GROUP B

Assignment no.11

1. Problem Statement: Case study on LIBSVM -A Library for Support Vector Machines. (using LDA for dimensionality reduction).

2. Objective:

1. To apply algorithmic strategies while solving problems
2. To develop time and space efficient algorithms
3. To develop problem solving abilities for gamifications.

3. Theory:

Support Vector Machine:

The support vector machines (SVM) were created by Vladimir Vapnik in the 1990s and can be used for solving classification and regression tasks. They are based on the principle of structural risk minimization, which enhances model robustness by ensuring that the model complexity is not too high as measured by the so called VC-dimension (Vapnik, 1998). For comparison, neural networks and other traditional black box techniques normally minimize the empirical risk which basically is the average quadratic error over a number of samples, the training set. Within the SVM framework, radial basis networks, single hidden layer sigmoidal neural networks as well as other kinds of models can be set up, depending on the chosen kernel. A nice property of the SVM is that it yields a unique optimal solution of the resulting optimization problem.

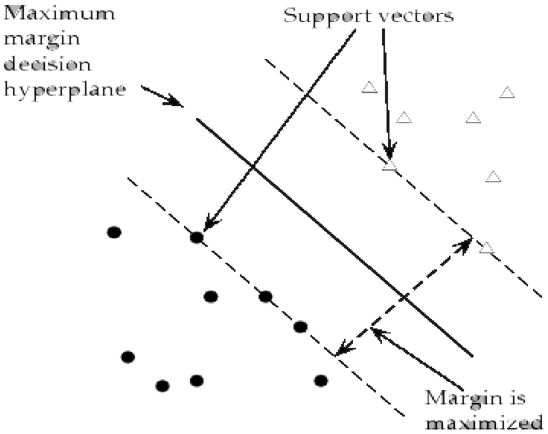


Fig.1. The support vectors are the 5 points right up against the margin of the classifier.

For two-class, separable training data sets, there are lots of possible linear separators. Intuitively, a decision boundary drawn in the middle of the void between data items of the two classes seems better than one which approaches very close to examples of one or both classes. While some learning methods such as the perceptron algorithm, find just any linear separator, others, like Naive Bayes, search for the best linear separator according to some criterion. The SVM in particular defines the criterion to be looking for a decision surface that is maximally far away from any data point. This distance from the decision surface to the closest data point determines the margin of the classifier. This method of construction necessarily means that the decision function for an SVM is fully specified by a (usually small) subset of the data which defines the position of the separator. These points are referred to as the *support vectors* (in a vector space, a point can be thought of as a vector between the origin and that point). Figure.1 shows the margin and support vectors for a sample problem. Other data points play no part in determining the decision surface that is chosen.

LIBSVM:

LIBSVM is an integrated software for support vector classification, (C-SVC, nu-SVC), regression (epsilon-SVR, nu-SVR) and distribution estimation (one-class SVM). It supports multi-class classification.

The primary goal is to help users from other fields to easily use SVM as a tool. **LIBSVM** provides a simple interface where users can easily link it with their own programs. Main features of **LIBSVM** include

- Different SVM formulations
- Efficient multi-class classification
- Cross validation for model selection
- Probability estimates
- Various kernels (including precomputed kernel matrix)
- Weighted SVM for unbalanced data
- Both C++ and Java sources
- GUI demonstrating SVM classification and regression
- Python, R, MATLAB, Perl, Ruby, Weka, Common LISP, CLISP, Haskell, OCaml, LabVIEW, and PHP interfaces. C# .NET code and CUDA extension is available. It's also included in some data mining environments: RapidMiner, PCP, and LIONsolver.
- Automatic model selection which can generate contour of cross validation accuracy.

The programs:

How to use the most important executables here. The filenames are a little bit different under Unix and Windows, apply common sense to see:

svmtrain : Use your data for training. Running SVM is often referred to as 'driving trains' by its non-native English speaking authors because of this program. svmtrain accepts some specifically format which will be explained below and then generate a 'Model' file. You may think of a 'Model' as a storage format for the internal data of SVM. This should appear very reasonable after some thought, since training with data is a time-consuming process, so we 'train' first and store the result enabling the 'predict' operation to go much faster.

svmpredict

Output the predicted class of the new input data according to a pre-trained model.

svmscale

Rescale data. The original data maybe too huge or small in range, thus we can rescale them to the proper range so that training and predicting will be faster.

File Format

libsvm "heart_scale": This is the input file format of SVM. You may also refer to the file "heart_scale" which is bundled in official libsvm source archive.

[label] [index1]:[value1] [index2]:[value2] ...
[label] [index1]:[value1] [index2]:[value2] ...
.
.

One record per line, as:

+1 1:0.708 2:1 3:1 4:-0.320 5:-0.105 6:-1

label

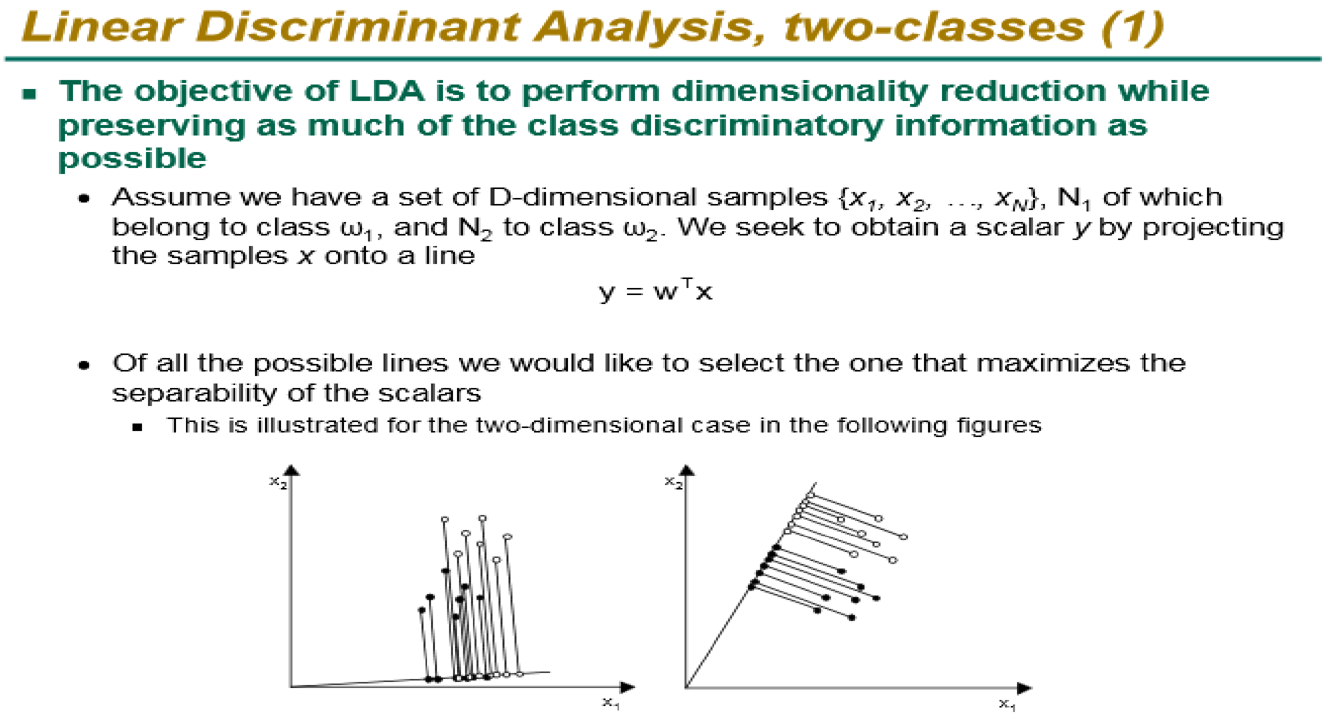
Sometimes referred to as 'class', the class (or set) of your classification. Usually we put integers here.

index

Ordered indexes. usually continuous integers.

value

The data for training. Usually lots of real (floating point) numbers.



Linear Discriminant Analysis, two-classes (1)

- **The objective of LDA is to perform dimensionality reduction while preserving as much of the class discriminatory information as possible**
  - Assume we have a set of D-dimensional samples $\{x_1, x_2, \ldots, x_N\}$, N_1 of which belong to class ω_1, and N_2 to class ω_2. We seek to obtain a scalar *y* by projecting the samples *x* onto a line

$$y = w^T x$$

  - Of all the possible lines we would like to select the one that maximizes the separability of the scalars
    - This is illustrated for the two-dimensional case in the following figures

4. Conclusion:
Hence we have done case study of the LIBVSM tool for Support Vector Machines.

5. Outcomes achieved (mark the outcomes achieved)

COURSE OUTCOME	ACHIEVED(√)
Problem solving abilities for smart devices.	
Problem solving abilities for gamifications.	
Problem solving abilities of pervasiveness, embedded security and NLP.	
To solve problems for multicore or distributed, concurrent/Parallel environments	√