THE UNIVERSITY OF TENNESSEE KNOXVILLE

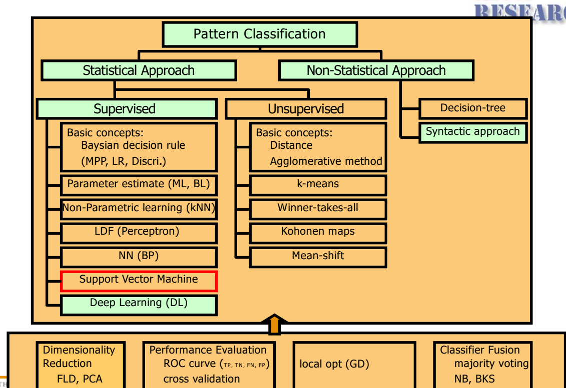
AICIP RESEARCH

ECE471-571 – Pattern Recognition

Lecture 17: Support Vector Machine

Hairong Qi, Gonzalez Family Professor
Electrical Engineering and Computer Science
University of Tennessee, Knoxville
http://www.eecs.utk.edu/faculty/qi
Email: hqi@utk.edu

AICIP RESEARCH



AICIP RESEARCH

- Reference: Christopher J.C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, 2, 121-167, 1998

THE UNIVERSITY OF TENNESSEE KNOXVILLE

A bit about Vapnik

AICIP RESEARCH

- Started SVM study in late 70s
- Fully developed in late 90s
- While at AT&T lab

THE UNIVERSITY OF TENNESSEE KNOXVILLE

http://en.wikipedia.org/wiki/Vladimir_Vapnik

4

Generalization and Capacity

AICIP RESEARCH

- For a given learning task, with a given finite amount of training data, the best generalization performance will be achieved if the right balance is struck between the accuracy attained on that particular training set, and the "capacity" of the machine
- Capacity – the ability of the machine to learn any training set without error
  - Too much capacity - overfitting

THE UNIVERSITY OF TENNESSEE KNOXVILLE

5

Bounds on the Balance

AICIP RESEARCH

- Under what circumstances, and how quickly, the mean of some empirical quantity converges uniformly, as the number of data point increases, to the true mean
- True mean error (or actual risk)

$$R(\alpha) = \int \tfrac{1}{2}\left|y - f(\mathbf{x},\alpha)\right| p(\mathbf{x},y)\, d\mathbf{x}dy$$

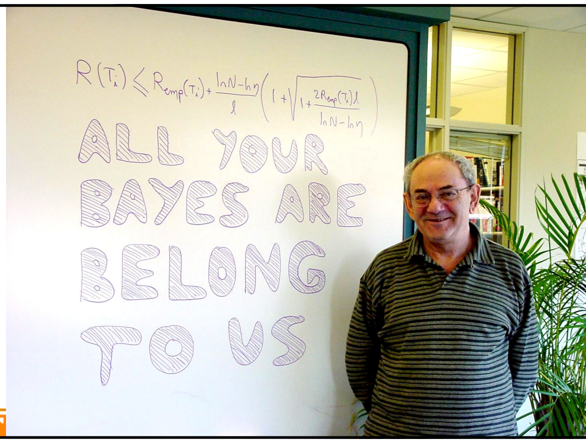
- One of the bounds

$$R(\alpha) \le R_{emp}(\alpha) + \sqrt{\left(\frac{h(\log(2l/h)+1)-\log(\eta/4)}{l}\right)} \qquad R_{emp}(\alpha) = \frac{1}{2l}\sum_{i=1}^{l}\left|y_i - f(\mathbf{x}_i,\alpha)\right|$$

> $f(\mathbf{x},\alpha)$: a machine that defines a set of mappings, $\mathbf{x} \rightarrow f(\mathbf{x},\alpha)$
> α: parameter or model learned
> h: VC dimension that measures the capacity. non-negative integer
> R_{emp}: empirical risk
> η: 1-η is confidence about the loss, η is between [0, 1]
> l: number of observations, y_i: label, {+1, -1}, $\mathbf{x}_i$ is n-D vector

Principled method: choose a learning machine that minimizes the RHS with a sufficiently small η

THE UNIVERSITY OF TENNESSEE KNOXVILLE



VC Dimension

AICIP RESEARCH

- For a given set of l points, there can be 2^l ways to label them. For each labeling, if a member of the set $\{f(\alpha)\}$ can be found that correctly classifies them, we say that set of points is shattered by that set of functions.
- VC dimension of that set of functions $\{f(\alpha)\}$ is defined as the maximum number of training points that can be shattered by $\{f(\alpha)\}$
- We should minimize h in order to minimize the bound

Example ($f(\alpha)$ is perceptron)

AICIP RESEARCH

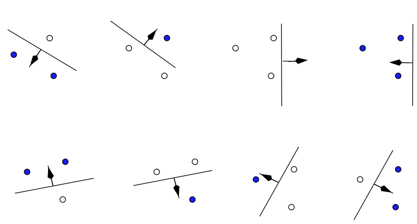
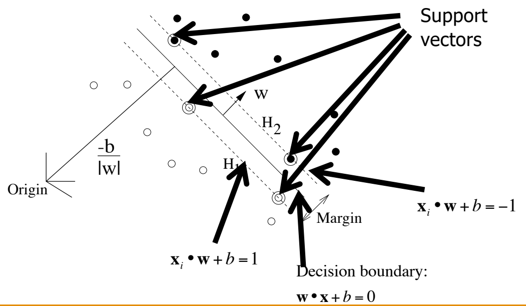


Figure 1. Three points in $\mathbf{R}^2$, shattered by oriented lines.

AICIP RESEARCH

Linear SVM – The Separable Case



THE UNIVERSITY OF TENNESSEE

10

AICIP RESEARCH

$$\begin{cases} \mathbf{x}_i \bullet \mathbf{w} + b \geq 1 \quad \text{for} \quad y_i = +1 \\ \mathbf{x}_i \bullet \mathbf{w} + b \leq -1 \quad \text{for} \quad y_i = -1 \end{cases}$$

Minimizing $\|\mathbf{w}\|^2$

s.j. $y_i(\mathbf{x}_i \bullet \mathbf{w} + b) - 1 \geq 0$

Minimize $L_P = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{l} \alpha_i y_i (\mathbf{x}_i \bullet \mathbf{w} + b) + \sum_{i=1}^{l} \alpha_i$

$$\frac{\partial L_P}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i, \qquad \frac{\partial L_P}{\partial b} = 0 \Rightarrow \sum_i \alpha_i y_i = 0$$

Maximize $L_D = -\frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \mathbf{x}_j + \sum_i \alpha_i$

THE UNIVERSITY OF TENNESSEE

11

AICIP RESEARCH

Non-separable Cases

- SVM with soft margin
- Kernel trick

THE UNIVERSITY OF TENNESSEE

12

AICIP RESEARCH

Non-separable Case – Soft Margin

$$\begin{cases} \mathbf{x}_i \bullet \mathbf{w} + b \geq 1 - \xi_i & \text{for } y_i = +1 \\ \mathbf{x}_i \bullet \mathbf{w} + b \leq -1 + \xi_i & \text{for } y_i = -1 \end{cases} \quad \text{for } \xi_i \geq 0$$

$$\text{Minimizing } \|\mathbf{w}\|^2$$
$$\text{s.j. } y_i(\mathbf{x}_i \bullet \mathbf{w} + b) - 1 + \xi_i \geq 0$$

$$\text{Minimize } L_P = \tfrac{1}{2}\|\mathbf{w}\|^2 - C\left(\sum_i \xi_i\right)^k$$

$$\text{Maximize } L_D = -\tfrac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \mathbf{x}_i \mathbf{x}_j + \sum_i \alpha_i$$
$$\text{s.j. } 0 \leq \alpha_i \leq C, \quad \sum_i \alpha_i y_i = 0$$

THE UNIVERSITY OF TENNESSEE

13

AICIP RESEARCH

Non-separable Cases – Kernel Trick

- If there were a “kernel function”, K, s.t.

$$K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) = e^{-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2}$$

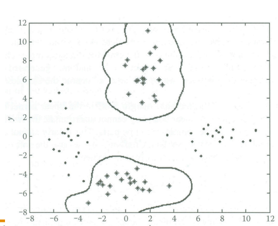
Gaussian Radial Basis Function (RBF)

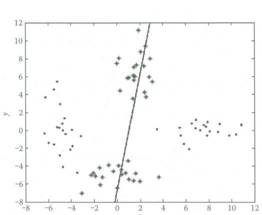
THE UNIVERSITY OF TENNESSEE

14

AICIP RESEARCH

Comparison - XOR





THE UNIVERSITY OF TENNESSEE

15

Limitation

- Need to choose parameters

Packages

- libSVM
  - Use one-against-one (1a)
- SVM^{light}

Package Installation

- Download: http://www.csie.ntu.edu.tw/~cjlin/libsvm/
- Installation (Three choices)
  - On Unix systems, type `make' to build the `svm-train' and `svm-predict' programs.
  - On other systems, consult `Makefile' to build them
  - Use the pre-built binaries (Windows binaries are in the directory 'windows').
  - More details pls refer to the README file

Steps

AICIP RESEARCH

- Step 1: Transform the data to the format of an SVM package
- Step 2: Conduct simple scaling on the data
- Step 3: Consider the RBF kernel $K(x,y) = e^{-\gamma\|x-y\|^2}$
- Step 4: Select the best parameter C and γ to train the whole training set
- Step 5: Test

THE UNIVERSITY OF TENNESSEE KNOXVILLE

Example

AICIP RESEARCH

- Dataset: pima.tr and pima.te
- Step 1: Transform the data to the format of an SVM package
  - $Ptr \in R^{m \times f}$ (training data: every row is a feature vector)
  - $Pte \in R^{n \times f}$ (testing data: every row is a feature vector)
  - ltr (label vector for training data pima.tr)
  - lte (label vector for testing data pima.te)

THE UNIVERSITY OF TENNESSEE KNOXVILLE

Example

AICIP RESEARCH

- Step 2: Data scaling
  - Avoid attributes in greater numeric ranges dominating those in smaller numeric ranges
  - Avoid numerical difficulties during the calculation
- How?
  - Calculate the min and max for every feature from the training dataset
  - For every feature (train or test) f, the scaled feature fs can be calculated by fs = (f-min)/(max-min)

Details pls refer to:
http://www.csie.ntu.edu.tw/~cjlin/libsvm/faq.html#f407
http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf

THE UNIVERSITY OF TENNESSEE KNOXVILLE

Example

AICIP RESEARCH

- Step 3: Train SVM on given parameters

```
model = svmtrain(ltr, sparse(pima_trs), '-c 16 , -g 0.1');
```

- model: Trained model
- ltr: Label vector for training dataset
- -c 16: Penalty parameter
- -g 0.1: Parameter for the kernel function RBF

1, Pima_trs is the matrix for the scaled features of training dataset
2, Sparse(pima_trs) is an operation to generate a sparse matrix in matlab, required by the libsvm package

THE UNIVERSITY OF TENNESSEE

Example

AICIP RESEARCH

- Step 4: Test on the trained model

```
[predict_labels, accuracy, dec_values] = svmpredict(lte, sparse(pima_tes), model);
```

- predict_labels: The predicted labels for testing dataset
- accuracy: Recognition accuracy for the testing dataset
- dec_values: Decision values used to classification
- lte: Label vector for testing dataset
- model: Trained model

1, Pima_tes is the matrix for the scaled features of testing dataset
2, Sparse(pima_tes) is an operation to generate a sparse matrix in matlab, required by the libsvm package

THE UNIVERSITY OF TENNESSEE

Example

AICIP RESEARCH

- Result:
  - Scaled:
    - Accuracy = 80.1205% (266/332)
  - Non-scaled:
    - Accuracy = 66.8675% (222/332)

THE UNIVERSITY OF TENNESSEE

Matlab Code

AICIP RESEARCH

```
% Demonstration of the usage of libSVM on pima data set
% Jiajia Luo

clear all;
clc;

% add the path
addpath('E:\My Code\SourceCodeInternet\libsvm-3.1\windows');

% Step 1: Collect the training and testing dataset
fid = fopen('pima.tr.txt');
tr = textscan(fid,'%f %f %f %f %f %f %f %s','HeaderLines',1);
fclose(fid);

fid = fopen('pima.te.txt');
te = textscan(fid,'%f %f %f %f %f %f %f %s','HeaderLines',1);
fclose(fid);

% Step 2: Generate the feature vectors and labels for tr and te
pima_tr = [tr{1} tr{2} tr{3} tr{4} tr{5} tr{6} tr{7}];
pima_te = [te{1} te{2} te{3} te{4} te{5} te{6} te{7}];
Ntr = size(pima_tr,1);
Nte = size(pima_te,1);
ltr = [tr{8}];
lte = [te{8}];
label_tr = zeros(Ntr,1);
label_te = zeros(Nte,1);
for i = 1:Ntr
    if strcmp(ltr{i},'Yes')
        label_tr(i,1) = 1;
    elseif strcmp(ltr{i},'No')
        label_tr(i,1) = 2;
    end
end
```

THE UNIVERSITY OF TENNESSEE KNOXVILLE

AICIP RESEARCH

```
for i = 1:Nte
    if strcmp(lte{i},'Yes')
        label_te(i,1) = 1;
    elseif strcmp(lte{i},'No')
        label_te(i,1) = 2;
    end
end


% Step 3: scale the data
pima_trs = (pima_tr - repmat(min(pima_tr,[],1),size(pima_tr,1),1))*...
    spdiags(1./(max(pima_tr,[],1)-
min(pima_tr,[],1))',0,size(pima_tr,2),size(pima_tr,2));
pima_tes = (pima_te - repmat(min(pima_tr,[],1),size(pima_te,1),1))*...
    spdiags(1./(max(pima_tr,[],1)-
min(pima_tr,[],1))',0,size(pima_te,2),size(pima_te,2));

% Step 4: train the data
model = svmtrain(label_tr, sparse(pima_trs), '-c 16 -g 0.1');

% Step 5: test the data
[predict_labels, accuracy, dec_value_s] = svmpredict(label_te, sparse(pima_tes),
model);

% Step 6: Evaluate the performance of using unscaled data
model = svmtrain(label_tr, sparse(pima_tr), '-c 2 -g 0.1');
[predict_label, accuracy, dec_value] = svmpredict(label_te, sparse(pima_te), model);
```

THE UNIVERSITY OF TENNESSEE KNOXVILLE