Support Vector Machine Approach to Automated Cardiac SPECT diagnosis

8th Semester Project Presentation

Varun B Patil

2012

Dept. of Computer Science & Engineering

Sri Jayachamarajendra College of Engineering, Mysore

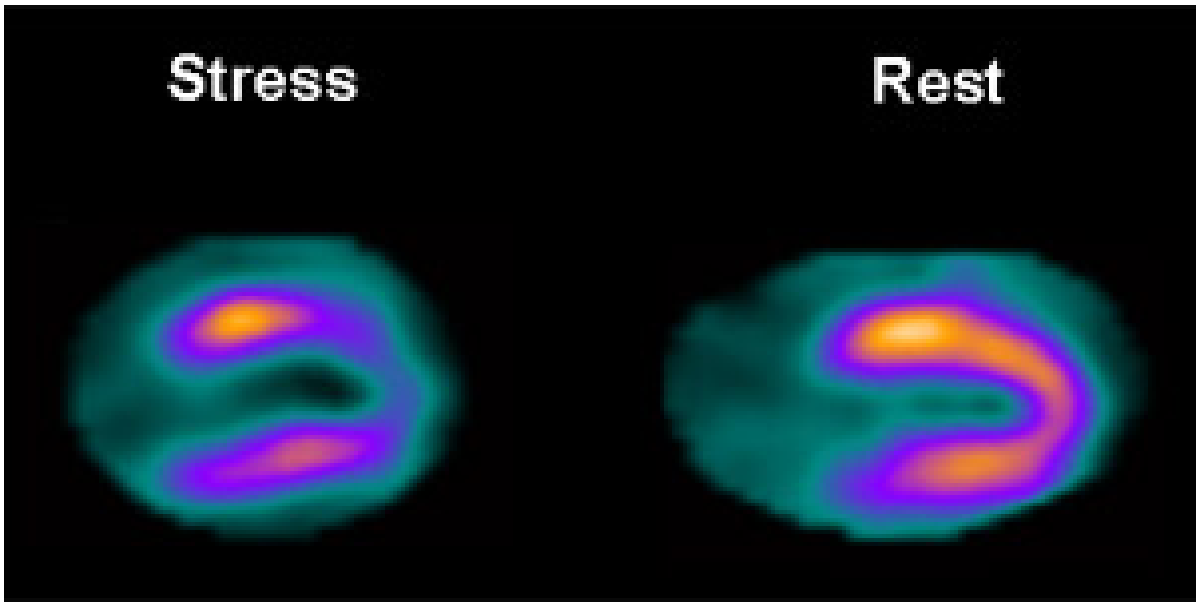
What is SPECT ?

SPECT stands for Single Photon Emission Computerized Tomography. It is the state-of-the-art in detecting various types of cardiac diseases. SPECT provides us with computer images which can be analyzed to determine the type of cardiac disease. The aim of this project is to make sense out of the data produced after processing these SPECT images.

What is SVM ?

SVM stands for Support Vector Machines. It is a supervised machine learning algorithm which we use here to classify instances of SPECT data as indicating possible cardiac disease or not. SVM learns from pre-classified data we provide and then automatically classifies un-classified data based on rules learned previously.

Some SPECT images . . .



Inputs to program

- SPECT is actually images. It has to be preprocessed to generate numerical data that the program can work on.
- SPECT images are few and far between. For all computational machine learning purposes, UCI maintains a comprehensive database of numerical SPECT data collected from some of the best medical institutions around the world that we will be using.
- The data set we will be using was obtained from http://archive.ics.uci.edu/ml/datasets/SPECT+Heart

Structure of input data

- The UCI data set is a comma-seperated-value file (CSV) which we read into the octave program as a two dimensional matrix using a built in function called csvread().
- There are binary values for about 23 features indicating the presence or absence of a particular feature in that particular SPECT image instance. These features include left ventricular ejection fraction, stroke volume, myocardial perfusion, etc.
- The data is provided in two parts: Training set used to train the machine learning classifier and Test set used to test the classifier's accuracy.

Dividing the Test Set

- The Test Set itself is further divided into a Cross Validation set and a Test set. The CV consists of random samples from the Test Set.
- The cross validation(CV) set is primarily used to tune the classifier program for best performance; it is not used to gauge the accuracy of the classifier. For example, the CV set is used to select the best possible values for C and sigma used in the SVM algorithm.
- The remaining part of the Test Set is the one that is actually used to measure the accuracy of the classifier after it has been trained on the Training Set.

Dividing the Test Set

- Usually the data is divided as training set(60%), CV set(20%) and test set(20%). The distribution between the sets is done randomly. Then, the hypothesis that minimizes the CV set error(fraction of CV set instances that are wrongly classified) is chosen and the error on the test set is reported as the generalized error.

SVM algorithm

- The SVM machine learning algorithm aims to find the best relationship between the features and its class so as to minimize the error in classification.
- The algorithm we employ is called SMO(Sequential Minimal Optimization) and it finds the best possible class to which a particular instance of SPECT data belongs to. The term “best class" here means the class which minimizes the classification error or likewise improves the classification accuracy.
- The two classes possible for any SPECT instance are: indicates possible cardiac disease, does not indicate cardiac disease(future enhancements to the project will include more fine grained classification or multiclass classification).

Choices in implementation

- The SVM algorithm we have implemented can either use a Linear Kernel or a Gaussian Kernel.
- A Linear Kernel can be imagined to be a straight-line seperation between the classes(or a planar seperation in the case of multidimensional data).
- The Gaussian Kernel on the other hand can be visualized as a non-linear seperation between classed which also means a non-planar or curved seperation between classes of multidimensional data.

$$K_{linear}(x^i, x^j) = x^i * x^j$$

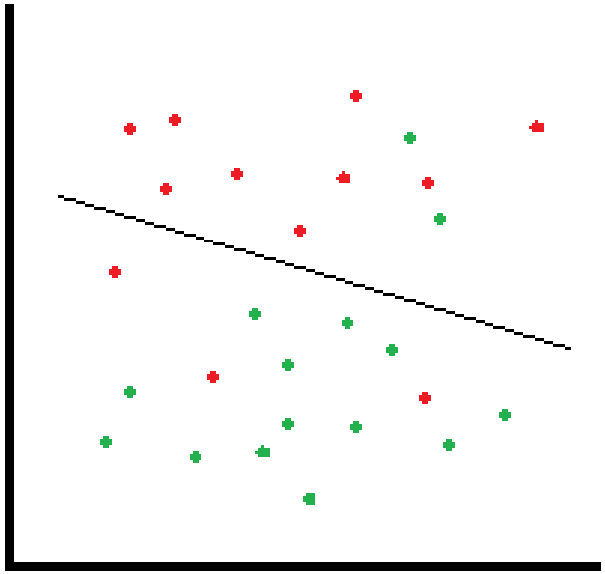
$$K_{gaussian}(x^i, x^j) = \exp(-\frac{||x^i - x^j||^2}{2\sigma^2})$$

Linear Vs. Gaussian

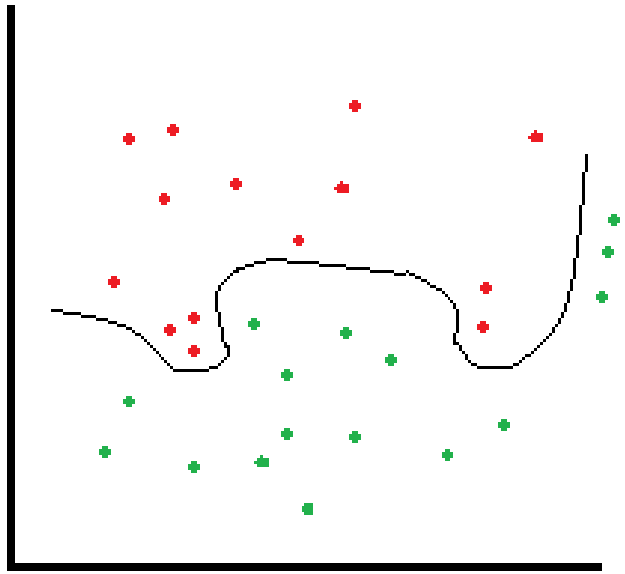
Which one do you use ?

- With choice comes the problem of deciding which one to use for the particular problem at hand.
- A Linear Kernel is usually employed when the number of features is more than the number of samples and a Gaussian Kernel is employed when the number of samples is more than the number of features.
- And also, the kernel used depends on whether the data itself is linearly-seperable or not.

Linear Vs. Gaussian



Linear Kernel



Gaussian Kernel

Optimization

- The optimization problem in SVM is that of minimizing:

$$J = (C * A) + B \text{ where,}$$

$$A = \Sigma(y^i * cost_1(z) + (1 - y^i) * cost_0(z))$$

$$B = \frac{1}{2}\Sigma(\Theta^2)$$

$$cost_1(z) = -\log(z)$$

$$cost_0(z) = -\log(1 - z)$$

- Once we know Θ, we can predict the class as follows:

$$if(\Theta^T x) > 0, \text{predict class} = 1$$

$$if(\Theta^T x) < 0, \text{predict class} = 0$$

Role of C and σ

- We noted earlier that C and σ are two important parameters that need to be decided by examining the Cross Validation set.
- Informally, the C parameter is a positive value that controls the penalty for misclassified training examples. A large C parameter tells the SVM to try to classify all the examples correctly(large margin). However, large C means that it is more susceptible to outliers and does not give a natural fit for the data.

Role of C and σ

- σ is a parameter of the Gaussian Kernel which determines how fast the "similarity measure" approaches 0 for data points that are further apart. If σ is large, the similarity changes slowly and can cause underfitting. If σ is small, the similarity changes fast and can cause overfitting.
- In practice, C and σ are selected from a set of highly possible values for both C and σ where the values vary approximately by a factor of 10, by testing all possible combinations of C and σ for the one that gives the least error in classification of the CV set. In our implementation, the values of C and σ take on the following set of values

$$C = [0.01, 0.03, 0.1, 0.3, 1, 3, 10, 30]$$

$$\sigma = [0.01, 0.03, 0.1, 0.3, 1, 3, 10, 30]$$

Expectations from the algorithm

- When all is said and done, you expect the classifier to tell you the class to which a particular SPECT instance belongs to. One way to check whether what the classifier predicts is “upto the mark" is to compare the result with already known values.
- There are several ways to gauge this “upto mark" notion in terms of numerical values.
- Prediction Accuracy is the fraction of the instances in the Test set that are correctly classified.

$$\text{Accuracy} = \frac{\text{Test set instances correctly classified}}{\text{total Test set instances}}$$

Expectations from the algorithm

- Sensitivity is the fraction of positive instances in the Test Set that are correctly classified as belonging to the positive class.

$$\text{Sensitivity} = \frac{\text{positive instances correctly classified}}{\text{total postive instances in Test set}}$$

- Specificity is the fraction of negative instances in the Test Set that are correctly classified as belonging to the negative class.

$$\text{Specificity} = \frac{\text{negative instances correctly classified}}{\text{total negative instances in Test set}}$$

- When you have high values for all of the above numerical measures, you know that your classifier is “upto the mark"

Why SVM ?

- One algorithm of comparable complexity is Neural Networks(NN). Infact we do have a working prototype to solve the same problem using NN.
- The first thing we observed with the neural networks prototype is that its accuracy heavily depends on its structure which is almost decided on “intuition" alone. However, machine learning experts will testify to the fact that trusting your intuition is not a good idea. The upshot is that a neural network structure that seems perfect for one problem might perform terribly on another problem. It is just not possible to stick to one structure.

Why SVM ?

- On the contrary, SVM does not rely on “intuition". It is able to decide on the best parameters using information from the problem itself. In our case, the best possible values for C and sigma were determined from the problem itself(CV set).
- Ofcourse several advanced algorithms can perform better and faster, but our goal has been to demonstrate the use of machine learning and data mining to improve and ease disease diagnosis and not designing the ultimate cardiologist substitute.

Possible future improvements

- Classification accuracy can further be improved by including the person's age, sex, cardiophysiology and previous medical history as features that aid the classification. Such data though hard to obtain due to privacy issues can be a major accuracy booster.
- Another feature that has been debated for a long time in medical research is the demography in which the person has been brought up. It is obvious why this is so important. Take for example the fact that a malaria outbreak in America is more likely to reach pandemic proportions when compared to Africa or the fact that Africans are more suited to marathon running compared to Indians. It is just in their genes !!!

Possible future improvements

- Another simple improvement would be to provide a more fine-grained classification involving several classes of cardiac diseases instead of just a yes/no solution. Classes could include Cardiomyopathy or Cardiac dysrhythmias or Endocarditis or Inflammatory cardiomegaly, and many more.