Can Support Vector Machine be a Major Classification Method?

Chih-Jen Lin

Department of Computer Science
National Taiwan University

Talk at Max Planck Institute, January 29, 2003
Motivation

- SVM: a hot machine learning issue
- However, not a major classification method yet
  KD Nuggets 2002 Poll: Neural Networks, Decision trees remain main tools
- How to make SVM a major one?
The Potential of SVM

- In my opinion, after careful data pre-processing
  Appropriately use NN or SVM $\Rightarrow$ similar accuracy
- But, users may not use them properly
- The chance of SVM
  Easier for users to appropriately use it
  The ambition: replacing NN on some applications
What Many Users are Doing Now

- Transfer data to the format of an SVM software
- May not conduct scaling
- Randomly try few parameters and kernels without validation
- Default parameters are surprisingly important
- If most users doing so, accuracy may not be satisfactory
We Hope Users At Least Do

- The following procedure
  1. Simple scaling (training and testing)
  2. Consider the RBF kernel

\[
K(x, y) = e^{-\gamma \|x-y\|^2} = e^{-\|x-y\|^2/(2\sigma^2)}
\]

and find the best $C$ and $\gamma$ (or $\sigma^2$)

- Why RBF:
  - Linear kernel: special case of RBF [Keerthi and Lin 2003]
  - Polynomial: numerical difficulties
    \[
    (< 1)^d \rightarrow 0, (> 1)^d \rightarrow \infty
    \]
  - tanh: still a mystery
    In general not PD

Chih-Jen Lin, National Taiwan University
In a coming paper [Lin and Lin 2003], for certain parameters, it behaves like RBF
Examples of the Proposed Procedure

• User 1:
  I am using libsvm in a astroparticle physics application (AMANDA experiment). First, let me congratulate you to a really easy to use and nice package.
  Unfortunately, it gives me astonishingly bad results...

• Answer:
  What is your procedure?

• User 1:
  I do for example the following steps (here for classification):
  ./svm-scale -l -1. -u +1. TRAINING.DAT

Chih-Jen Lin, National Taiwan University
>TRAINING.SCALE.DAT
./svm-train -s 0 -t 2 -c 10 TRAINING.SCALE.DAT
./svm-predict TESTING_SIGNAL.SCALE.DAT
TRAINING.SCALE.DAT.model s_0_2_10.out
Accuracy = 75.2%

• Answer:
  OK. Send me the data

• Answer:
  First I scale the training and testing TOGETHER:
  /mnt/professor/cjlin/tmp% libsvm-2.36/svm-scale
total > total.scale
Then separate them again.
Using the model selection tool (cross validation) to find out the best parameter:
/mnt/professor/cjlin/tmp%python grid.py train
sort the results: (find the best cv accuracy)
/mnt/professor/cjlin/tmp% sort -k 3 train.out

2 1 96.9569
8 1 96.9569
so c = 4 and g = 1 might be the best.
Train the training data again:
/mnt/professor/cjlin/tmp/libsvm-2.36%/svm-train -m 300 -c 4 -g 2 ../train
Finally test the independent data:
/mnt/professor/cjlin/tmp/libsvm-2.36%/svm-predict ../testdata train.model o Accuracy = 97.3

• User 1:
  You earned a copy of my PhD thesis

• User 2:
I am a developer in a bioinformatics laboratory at ... We would like to use LIBSVM in a project ... The datasets are reasonable unbalanced - there are 221 examples in the first set, 117 in the second set and 53 in the third set.

But results not good

• Answer:
  Have you scaled the data? What is your accuracy?

• User 2: Yes, to [0,1]. 36%

• Answer:
  OK. Send me the data

• Answer:
  I am able to give 83.88% cv accuracy. Is that good enough for you?
• User 2:

83.88% accuracy would be excellent...
Model Selection is Important

- In fact, two-parameter search
- By bounds of loo
- By two line search
- By grid search
Bound of loo

- Many loo bounds
- Main reason: save computational cost
- Bounds where a path may be found
- Radius margin bound
- Span bound

- A recent paper [Chung et al. 2002] on radius margin bound
  - Minima in a good region more important than tightness
    Good bound should avoid that minima happen at the boundary (i.e., too small or too large $C$ and $\sigma^2$)
  - Modification for L1-SVM
  - Differentiability
    \[ \min_{C,\sigma^2} f(\alpha(C, \sigma^2)) \]
  - Reliable Implementation
<table>
<thead>
<tr>
<th></th>
<th>L1-SVM</th>
<th>L2-SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#fun</td>
<td>#grad</td>
</tr>
<tr>
<td>banana</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>image</td>
<td>17</td>
<td>13</td>
</tr>
<tr>
<td>splice</td>
<td>13</td>
<td>12</td>
</tr>
<tr>
<td>tree</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>waveform</td>
<td>16</td>
<td>13</td>
</tr>
<tr>
<td>ijcnn1</td>
<td>9</td>
<td>9</td>
</tr>
</tbody>
</table>

- A coming paper [Chang and Lin 2003]: non-smooth optimization techniques for bounds
  - Allow us to use more (i.e. non-differentiable) bounds
  - Sensitive analysis
  - Nonsmooth Optimization

Chih-Jen Lin, National Taiwan University
- Boundle (cutting plane) methods

Piecewise diff. $\rightarrow$ Semi-smooth $\uparrow$ Directionally diff.
$\downarrow$ Locally Lipschitz cont.
Two Line Searches

- CV (loo) contour of RBF kernel [Keerthi and Lin 2003]:
  \[ \log \sigma^2 = \log C - \log \tilde{C} \]

  ![Diagram](image)

- When \( \sigma^2 \) large
  \[(C, \sigma^2)\] of RBF \(\equiv C/\sigma^2\) of linear

- A heuristic for model selection
  1. Search for the best \(C\) of Linear SVM and call it \(\tilde{C}\).
2. Fix $\tilde{C}$ and search for the best $(C, \sigma^2)$ satisfying
   \[ \log \sigma^2 = \log C - \log \tilde{C} \]
   using RBF

<table>
<thead>
<tr>
<th>Problem</th>
<th>$n$</th>
<th>#test</th>
<th>Test error of grid method</th>
<th>Test error of new method</th>
</tr>
</thead>
<tbody>
<tr>
<td>banana</td>
<td>400</td>
<td>4900</td>
<td>0.1235 (6,-0)</td>
<td>0.1178 (-2,-2)</td>
</tr>
<tr>
<td>image</td>
<td>1300</td>
<td>1010</td>
<td>0.02475 (9,4)</td>
<td>0.02475 (1,0.5)</td>
</tr>
<tr>
<td>splice</td>
<td>1000</td>
<td>2175</td>
<td>0.09701 (1,4)</td>
<td>0.1011 (0.4)</td>
</tr>
<tr>
<td>ringnorm</td>
<td>400</td>
<td>7000</td>
<td>0.01429(-2,2)</td>
<td>0.018 (-3,2)</td>
</tr>
<tr>
<td>twonorm</td>
<td>400</td>
<td>7000</td>
<td>0.031 (1,3)</td>
<td>0.02914 (1,4)</td>
</tr>
<tr>
<td>tree</td>
<td>700</td>
<td>11692</td>
<td>0.1132 (8,4)</td>
<td>0.1246 (2,2)</td>
</tr>
<tr>
<td>adult</td>
<td>1605</td>
<td>29589</td>
<td>0.1614 (5,6)</td>
<td>0.1614 (5,6)</td>
</tr>
<tr>
<td>web</td>
<td>2477</td>
<td>38994</td>
<td>0.02223 (5,5)</td>
<td>0.02223 (5,5)</td>
</tr>
</tbody>
</table>

- 441 verses 54 SVMs
However, I Prefer Simple Grid Search

- Reasons for not using bounds (if two parameters)
  - Psychologically, not feel safe
  - In practice: IJCNN competition:
    - 97.09% and 97.83% using RM bounds for L1 and L2-SVM
    - 98.59% using 25-point grid
    - 2668, 1990, and 1293 testing errors
  - Useful if more than 2 parameters

- About two-line search:
  - Solving linear not as easy as we thought:
- A paper [Chung et al. 2003]: efficient decomposition methods for linear SVMs
- Decision of the best $C$ for linear SVMs sometimes ambiguous
After $C \geq C^*$, everything is the same

- We propose that users do
  - Start from a loose grid
  - Identify good regions and finer grid

- The grid search tool in libsvm
- Easy parallelization
  Every problem is independent
loo bounds: 20 steps ⇒ more time than $10 \times 10$ grids with five computers

Automatic load balancing

- No need for $\alpha$-seeding, passing cache etc.

- This simple tool
  - Enough for median-sized problems
  - Advantage of having only one figure for multi-class problems

- Further improvement
  
  Possible but many considerations
Challenges

• Using this, if for *enough* problems, satisfactory results obtained
  ⇒ then SVM can be a major method eventually
  How do we ask users to at least do this?
  How do we know if it is or not?

• If not
  What is the next general thing to be added for users?