Prctica 2: Comparison of SVM with BP and Linear Regression

Pulak Roy

Universitat Rovira i Virgili,
Av. Països Catalans 26,
E-43007 Tarragona, Catalonia, Spain
e-mail {pulakroy.pulak}@estudiants.urv.cat

1 Introduction

The goal of any supervised learning algorithm is to find a function that best maps a set of inputs to its correct output.Back-propagation is a common method for training a neural network.In machine learning, support vector machines (SVMs) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Back-propagation algorithm was to find a way to train a multi-layered neural network such that it can learn the appropriate internal representations to allow it learn any arbitrary mapping of input to output.Alongside Multiple linear regression attempts to model the relationship between two or more explanatory variables and a response variable by fitting a linear equation to observed data. Every value of the independent variable x is associated with a value of the dependent y.

2 Objective

Objective: Compare the classification results with the following algorithms:

Support Vector Machines (SVM)
Back-Propagation (BP)
Multiple Linear Regression (MLR)

Data:

File::

Training: ring-separable.txt, ring-merged.txt
Test: ring-test.txt

Columns: 2 variables, 1 value to predict

Variables:

x [-1,1]

y [-1,1]

Prediction: class of the pattern, either 0 or 1

ring-separable.txt: class 1) inside a ring of internal radius 0.2 and external radius 0.8; class 0) outside the ring, inside a square of sides between -1 and 1

ring-merged.txt: here the belonging to each class depends on a probability distribution which is a function of the distance to the center of the ring. The Bayesian decision boundary between both classes is once again the circles of radii 0.2 and 0.8 respectively,in such a way that class 1 is more likely inside the ring, and class 0 outside it.

Patterns:: 10000 patterns

Support Vector Machine (SVM)

Classifying data is a common task in machine learning. Suppose some given data points each belong to one of two classes, and the goal is to decide which class a new data point will be in. Given a set of training examples, each marked for belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

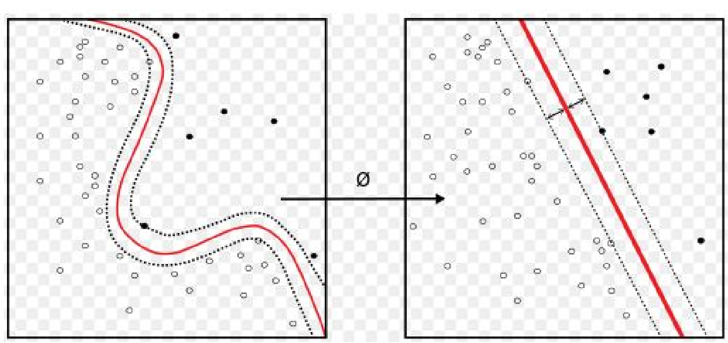

Figure1: Support Vector machine

Implementation approach:
Tool: In this case to classify our data I have used open source machine learning library. LIBSVM is a popular open source machine learning libraries, developed at the National Taiwan University and both written in C++ though with a C API. LIBSVM implements the SMO algorithm for kernelized support vector machines (SVMs), supporting classification and regression.

Implementation decisions:
To classify you need to convert both training and test data according to the format supported by LIBSVM. In this case I have prepared both train and test file by writing a java program.Then to train ring-separable.txt I have used commands: "svm-train.exe -c 1000 -t 2 -g 3 ring-separable.train". After executing this command a model file named "ring-separable.train.model" will be generated. Then to predict the command will be : "svm-predict.exe ring-tests.test ring-separable.train.model ring-separable.out". After this it showed prediction accuracy 99.64%
and a ring-separable.out file will generated. Similar procedure has been followed for ring-merge.txt data file. In this case it showed 98.51% prediction accuracy.

Result(predictions):
ring-separable.txt:99.64%
ring-merge.txt : 98.51%

Evaluation of the predictions(Error and Plot):
When I took output file's content into excel and calculated error.
Error:
ring-separable.txt: 0.77%
ring-merge.txt: 8.59%

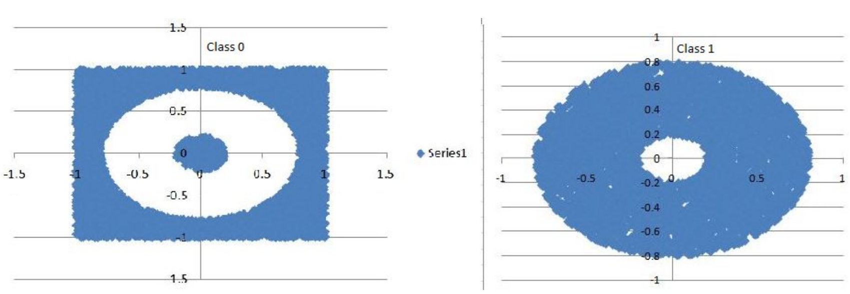

Figure2: SVM prediction: ring-separable

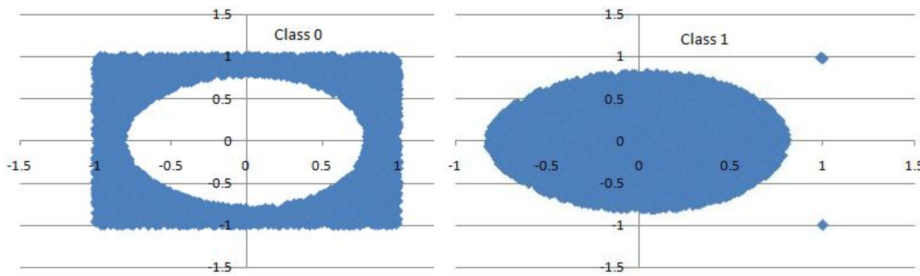

Figure3: SVM prediction: ring-merge

Back-Propagation (BP)

Backpropagation is a supervised learning algorithm and is mainly used by Multi-Layer-Perceptrons to change the weights connected to the net's hidden neuron layer(s). The back-propagation algorithm uses a computed output error to change the weight values in backward direction. To get this net error, a forward-propagation phase must have been done before.While propagating in forward direction, the neurons are being activated using the sigmoid activation function.The formula of sigmoid activation is: f(x) = $1/(1+e^{-x})$. The algorithm works as follows:

```
Randomly initialize weights and thresholds
for epoch 1 to total number of epoch
    for pattern 1 to total number of pattern
        choose random pattern.
        feed-forward
        back-propagation of error
        calculation of weights and thresholds updates
        update weights and thresholds
    end
end
```

Implementation decisions:

In this current implementation we can choose number of hidden layers and number of neurons per layer. For our data it is better to use two hidden layers. On the other hand to choose I have chosen thirteen nodes in first hidden layer and 5 at second hidden layer.Because this network architecture gives best prediction with error rate 02.81% for ring-separable and 11.72% for ring-merge. Along side number of epoch and learning rate plays an important role in the prediction. In this case I preferred epoch

= 10000000 and Learning rate = 0.1

Language: Java
Tool: Eclipse JavaEE mars
Result(predictions)

ring-separable.txt: 97.19%
ring-merge.txt : 89.28%

Evaluation of the predictions(Error and Plot):
Error

ring-separable.txt: 2.81%
ring-merge.txt: 11.72%

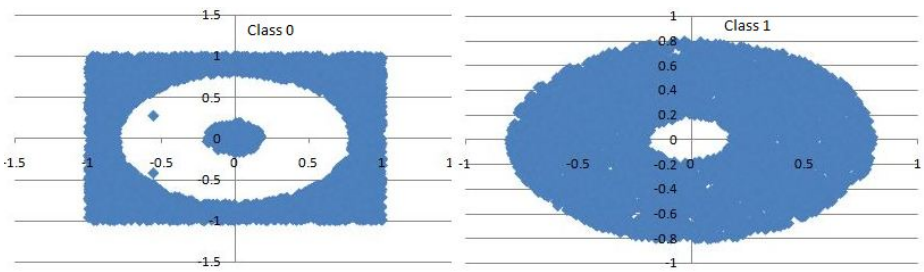

Figure4: BP prediction: ring-separable

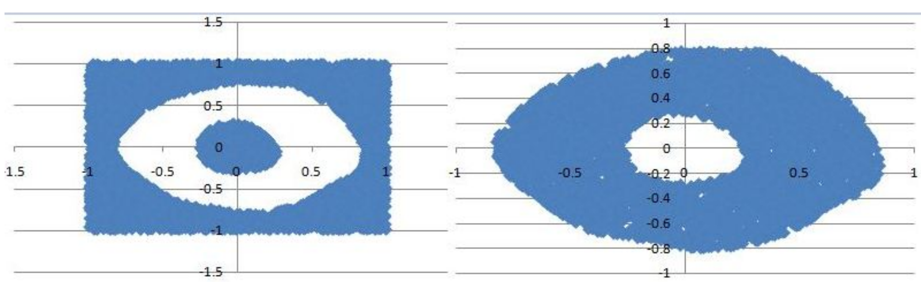

Figure5: BP prediction: ring-merge

Multiple Linear Regression
Multiple linear regression attempts to model the relationship between two or more explanatory variables and a response variable by fitting a linear equation to observed data. Every value of the independent variable x is

associated with a value of the dependent variable y. Formally, the model for multiple linear regression, given n observations, is $y = b_0 + b_1x_1 + b_2x_2 + ... + b_nx_n$ for i = 1,2......n

Implementation decisions:
We first evaluate coefficient to form a model. In this case we get b0, b1 and b2 for each data set and then form the model according to the above equation. Then we use this function to predict for ring-test data set.
Language: Using software
Tool: Microsoft office Excel 2007
Result(predictions):

ring-separable.txt:0%
ring-merge.txt : 0%

Evaluation of the predictions(Error and Plot):
Error:

ring-separable.txt: 100.94%
ring-merge.txt: 100%

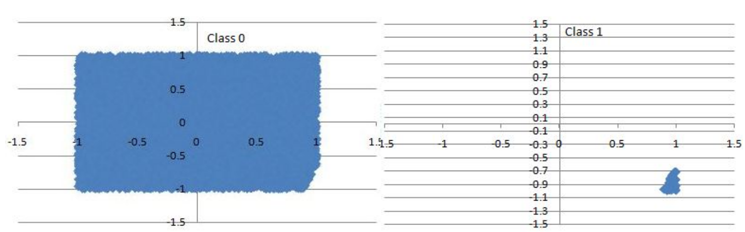

Figure6: Multi linear regression prediction: ring-separable

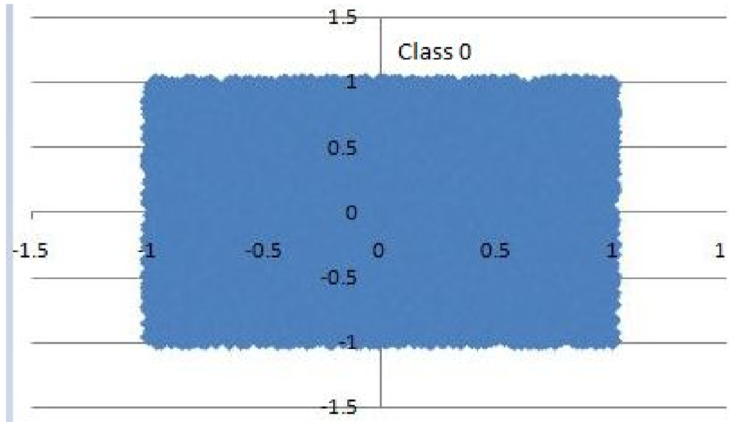

Figure7: Multi linear regression prediction:ring merge

Conclusion

As we can see MLR could not classify between two classes, it keeps all data within class 0. As we can see SVM gives higher precisions and lower error rate in comparison to back-propagation.But in the case of complex data prediction, MLR does not work well.It's predictions is far way from SVM and back-propagation mechanism.Alongside SVM is much faster than back-propagation.

References

1. "Backpropagation", [Online]. Available: , https://en.wikipedia.org/wiki/Backpropagation [Accesed: Jun. 5, 2016]

2. "Support vector machine", [Online]. Available: , https://en.wikipedia.org/wiki/Support_vector_machine [Accesed: Jun. 5, 2016]

3. "Multiple Linear Regression ", [Online]. Available: , http://www.stat.yale.edu/Courses/1997-98/101/linmult.htm [Accesed: Jun. 5, 2016]