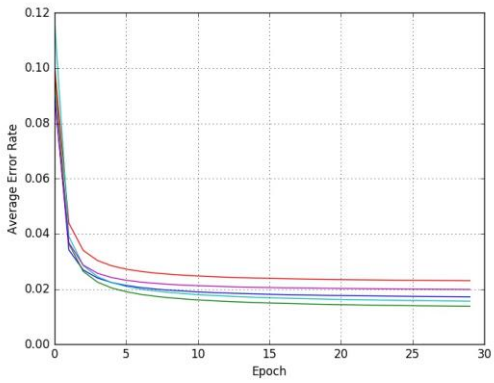

Figure 5: Classification of synthetic 2D gaussian by K=5

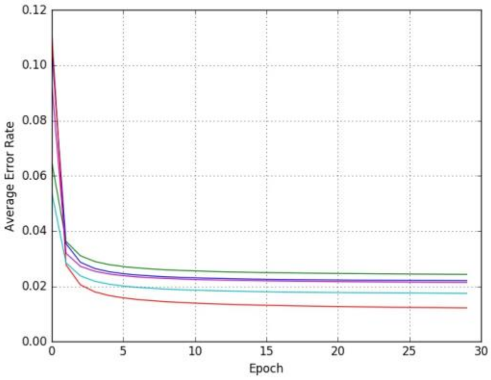

Figure 6: Classification of synthetic 2D gaussian by K=10

For this dataset after classification, we plot classified samples by 2 colors to see the performance of classifier visually, and the result is promising as we can see most of samples are classified correctly. In figure 7 samples are shown in their correct classes as their labels and in figure 8 we can see class of each sample after classification.

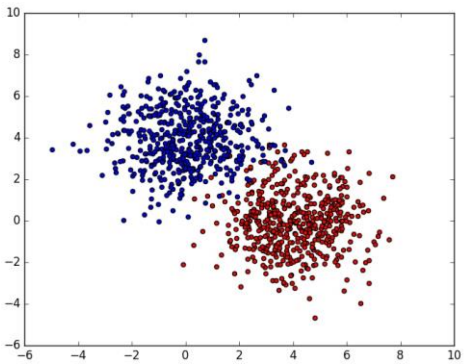
Figure 7: synthetic 2D Gaussian distribution

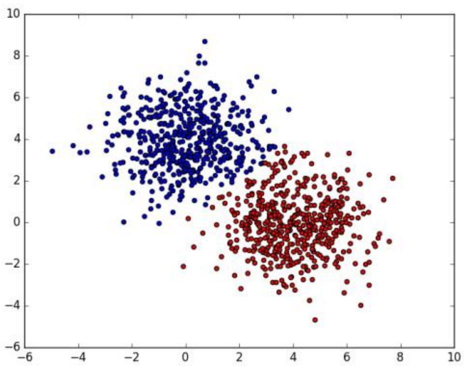
Figure 8: synthetic 2D Gaussian classified by RBF

4.2 Iris Dataset

We have done similar tests on iris datasets with different number of hidden layers 3, 7 and 12 and following figures 7, 8 and 9 are showing corresponding performance of RBF model to classify this dataset.

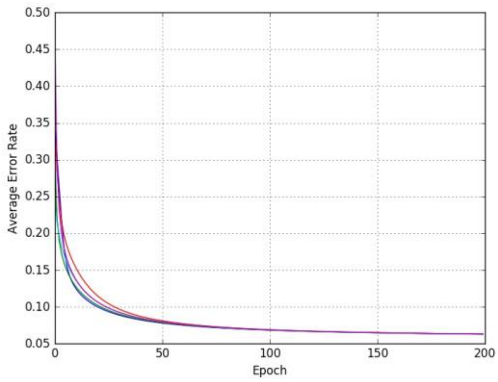

Figure 9: Classification of Iris dataset by K=3.

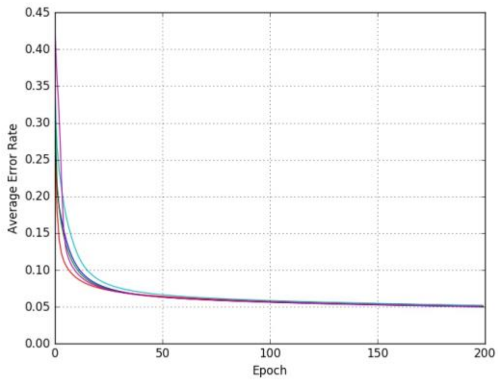

Figure 10: Classification of Iris dataset by K=7.

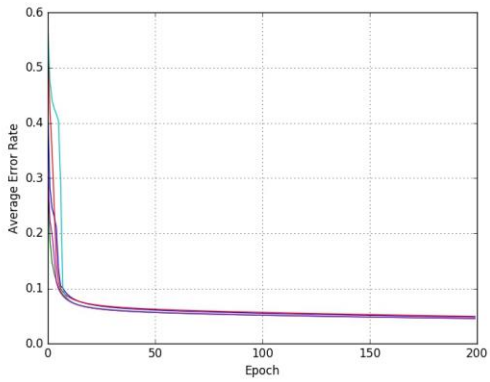

Figure 11: Classification of Iris dataset by K=12.

4.3 Satimage Dataset

We continued our tests to challenge RBF classifier to classify satimage tough dataset. We have classified satimage dataset by three different topology, these topologies using different neuron in there hidden layers, i.e. 6, 12 and 18 neurons for their hidden layers. In figure 10 we draw performance of all topologies with different colors, blue 6 neurons, green 12 neurons and red 18 neurons.

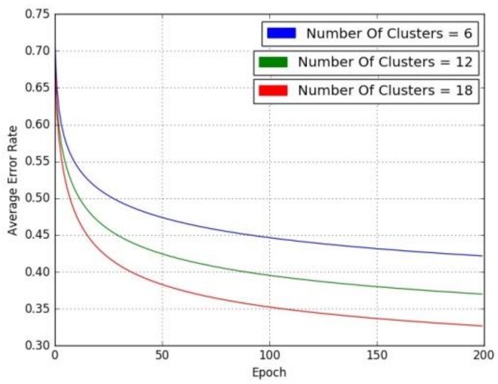

Figure 12: Classification of Satimage dataset by K=6 (blue), K=12 (green) and K=18 (red).

5 Conclusion

As shown in figures it's obvious that RBF classifier perform promising on these three datasets as we can see its performance on synthetic 2d Gaussian distribution, Iris and satimage datasets is correspondingly over **98%**, **95%** and **67%**. Then we can roughly conclude that RBF is a universal classifier. Like other Neural Networks, the performance of RBF network will improves by training model in more iterations (epochs).

Form several tests with different neurons in hidden layer we can conclude that more neurons in hidden layer transform inputs from feature space to more complex space in hidden layer, and this results more faster convergence. For example as we can see in figure 12 the performance of classification improved by adding more neurons in hidden layer. Error rate of **43%**, **37%** and **33%** for 6, 12 and 18 neurons.

References

[1] Bishop, C. "Pattern Recognition and Machine Learning (Information Science and Statistics), 1st edn. 2006. corr. 2nd printing edn." *Springer, New York* (2007).

[2]https://en.wikipedia.org/wiki/Radial_basis_function_network

[3] Schwenker, Friedhelm, Hans A. Kestler, and Günther Palm. "Three learning phases for radial-basis-function networks." *Neural networks* 14.4 (2001): 439-458.