Bounds on the Generalization Performance of Kernel Machines Ensembles

Theodoros Evgeniou THEOS@AI.MIT.EDU
Luis Perez-Breva LPBREVA@AI.MIT.EDU
Massimiliano Pontil PONTIL@AI.MIT.EDU
Tomaso Poggio TP@AI.MIT.EDU
Center for Biological and Computational Learning, MIT, 45 Carleton Street E25-201, Cambridge, MA 02142

Abstract

We study the problem of learning using combinations of machines. In particular we present new theoretical bounds on the generalization performance of voting ensembles of kernel machines. Special cases considered are bagging and support vector machines. We present experimental results supporting the theoretical bounds, and describe characteristics of kernel machines ensembles suggested from the experimental findings. We also show how such ensembles can be used for fast training with very large datasets.

1. Introduction

Two major recent advances in learning theory are support vector machines (SVM) (Vapnik, 1998) and ensemble methods such as boosting and bagging (Breiman, 1996; Schapire et al., 1998). Distribution independent bounds on the generalization performance of these two techniques have been suggested recently (Shawe-Taylor & Cristianini, 1998; Bartlett, 1998; Schapire et al., 1998), and similarities between these bounds in terms of a geometric quantity known as the *margin* have been proposed. More recently bounds on the generalization performance of SVM based on cross-validation have been derived (Vapnik, 1998; Chapelle & Vapnik, 1999). These bounds depend also on geometric quantities other than the margin (such as the radius of the smallest sphere containing the support vectors).

In this paper we study the generalization performance of ensembles of general kernel machines using cross-validation arguments. The kernel machines considered are learning machines of the form

$$f(\mathbf{x}) = \sum_{i=1}^{\ell} \alpha_i K(\mathbf{x}_i, \mathbf{x}), \qquad (1)$$

where $(\mathbf{x}_i, y_i)$ $i = 1 \ldots \ell$ are the training points and K is a kernel function, for example a Gaussian - see (Vapnik, 1998; Wahba, 1990) for a number of kernels. The coefficients α_i are learned by solving the following optimization problem:

$$\begin{array}{l} \max_{\alpha} \sum_{i=1}^{\ell} S(\alpha_i) - \frac{1}{2} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j y_i y_i K_{ij} \\ \text{subject to : } \; 0 \leq \alpha_i \leq C \end{array} \qquad (2)$$

where $S(\cdot)$ is a cost function, C a constant, and we have defined $K_{ij} = K_{ij}$. SVM are a particular case of these machines for $S(\alpha) = \alpha$. For SVM, points for which $\alpha_i \neq 0$ are called support vectors. Notice that the bias term (threshold b in the general case of machines $f(\mathbf{x}) = \sum_{i=1}^{\ell} \alpha_i K(\mathbf{x}_i, \mathbf{x}) + b$) is incorporated in the kernel K.

We study the cross-validation error of general voting ensembles of kernel machines (2). Particular cases are that of bagging kernel machines each trained on different subsamples of the initial training set, or that of voting kernel machines each using a different kernel (also different subsets of features/components of the initial input features). We first present bounds on the generalization error of such ensembles, and then discuss experiments where the derived bounds are used for model selection. We also show experimental results showing that a validation set can be used for model selection for kernel machines and their ensembles, without having to decrease the training set size in order to create a validation set. Finally we show how such ensembles can be used for fast training with very large data sets.

2. Generalization Performance of Kernel Machines Ensembles

The theoretical results of the paper are based on the cross-validation (or leave-one-out) error. The cross-validation procedure consists of removing from the training set one point at a time, training a machine on the remaining points and then testing on the removed one. The number of errors counted throughout this process, $\mathcal{L}((\mathbf{x}_i, y_1), \ldots (\mathbf{x}_\ell, y_\ell))$, is called the cross-validation error. It is known that this quantity provides an estimate of the generalization performance of a machine (Wahba, 1990; Vapnik, 1998). In particular the expectation of the generalization error of a machine trained using ℓ points is bounded by the expectation of the cross validation error of a machine trained on $\ell+1$ points (Luntz and Brailovsky theorem (Vapnik, 1998)).

We begin with some known results on the cross-validation error of kernel machines. The following theorem is from (Jaakkola & Haussler, 1998) :

Theorem 2.1 *The cross-validation error of a kernel machine (2) is upper bounded as:*

$$\mathcal{L}((\mathbf{x}_i, y_1), \ldots (\mathbf{x}_\ell, y_\ell)) \leq \sum_{i=1}^{\ell} \theta(\alpha_i K_{ii} - y_i f(\mathbf{x}_i)) \quad (3)$$

where θ is the Heavyside function, and f is the optimal function found by solving maximization problem (2).

In the particular case of SVM where the data are separable (3) can be bounded by geometric quantities, namely (Vapnik, 1998):

$$\sum_{i=1}^{\ell} \theta(\alpha_i K_{ii} - y_i f(\mathbf{x}_i)) \leq \frac{D_{sv}^2}{\rho^2} \quad (4)$$

where D_{sv} is the radius of the smallest sphere in the feature space induced by kernel K (Wahba, 1990; Vapnik, 1998) centered at the origin containing the support vectors, and ρ is the margin ($\rho^2 = \frac{1}{\|f\|_K^2}$) of the SVM.

Using this result, the following theorem is a direct application of the Luntz and Brailovsky theorem (Vapnik, 1998):

Theorem 2.2 *The average generalization error of an SVM (with zero threshold b, and in the separable case) trained on ℓ points is upper bounded by*

$$\frac{1}{\ell+1} E\left(\frac{D_{sv(\ell)}^2}{\rho^2(\ell)}\right),$$

where the expectation E is taken with respect to the probability of a training set of size ℓ.

Notice that this result shows that the performance of SVM does not depend only on the margin, but also on other geometric quantities, namely the radius D_{sv}. In the non-separable case, it can be shown (the proof is similar to that of corollary 2.2 below) that equation (4) can be written as:

$$\sum_{i=1}^{\ell} \theta(\alpha_i K_{ii} - y_i f(\mathbf{x}_i)) \leq EE_1 + \frac{D_{sv}^2}{\rho^2} \quad (5)$$

where EE_1 is the hard margin empirical error of the SVM (the number of training points with $yf(\mathbf{x}) < 1$).

We now extend these results to the case of ensembles of kernel machines. We consider the general case where each of the machines in the ensemble uses a different kernel. Let T be the number of machines, and let $K^{(t)}$ be the kernel used by machine t. Notice that, as a special case, appropriate choices of $K^{(t)}$ lead to machines that may have different subsets of features from the original ones. Let $f^{(t)}(\mathbf{x})$ be the optimal solution of machine t (real-valued), and $\alpha_i^{(t)}$ the optimal weight that machine t assigns to point $(\mathbf{x}_i, y_i)$ (after solving problem (2)). We consider ensembles that are linear combinations of the individual machines. In particular, the separating surface of the ensemble is:

$$F(\mathbf{x}) = \sum_{t=1}^{T} c_t f^{(t)}(\mathbf{x}) \quad (6)$$

and the classification is done by taking the sign of this function. The coefficients c_t are not learned (i.e. $c_t = \frac{1}{T}$), and $\sum_{t=1}^{T} c_t = 1$ (for scaling reasons), $c_t > 0$. All parameters (C's and kernels) are fixed before training. In the particular case of bagging, the subsampling of the training data should be deterministic. With this we mean that when the bounds are used for model (parameter) selection, for each model the same subsample sets of the data need to be used. These subsamples, however, are still random ones. We believe that the results presented below also hold (with minor modifications) in the general case that the subsampling is always random. We now consider the cross-validation error of such ensembles.

Theorem 2.3
The cross-validation error $\mathcal{L}((\mathbf{x}_i, y_1), \ldots (\mathbf{x}_\ell, y_\ell))$ of a kernel machines ensemble is upper bounded by:

$$\sum_{i=1}^{\ell} \theta(\sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} - y_i F(\mathbf{x}_i)) \quad (7)$$

The proof of this theorem is based on the following lemma shown in Vapnik (1998) and in Jaakkola and Haussler (1998):

Lemma 2.1 *Let α_i be the coefficient of the solution $f(\mathbf{x})$ of machine (2) corresponding to point $(\mathbf{x}_i, y_i)$,*

$\alpha_i \neq 0$. *Let* $f_i(\mathbf{x})$ *be the solution of machine (2) found when point* $(\mathbf{x}_i, y_i)$ *is removed from the training set. Then:* $y_i f_i(\mathbf{x}_i) \geq y_i f(\mathbf{x}_i) - \alpha_i K_{ii}$.

Using lemma 2.1 we can now prove theorem 2.3.

Proof of theorem 2.3: Let $F_i(\mathbf{x}) = \sum_{t=1}^{T} c_t f_i^{(t)}(\mathbf{x})$ be the final machine trained with all initial training data except $(\mathbf{x}_i, y_i)$. Lemma 2.1 gives that

$$
\begin{aligned}
y_i F_i(\mathbf{x}_i) = y_i \sum_{t=1}^{T} c_t f_i^{(t)}(\mathbf{x}_i) \geq \\
\geq y_i \sum_{t=1}^{T} c_t f^{(t)}(\mathbf{x}) - \sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} = \\
= y_i F(\mathbf{x}_i) - \sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} \Rightarrow \\
\Rightarrow \theta(-y_i F_i(\mathbf{x}_i)) \leq \theta(\sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} - y_i F(\mathbf{x}_i))
\end{aligned}
$$

therefore the leave one out error $\sum_{i=1}^{\ell} \theta(-y_i F_i(\mathbf{x}_i))$ is not more than $\sum_{i=1}^{\ell} \theta(\sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} - y_i F(\mathbf{x}_i))$ which proves the theorem. □

Notice that the bound has the same form as bound (3): for each point $(\mathbf{x}_i, y_i)$ we only need to take into account its corresponding parameter $\alpha_i^{(t)}$ and "remove" the effects of $\alpha_i^{(t)}$ from the value of $F(\mathbf{x}_i)$.

The cross-validation error can also be bounded using geometric quantities. To this purpose we introduce one more parameter that we call the *ensemble margin* (in contrast to the margin of a single SVM). For each point $(\mathbf{x}_i, y_i)$ we define its ensemble margin to be simply $y_i F(\mathbf{x}_i)$. This is exactly the definition of margin in (Schapire et al., 1998). For any given $\delta > 0$ we define EE_δ to be the number of training points with ensemble margin $< \delta$ (empirical error with margin δ), and by N_δ the set of the remaining training points - the ones with ensemble margin $\geq \delta$. Finally, we note by $D_{t(\delta)}$ to be the radius of the smallest sphere in the feature space induced by kernel $K^{(t)}$ centered at the origin containing the points of machine t with $\alpha_i^{(t)} \neq 0$ and ensemble margin larger than δ (in the case of SVM, these are the support vectors of machine t with ensemble margin larger than δ). A simple consequence of theorem 2.3 and of the inequality $K_{ii}^{(t)} \leq D_{t(\delta)}^2$ for points $\mathbf{x}_i$ with $\alpha_i^{(t)} \neq 0$ and ensemble margin $y_i F(\mathbf{x}_i) \geq \delta$ is the following:

Corollary 2.1 *For any* $\delta > 0$ *the cross-validation error* $\mathcal{L}\left((\mathbf{x}_i, y_1), \ldots (\mathbf{x}_\ell, y_\ell)\right)$ *of a kernel machines ensemble is upper bounded by:*

$$
EE_\delta + \frac{1}{\delta} \left(\sum_{t=1}^{T} c_t D_{t(\delta)}^2 (\sum_{i \in N_\delta} \alpha_i^{(t)}) \right) \qquad (8)
$$

Proof: For each training point $(\mathbf{x}_i, y_i)$ with ensemble margin $y_i F(\mathbf{x}_i) < \delta$ we upper bound $\theta(\sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} - y_i F(\mathbf{x}_i))$ with 1 (this is a trivial bound). For the remaining points (the points in N_δ) we show that:

$$
\theta(\sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} - y_i F(\mathbf{x}_i)) \leq \frac{1}{\delta} \sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)}.
$$

If $\sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} - y_i F(\mathbf{x}_i) < 0$, then:

$$
\theta(\sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} - y_i F(\mathbf{x}_i)) = 0 \leq \frac{1}{\delta} \sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)}.
$$

On the other hand, if $\sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} - y_i F(\mathbf{x}_i) \geq 0$, then $\theta(\sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} - y_i F(\mathbf{x}_i)) = 1$, while

$$
\sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} \geq y_i F(\mathbf{x}_i) \geq \delta \Rightarrow \frac{1}{\delta} \sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} \geq 1.
$$

So in both cases inequality (9) holds. Therefore:

$$
\begin{aligned}
\sum_{i=1}^{\ell} \theta(\sum_{t=1}^{T} c_t \alpha_i^{(t)} K_{ii}^{(t)} - y_i F(\mathbf{x}_i)) \leq \\
\leq EE_\delta + \frac{1}{\delta} \left(\sum_{i \in N_\delta} \sum_{t=1}^{T} c_t K_{ii}^{(t)} \alpha_i^{(t)} \right) \leq \\
\leq EE_\delta + \frac{1}{\delta} \left(\sum_{t=1}^{T} c_t D_{t(\delta)}^2 (\sum_{i \in N_\delta} \alpha_i^{(t)}) \right)
\end{aligned}
$$

which proves the corollary. □

Notice that equation (8) holds for any $\delta > 0$, so the best bound is obtained for the minimum of the right hand side with respect to $\delta > 0$. Using the Luntz and Brailovsky theorem, theorems 2.3 and 2.1 provide bounds on the generalization performance of general kernel machines ensembles like that of theorem 2.2.

We now consider the particular case of SVM ensembles. In this case, for example choosing $\delta = 1$ (8) becomes:

Corollary 2.2 *The leave-one-out error of an ensemble of SVMs is upper bounded by:*

$$
\mathcal{L}\left((\mathbf{x}_i, y_1), \ldots (\mathbf{x}_\ell, y_\ell)\right) \leq EE_1 + \sum_{t=1}^{T} c_t \frac{D_t^2}{\rho_t^2} \qquad (9)
$$

where EE_1 is the margin empirical error with ensemble margin 1, D_t is the radius of the smallest sphere centered at the origin, in the feature space induced by kernel $K^{(t)}$, containing the support vectors of machine t, and ρ_t is the margin of SVM t.

This is because clearly $D_t \geq D_{t(\delta)}$ for any δ, and $\sum_{i \in N_\delta} \alpha_i^{(t)} \leq \sum_{i=1}^{\ell} \alpha_i^{(t)} = \frac{1}{\rho_t^2}$ (see (Vapnik, 1998) for a proof of this equality). A number of remarks can be made from equation (9).

First notice that the generalization performance of the SVM ensemble now depends on the "average" (convex combination of) $\frac{D^2}{\rho^2}$ of the individual machines. In some cases this may be smaller than the $\frac{D^2}{\rho^2}$ of a single SVM. For example, suppose we train many SVMs on different subsamples of the training points and we want to compare such an ensemble with a single SVM using all the points. If all SVMs (the single one, as well as the individual ones of the ensemble) use most of their training points as support vectors, then clearly the D^2 of each SVM in the ensemble is smaller than that of the single SVM. Moreover the margin of each SVM in the ensemble is expected to be larger than that of the single SVM using all the points. So the "average" $\frac{D^2}{\rho^2}$ in this case is expected to be smaller than that of the single SVM. Another case where an ensemble of SVMs may be better than a single SVM is the one where there are outliers among the training data: if the individual SVMs are trained on subsamples of the training data, some of the machines may have smaller $\frac{D^2}{\rho^2}$ because they do not use some outliers. In general it is not clear when ensembles of kernel machines are better than single machines. The bounds in this section may provide some insight to this question.

Notice also how the ensemble margin δ plays a role for the generalization performance of kernel machine ensembles. This margin is also shown to be important for boosting (Schapire et al., 1998). Finally, notice that all the results discussed hold for the case that there is no bias (threshold b), or the case where the bias is included in the kernel (as discussed in the introduction). In the experiments discussed below we use the results also for cases where the bias is not regularized, which is common in practice. It may be possible to use recent theoretical results (Chapelle & Vapnik, 1999) on the leave-one-out bounds of SVM when the bias b is taken into account in order to study the generalization performance of kernel machines ensembles with the bias b.

3. Experiments

To test how tight the bounds we presented are, we conducted a number of experiments using data sets from UCI,[1] as well as the US Postal Service (USPS) data set (LeCun et al., 1990) . We show results for some of the sets in Figures 1-2. For each data set we split the overall set in training and testing (the sizes are shown in the figures) in 50 different (random) ways, and for each split:

1. We trained one SVM with $b = 0$ using all training data, computed the leave-one-bound given by theorem 2.1, and then compute the test performance using the test set.

2. We repeated (1) this time with with $b \neq 0$.

3. We trained 30 SVMs with $b = 0$ each using a random subsample of size 40% of the training data (bagging), computed the leave-one-bound given by theorem 2.3 using $c_t = \frac{1}{30}$, and then compute the test performance using the test set.

4. We repeated (3) this time with with $b \neq 0$.

We then averaged over the 50 training-testing splits the test performances and the leave-one-out bounds found, and computed the standard deviations. All machines were trained using a Gaussian kernel, and we repeated the procedure for a number of different σ's of the Gaussian, and for a *fixed* C (show in the figures). We show the averages and standard deviations of the results in the figures. In all figures we use the following notation: top left figure: bagging with $b = 0$; top right figure: single SVM with $b = 0$; bottom left figure: bagging with $b \neq 0$; and bottom right figure: single SVM with $b \neq 0$. In all plots the solid line is the mean test performance and the dashed line is the error bound computed using the leave-one-out theorems (theorems 2.1 and 2.3). The dotted line is the validation set error discussed below. For simplicity, only one error bar (standard deviation over the 50 training-testing splits) is shown (the others were similar). The cost parameter C used is given in each of the figures. The horizontal axis is the natural logarithm of the σ of the Gaussian kernel used, while the vertical axis is the error.

An interesting observation is that *the bounds are always tighter for the case of bagging than they are for the case of a single SVM*. This is an interesting experimental finding for which we do not have a theoretical explanation. It may be because the generalization performance of a machine is related to the *expected* leave-one-out error of the machine (Vapnik,

[1] `http://www.ics.uci.edu/ mlearn/MLRepository.html`

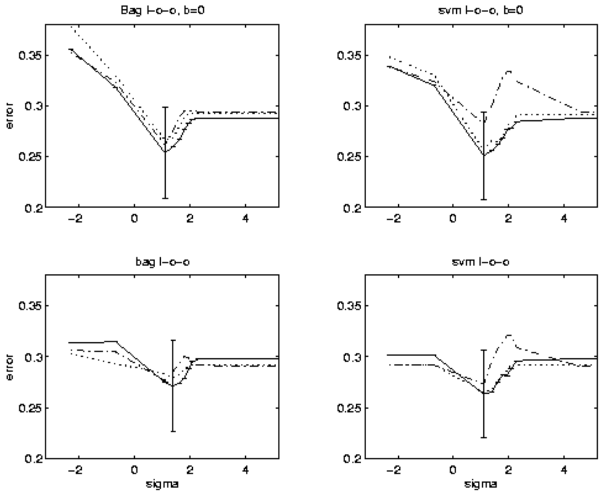

Figure 1. Breast cancer data: see text for description.

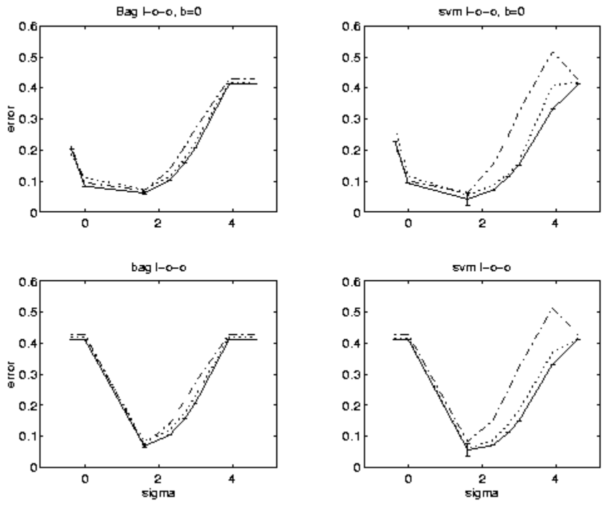

Figure 2. USPS data: see text for description.

1998), and by combining many machines each using a different (random) subset of the training data we better approximate the "expected" leave-one-out than we do when we only compute the leave-one-out of a single machine. This finding can practically justify the use of ensembles of machines for model selection: parameter selection using the leave-one-out bounds presented in this paper is easier for ensembles of machines than it is for single machines.

Another interesting observation is that the bounds seem to work similarly in the case that the bias b is not 0. In this case, as before, the bounds are tighter for ensembles of machines than they are for single machines.

In all our experiments we always found that the bounds presented here (equations (3) and (7)) get looser in the case that the parameter C used during the training, is large. A representative example is shown in figure 3. This result can be understood looking at the leave-one-out bound for a single SVM (equation (3)). Let $(\mathbf{x}_i, y_i)$ be a support vector for which $y_i f(\mathbf{x}_i) < 1$. It is known (Vapnik, 1998) that for these support vectors the coefficient α_i is C. If C is such that $CK_{ii} > 1$ (for the Gaussian kernel this reduces to $C > 1$, as $K(\mathbf{x}, \mathbf{x}) = 1$), then clearly $\theta(CK_{ii} - y_i f(\mathbf{x}_i)) = 1$. In this case the bound in equation (3) effectively counts *all support vectors with margin less then one* (plus some of the ones *on* the margin - $yf(\mathbf{x}) = 1$). This means that for "large" C (in the case of Gaussian kernels this can be for example for any $C > 1$), the bounds of this paper effectively are similar (not larger than) to another known leave-one-out bound for SVMs, namely one that uses the number of all support vectors to bound generalization performance (Vapnik, 1998). So effectively our experimental results show that *the number of support vectors does not provide a good estimate of the generalization performance of the SVMs and their ensembles.*

4. Validation Set for Model Selection

Instead of using bounds on the generalization performance of learning machines like the ones discussed above, an alternative approach for model selection is to use a validation set to choose the parameters of the machines. We consider first the simple case where we have N machines and we choose the "best" one based on the error they make on a fixed validation set of size V. This can be thought of as a special case where we consider as our hypothesis space to be the set of the N machines, and then we "train" by simply picking the machine with the smallest "empirical" error (in this case this is the validation error). It is known that if VE_i is the validation error of machine i and TE_i is its true test error, then for all N machines simultaneously the following bound holds with probability $1 - \eta$ (Devroye et al., 1996; Vapnik, 1998) :

$$TE_i \leq VE_i + \sqrt{\frac{\log(N) - \log(\frac{\eta}{4})}{V}} \tag{10}$$

So how "accurately" we pick the best machine using the validation set depends, as expected, on the number of machines N and on the size V of the validation set. The bound suggests that a validation set can be used to accurately estimate the generalization performance of a relatively small number of machines (i.e. small number of parameter values examined), as done often in practice.

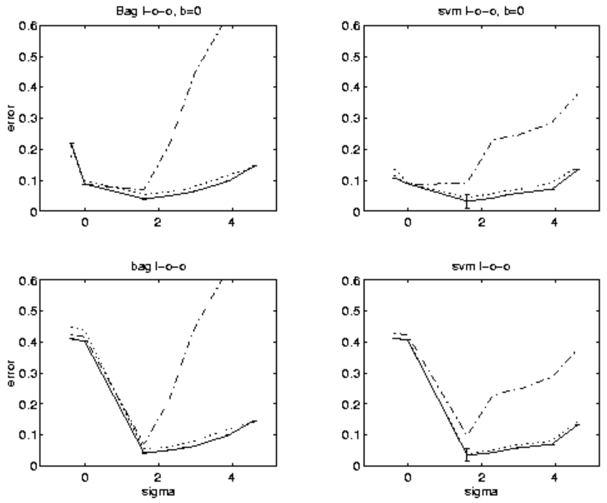

Figure 3. USPS data: using a large C (C=50). In this case the bounds do not work; see text for an explanation.

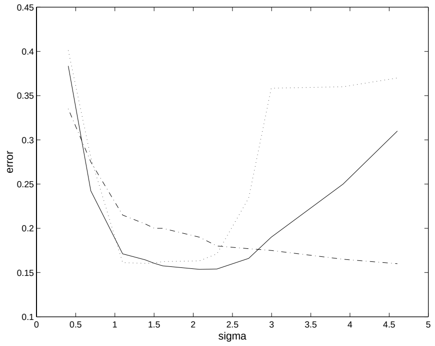

Figure 4. When the coefficients of the second layer are learned using a linear SVM the system is less sensitive to changes of the σ of the Gaussian kernel used by the individual machines of the ensemble. Solid line is one SVM, dotted is ensemble of 30 SVMs with fixed $c_t = \frac{1}{30}$, and dashed line is ensemble of 30 SVMs with the coefficients c_t learned. The horizontal axis shows the natural logarithm of the σ of the Gaussian kernel. The data use is the heart data set. The threshold b is non-zero for this experiment.

We used this observation for parameter selection for SVM and for their ensembles. Experimentally we followed a slightly different procedure from what is suggested by bound (10): for each machine (that is, for each σ of the Gaussian kernel in our case, both for a single SVM and for an ensemble of machines) we split the training set (for each training-testing split of the overall data set as described above) into a smaller training set and a validation set (70-30% respectively). We trained each machine using the new, smaller training set, and measured the performance of the machine on the validation set. Unlike what bound (10) suggests, instead of comparing the validation performance found with the generalization performance of the machines trained on the smaller training set (which is the case for which bound (10) holds), we compared the validation performance with the test performance of the machine trained using *all* the initial (larger) training set. This way *we did not have to use less points for training the machines*, which is a typical drawback of using a validation set, and we could compare the validation performance with the leave-one-out bounds and the test performance of the *exact same* machines we used in the previous section.

We show the results of these experiments in figures 1-2: see the dotted lines in the plots. We observe that *although the validation error is that of a machine trained on a smaller training set, it still provides a very good estimate of the test performance of the machines trained on the whole training set.* In all cases, including the case of $C > 1$ for which the leave-one-out bounds discussed above did not work well, the validation set error provided a very good estimate of the test performance of the machines.

5. Other Ensembles

The ensemble kernel machines (6) considered so far are voting combinations where the coefficients c_t in (6) of the linear combination of the machines are fixed. We now consider the case where these coefficients are also learned from the training subsets. In particular we consider the following architecture:

a. A number T of kernel machines is trained as before (for example using different training data, or different parameters).

b. The T outputs (real valued in our experiments, but could also be thresholded - binary) of the machines at each of the training points are computed.

c. A linear machine (i.e. linear SVM) is trained using as inputs the outputs of the T machines on the training data, and as labels the original training labels. The solution is used as the coefficients c_t of the linear combination of the T machines.

Notice that for this type of machines the leave-one-out bound of theorem 2.3 does not hold since the theorem assumes fixed coefficients c_t. A validation set can still be used for model selection for these machines. On the other hand, an important characteristic of this type of ensembles is that independent of what kernels/parameters each of the individual machines of the ensemble use, the "second layer" machine (which finds coefficients c_t) uses always a linear kernel. This may imply that *the overall architecture may not be very sensitive to the kernel/parameters of the machines of the ensemble.* We tested this hypothesis experimentally by comparing how the test performance of this type of machines changes with the σ of the Gaussian kernel used from the individual machines of the ensemble, and compared the behavior with that of single machines and ensembles of machines with fixed c_t. In figure 5 we show two example. In our experiments, for all data sets except from one, learning the coefficients c_t of the combination of the machines using a linear machine (we used a linear SVM) made the overall machine *less sensitive* to changes of the parameters of the individual machines (σ of the Gaussian kernel). This can be practically a useful characteristic of the architecture outlined in this section: for example the choice of the kernel parameters of the machines of the ensembles need not be tuned accurately.

6. Ensembles versus Single Machines

So far we concentrated on the theoretical and experimental characteristics of ensembles of kernel machines. We now discuss how ensembles compare with single machines.

Table 1 shows the test performance of one SVM compared with that of an ensemble of 30 SVMs combined with $c_t = \frac{1}{30}$ and an ensemble of 30 SVMs combined using a linear SVM for some UCI data sets (characteristic results). For the tables of this section we use, for convenience, the following notation:

VCC stands for "Voting Combinations of Classifiers", meaning that the coefficients c_t of the combination of the machines are fixed.

ACC stands for "Adaptive Combinations of Classifiers", meaning that the coefficients c_t of the combination of the machines are learned-adapted.

We only consider SVM and ensembles of SVMs with the threshold b. The table shows mean test errors and standard deviations for the best (decided using the validation set performance in this case) parameters of the machines (σ's of Gaussians *and* parameter C - hence different from figures 1-2 which where for a given C). As the results show, the best SVM and the best ensembles we found have about the same test performance. Therefore, with appropriate tuning of the parameters of the machines, combining SVM's does not lead to performance improvement compared to a single SVM. Although the "best" SVM and the "best" ensemble (that is, after accurate parameter tuning) perform similarly, an important difference of the ensembles compared to a single machine is that the training of the ensemble consists of a large number of (parallelizable) small-training-set kernel machines - in the case of bagging. This implies that one can gain performance similar to that of a single machine by training many faster machines using smaller training sets. This can be an important practical advantage of ensembles of machines especially in the case of large data sets. Table 2 compares the test performance of a single SVM with that of an ensemble of SVM each trained with as low as 1% of the initial training set (for one data set). For fixed c_t the performance decreases only slightly in all cases (thyroid, that we show, was the only data set we found in our experiments for which the change was significant for the case of VCC), while in the case of the architecture of section 5 even with 1% training data the performance does not decrease: this is because the linear machine used to learn coefficients c_t uses all the training data. Even in this last case the overall machine can still be faster than a single machine, since the second layer learning machine is a linear one, and fast training methods for the particular case of linear machines exist (Platt, 1998). Finally, it may be the case that ensembles of machines perform better for some problems in the presence of outliers (as discussed in section 3.1), or, if the ensemble consists of machines that use different kernels and/or different input features, in the presence of irrelevant features. The leave-one-out bounds presented in this paper may be used for finding these cases and for better understanding how bagging and general ensemble methods work (Breiman, 1996; Schapire et al., 1998) .

Table 1. Average errors and standard deviations (percentages) of the "best" machines (best σ of the Gaussian kernel and best C) - chosen according to the validation set performances. The performances of the machines are about the same. VCC and ACC use 30 SVM classifiers.

Data Set	**SVM**	**VCC**	**ACC**
Breast	25.5 ± 4.3	25.6 ± 4.5	25 ± 4
thyroid	5.1 ± 2.5	5.1 ± 2.1	4.6 ± 2.7
diabetes	23 ± 1.6	23.1 ± 1.4	23 ± 1.8
heart	15.4 ± 3	15.9 ± 3	15.9 ± 3.2

Table 2. Comparison between error rates of a single SVM v.s. error rates of VCC and ACC of 100 SVMs for different percentages of subsampled data. The last data set is from (Osuna et al., 1997) .

Data Set	**VCC 5%**	**VCC 1%**	**SVM**
Diabetes	26.2	-	23 ± 1.6
Thyroid	22.2	-	5.1 ± 2.5
Faces	.2	.5	.1

7. Conclusions

We presented theoretical bounds on the generalization error of ensembles of kernel machines. Our results apply to the quite general case where each of the machines in the ensemble is trained on different subsets of the training data and/or uses different kernels or input features. Experimental results supporting our theoretical findings have been presented.

A number of observations have been made from the experiments. We summarize some of them below:

1. The leave-one-out bounds for ensembles of machines have a form similar to that of single machines. In the particular case of SVMs, the bounds are based on an "average" geometric quantity of the individual SVMs of the ensemble (average margin and average radius of the sphere containing the support vectors).

2. The leave-one-out bounds presented are experimentally found to be tighter than the equivalent ones for single machines.

3. For SVM, the leave-one-out bounds based on the number of support vectors are experimentally found not to be tight.

4. Experimentally we found that a validation set can be used for accurate model selection without having to decrease the size of the training set used in order to create a validation set.

5. With accurate parameter tuning (model selection) single SVMs and ensembles of SVMs perform similarly.

6. Ensembles of machines for which the coefficients of combining the machines are also learned from the data are less sensitive to changes of parameters (i.e. kernel) than single machines are.

7. Fast (parallel) training without significant loss of performance relatively to single whole-large-training-set machines can be achieved using ensembles of machines.

A number of questions and research directions are open. An important theoretical question is how the bounds and experiments presented in this paper are related with Breiman's (1996) bias-variance analysis of bagging. For example, the fact that single SVM perform about the same as ensembles of SVMs may be because SVMs are stable machines, i.e., they have small variance (Breiman, 1996). Experiments on the bias-variance decomposition of SVM and kernel machines may lead to further understanding of these machines. The theoretical results we presented here approach the problem of learning with ensembles of machines from a different perspective - from that of studying the leave-one-out geometric bounds of these machines. On the practical side, further experiments using very large data sets are needed to support our experimental finding that the ensembles of machines can be used for fast training without significant loss in performance. Finally, other theoretical questions are how to extend the bounds of section 2 to the type of machines discussed in section 5, and how to use more recent leave-one-out bounds for SVM (Chapelle & Vapnik, 1999) to better characterize the performance of ensembles of machines.

References

Bartlett, P. (1998). The sample complexity of pattern classification with neural networks: the size of the weights is more important that the size of the network. *IEEE Transactions on Information Theory*, 44, 525–536.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 26, 123–140.

Chapelle, O., & Vapnik, V. (1999). Model selection for support vector machines. *Proceedings of the Twelfth Conference on Neural Information Processing Systems.*

Devroye, L., Györfi, L., & Lugosi, G. (1996). *A probabilistic theory of pattern recognition.* New York, NY: Springer.

Jaakkola, T., & Haussler, D. (1998). Exploiting generative models in discriminative classifiers. *Proceedings of the Eleventh Conference on Neural Information Processing Systems.*

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R.E., Hubbard, W., & Jackel, L.J. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1, 541–551.

Osuna, E., Freund, R., & Girosi, F. (1997). Support vector machines: Training and applications. A.I. Memo 1602, AI Laboratory, Massachusetts Institute of Technology, Cambridge, MA.

Platt, J. (1998). Fast training of support vector machines using sequential minimal optimization. In C. Burges & B. Scholkopf, (Eds.), *Advances in kernel methods–support vector learning.* Cambridge, MA: MIT Press.

Schapire, R., Freund, Y., Bartlett, P. & Lee, W.S. (1998). Boosting the margin: A new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 26, 824–832.

Shawe-Taylor, J. & Cristianini, N. (1998). Robust bounds on generalization from the margin distribution (Technical Report NC2-TR-1998-029). NeuroCOLT2, Royal Holloway, University of London, UK.

Vapnik, V. (1998). *Statistical learning theory.* New York, NY: Wiley.

Wahba, G. (1990). *Splines models for observational data.* Series in Applied Mathematics, Vol. 59, SIAM, Philadelphia.