Low Bias Bagged Support Vector Machines

Giorgio Valentini
Dipartimento di Scienze dell Informazione
Università degli Studi di Milano, Italy
valentini@dsi.unimi.it

Thomas G. Dietterich
Department of Computer Science
Oregon State University
Corvallis, Oregon 97331 USA
http://www.cs.orst.edu/~tgd
Two Questions:

- Can bagging help SVMs?
- If so, how should SVMs be tuned to give the best bagged performance?
The Answers

- Can bagging help SVMs?
  - Yes
- If so, how should SVMs be tuned to give the best bagged performance?
  - Tune to minimize the bias of each SVM
SVMs

minimize: \[ ||w||^2 + C \sum \xi_i \]
subject to: \[ y_i (w \cdot x_i + b) + \xi_i \leq 1 \]

- Soft Margin Classifier
  - Maximizes VC dimension subject to soft separation of the training data
  - Dot product can be generalized using kernels \( K(x_j, x_i; \sigma) \)
  - Set \( C \) and \( \sigma \) using an internal validation set
- Excellent control of the bias/variance tradeoff: Is there any room for improvement?
Bias/Variance Error Decomposition for Squared Loss

- For regression problems, loss is \((\hat{y} - y)^2\)
  - \(\text{error}^2 = \text{bias}^2 + \text{variance} + \text{noise}\)
  - \(E_S[(\hat{y}-y)^2] = (E_S[\hat{y}] - f(x))^2 + E_S[(\hat{y} - E_S[\hat{y}])^2] + E[(y - f(x))^2]\)
- Bias: Systematic error at data point \(x\) averaged over all training sets \(S\) of size \(N\)
- Variance: Variation around the average
- Noise: Errors in the observed labels of \(x\)
Example: 20 points
\[ y = x + 2 \sin(1.5x) + N(0,0.2) \]
Example: 50 fits
(20 examples each)
Bias
Variance
Noise
Variance Reduction and Bagging

- Bagging attempts to simulate a large number of training sets and compute the average prediction $y_m$ of those training sets.
- It then predicts $y_m$.
- If the simulation is good enough, this eliminates all of the variance.
Bias and Variance for 0/1 Loss
(Domingos, 2000)

- At each test point $x$, we have 100 estimates: $\hat{y}_1, \ldots, \hat{y}_{100} \in \{-1, +1\}$
- Main prediction: $y_m = \text{majority vote}$
- $\text{Bias}(x) = 0$ if $y_m$ is correct and 1 otherwise
- $\text{Variance}(x) = \text{probability that } \hat{y} \neq y_m$
  - Unbiased variance $V_U(x)$: variance when $\text{Bias} = 0$
  - Biased variance $V_B(x)$: variance when $\text{Bias} = 1$
- $\text{Error rate}(x) = \text{Bias}(x) + V_U(x) - V_B(x)$
- Noise is assumed to be zero
Good Variance and Bad Variance

- Error rate($x$) = $\text{Bias}(x) + V_U(x) - V_B(x)$
- $V_B(x)$ is “good” variance, but only when the bias is high
- $V_U(x)$ is “bad” variance
- Bagging will reduce both types of variance. This gives good results if $\text{Bias}(x)$ is small.
- Goal: Tune classifiers to have small bias and rely on bagging to reduce variance
Lobag

- **Given:**
  - Training examples \( \{(x_i, y_i)\}_{i=1}^N \)
  - Learning algorithm with tuning parameters \( \alpha \)
  - Parameter settings to try \( \{\alpha_1, \alpha_2, \ldots\} \)

- **Do:**
  - Apply internal bagging to compute out-of-bag estimates of the bias of each parameter setting. Let \( \alpha^* \) be the setting that gives minimum bias
  - Perform bagging using \( \alpha^* \)
Experimental Study

- Seven data sets: P2, waveform, grey-landsat, spam, musk, letter2 (letter recognition ‘B’ vs ‘R’), letter2+noise (20% added noise)
- Three kernels: dot product, RBF ($\sigma =$ gaussian width), polynomial ($\sigma =$ degree)
- Training set: 100 examples
- Bias and variance estimated on test set from 100 replicates
Example:
Letter2, RBF kernel, $\sigma = 100$
Results: Dot Product Kernel

![Graph showing the change in error rate for different datasets using bagging and lobag methods. The x-axis represents different datasets such as P2, waveform, landsat, letter2letter2+noise, spam, and musk. The y-axis shows the change in error rate ranging from -0.03 to -0.005. The graph compares the performance of single, bagging, and lobag methods.]
Results (2): Gaussian Kernel

![Graph showing change in error rate for different datasets with bagging and single models.](image)
Results (3): Polynomial Kernel
McNemar’s Tests: Bagging versus Single SVM

Win Rates:
- Linear: 70%
- Polynomial: 60%
- RBF: 50%

Tie Rates:
- Linear: 20%
- Polynomial: 30%
- RBF: 40%

Loss Rates:
- Linear: 10%
- Polynomial: 10%
- RBF: 10%
McNemar’s Test: Lobag versus Single SVM

- Linear
- Polynomial
- RBF
McNemar’s Test: Lobag versus Bagging
Results: McNemar’s Test (wins – ties – losses)

<table>
<thead>
<tr>
<th>Kernel</th>
<th>Lobag vs Bagging</th>
<th>Lobag vs Single</th>
<th>Bagging vs Single</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>3 – 26 – 1</td>
<td>23 – 7 – 0</td>
<td>21 – 9 – 0</td>
</tr>
<tr>
<td>Polynomial</td>
<td>12 – 23 – 0</td>
<td>17 – 17 – 1</td>
<td>9 – 24 – 2</td>
</tr>
<tr>
<td>Gaussian</td>
<td>17 – 17 – 1</td>
<td>18 – 15 – 2</td>
<td>9 – 22 – 4</td>
</tr>
</tbody>
</table>
Discussion

- For small training sets
  - Bagging can improve SVM error rates, especially for linear kernels
  - Lobag is at least as good as bagging and often better
- Consistent with previous experience
  - Bagging works better with unpruned trees
  - Bagging works better with neural networks that are trained longer or with less weight decay
Conclusions

- Lobag is recommended for SVM problems with high variance (small training sets, high noise, many features)
- Small added cost:
  - SVMs require internal validation to set C and $\sigma$
  - Lobag requires internal bagging to estimate bias for each setting of C and $\sigma$
- Future research:
  - Smart search for low-bias settings of C and $\sigma$
  - Experiments with larger training sets