

Rademacher Analysis and Multi-view Classification

John Shawe-Taylor

School of Electronics and Computer Science

University of Southampton

`jst@ecs.soton.ac.uk`

May, 2006

BCS/RSS Meeting, May 2006

STRUCTURE

1. General Statistical Considerations
2. Real-valued Function Classes and the Margin
3. Concentration Inequalities
4. Rademacher complexity and Main Theory
5. Applications to classification
6. SVM-2K for multi-view learning
7. Conclusions

Aim:

- Some thoughts on why theory and frequentist approach
- Introduction to Rademacher complexity
- Complete proof of SVM bound using Rademacher approach
- Theory and application of multi-view learning

Theories of learning

- Basic approach of SLT is to view learning from a statistical viewpoint.
- Aim of any theory is to model real/ artificial phenomena so that we can better understand/ predict/ exploit them.
- SLT is just one approach to understanding/ predicting/ exploiting learning systems, others include Bayesian inference, inductive inference, statistical physics, traditional statistical analysis.

Theories of learning cont.

- Each theory makes assumptions about the phenomenon of learning and based on these derives predictions of behaviour as well as algorithms that aim at optimising the predictions.
- Each theory has strengths and weaknesses – the better it captures the details of real world experience, the better the theory and the better the chances of it making accurate predictions and driving good algorithms.

General statistical considerations

- Statistical models (not including Bayesian) begin with an assumption that the data is generated by an underlying distribution P typically not given explicitly to the learner.
- If we are trying to classify cancerous tissue from healthy tissue, there are two distributions, one for cancerous cells and one for healthy ones.

General statistical considerations cont.

- Usually the distribution subsumes the processes of the natural/artificial world that we are studying.
- Rather than accessing the distribution directly, statistical learning typically assumes that we are given a ‘training sample’ or ‘training set’

$$S = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_m, y_m)\}$$

generated identically and independently (i.i.d.) according to the distribution P .

Generalisation of a learner

- Assume that we have a learning algorithm \mathcal{A} that chooses a function $\mathcal{A}_{\mathcal{F}}(S)$ from a function space \mathcal{F} in response to the training set S .
- From a statistical point of view the quantity of interest is the random variable:

$$\epsilon(S, \mathcal{A}, \mathcal{F}) = \mathbb{E}_{(\mathbf{x}, y)} [\ell(\mathcal{A}_{\mathcal{F}}(S), \mathbf{x}, y)],$$

where ℓ is a ‘loss’ function that measures the discrepancy between $\mathcal{A}_{\mathcal{F}}(S)(\mathbf{x})$ and y .

Generalisation of a learner

- For example, in the case of classification ℓ is 1 if the two disagree and 0 otherwise, while for regression it could be the square of the difference between $\mathcal{A}_{\mathcal{F}}(S)(\mathbf{x})$ and y .
- We refer to the random variable $\epsilon(S, \mathcal{A}, \mathcal{F})$ as the generalisation of the learner.

Example of Generalisation I

- We consider the Breast Cancer dataset from the UCI repository.
- Use the simple Parzen window classifier described by Bernhard Schölkopf: weight vector is

$$\mathbf{w}^+ - \mathbf{w}^-$$

where \mathbf{w}^+ is the average of the positive training examples and \mathbf{w}^- is average of negative training examples. Threshold is set so hyperplane bisects the line joining these two points.

Example of Generalisation II

- Given a size m of the training set, by repeatedly drawing random training sets S we estimate the distribution of

$$\epsilon(S, \mathcal{A}, \mathcal{F}) = \mathbb{E}_{(\mathbf{x}, y)} [\ell(\mathcal{A}_{\mathcal{F}}(S), \mathbf{x}, y)],$$

by using the test set error as a proxy for the true generalisation.

- We plot the histogram and the average of the distribution for various sizes of training set – initially the whole dataset gives a single value if we use training and test as the all the examples, but then we plot for training set sizes:

342, 273, 205, 137, 68, 34, 27, 20, 14, 7.

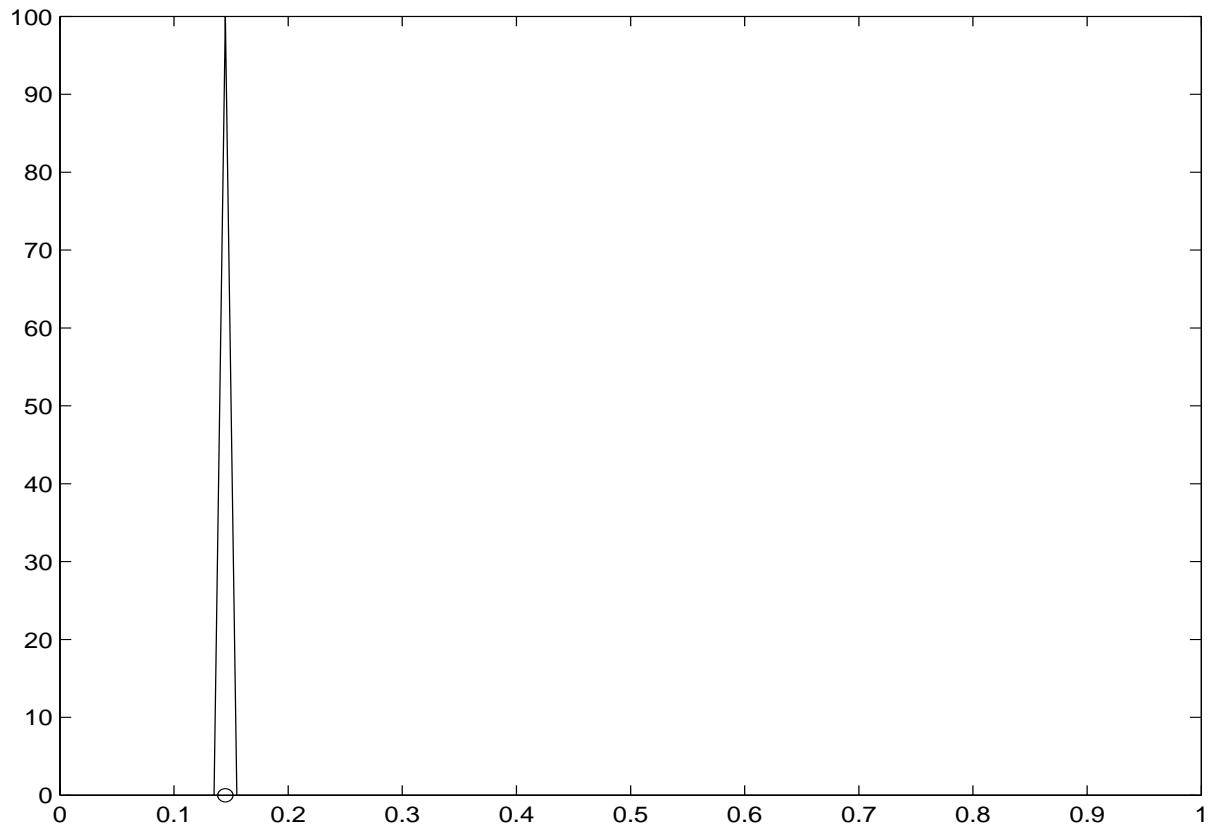
Example of Generalisation III

- Since the expected classifier is in all cases the same:

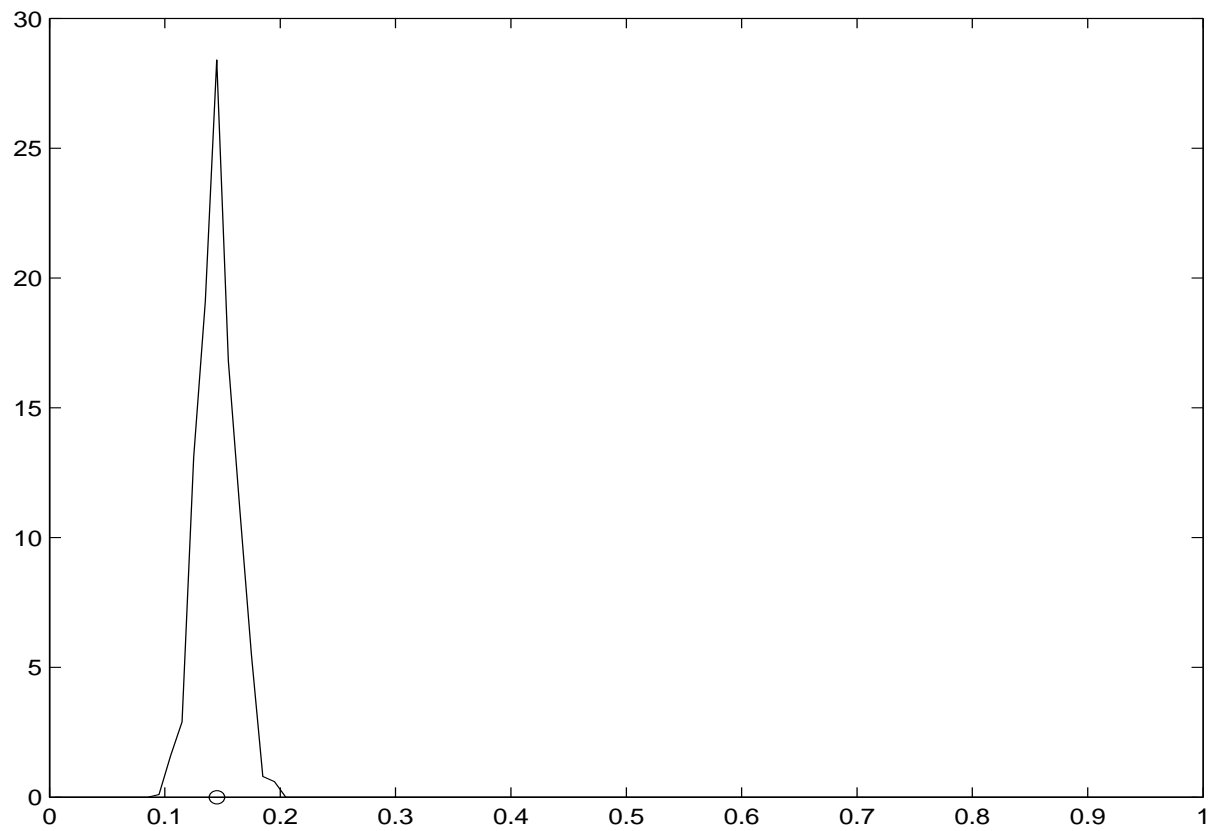
$$\begin{aligned}\mathbb{E}[\mathcal{A}_{\mathcal{F}}(S)] &= \mathbb{E}_S[\mathbf{w}_S^+ - \mathbf{w}_S^-] \\ &= \mathbb{E}_S[\mathbf{w}_S^+] - \mathbb{E}_S[\mathbf{w}_S^-] \\ &= \mathbb{E}_{y=+1}[\mathbf{x}] - \mathbb{E}_{y=-1}[\mathbf{x}],\end{aligned}$$

we do not expect large differences in the average of the distribution, though the non-linearity of the loss function means they won't be the same exactly.

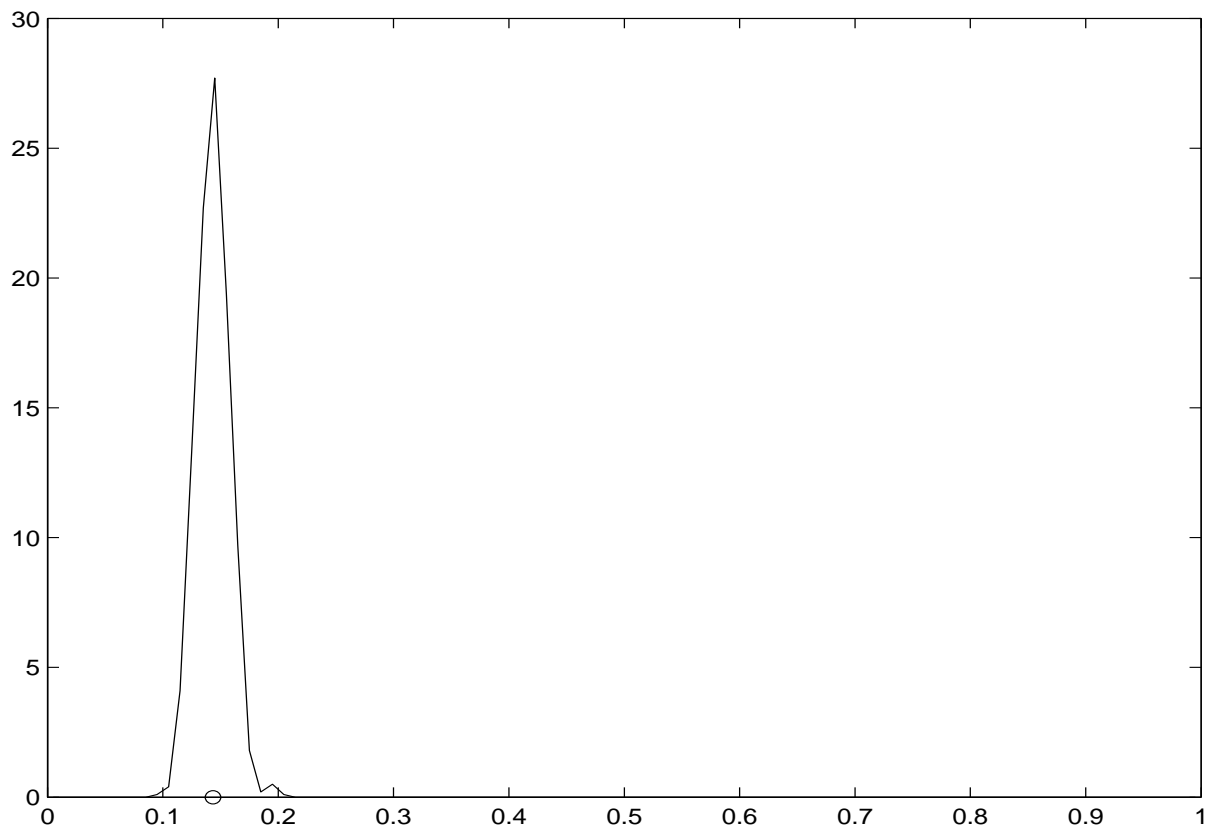
Error distribution: full dataset



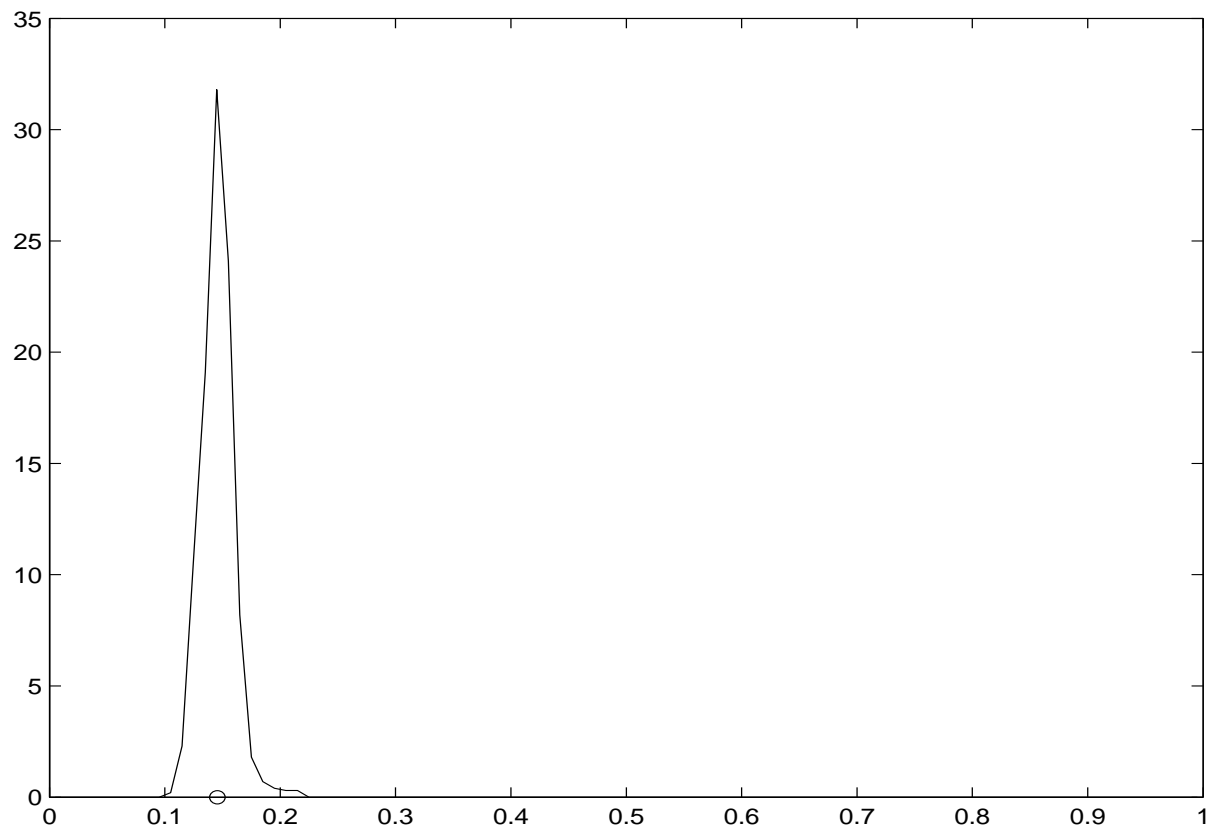
Error distribution: dataset size: 342



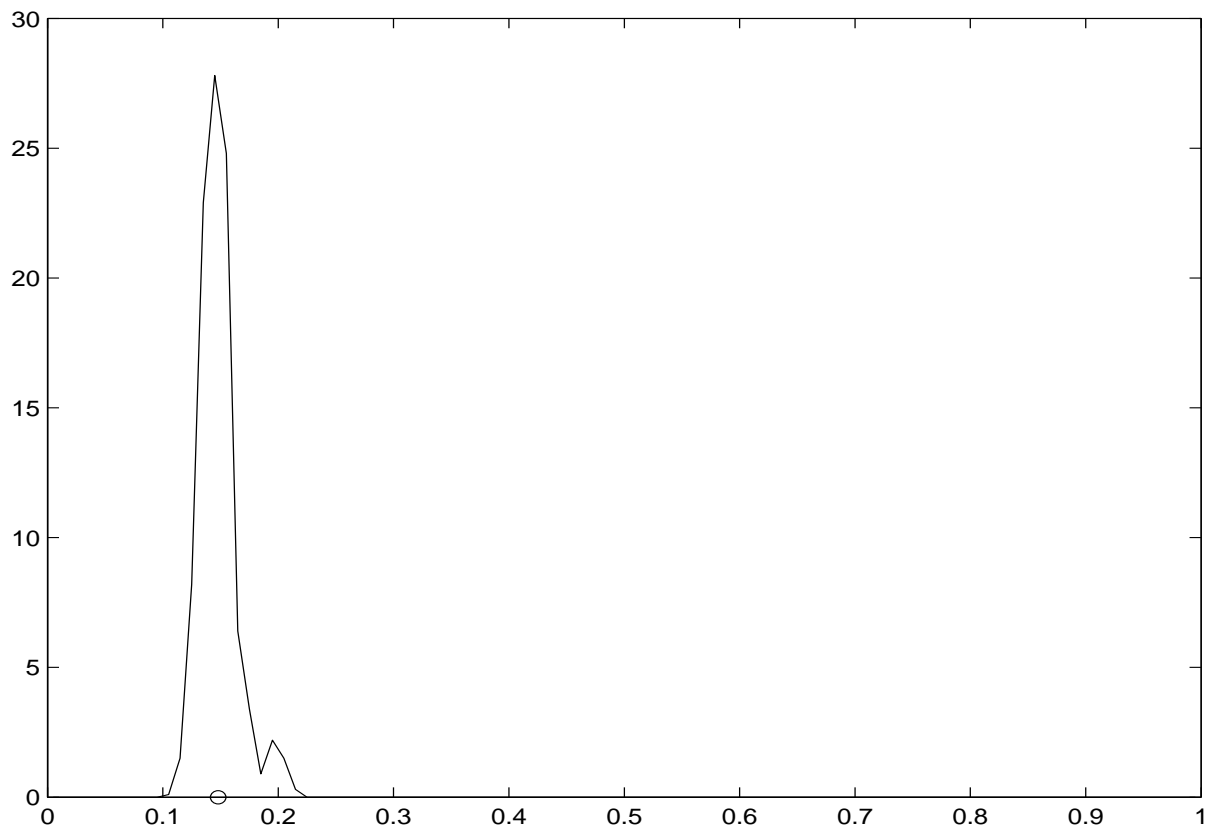
Error distribution: dataset size: 273



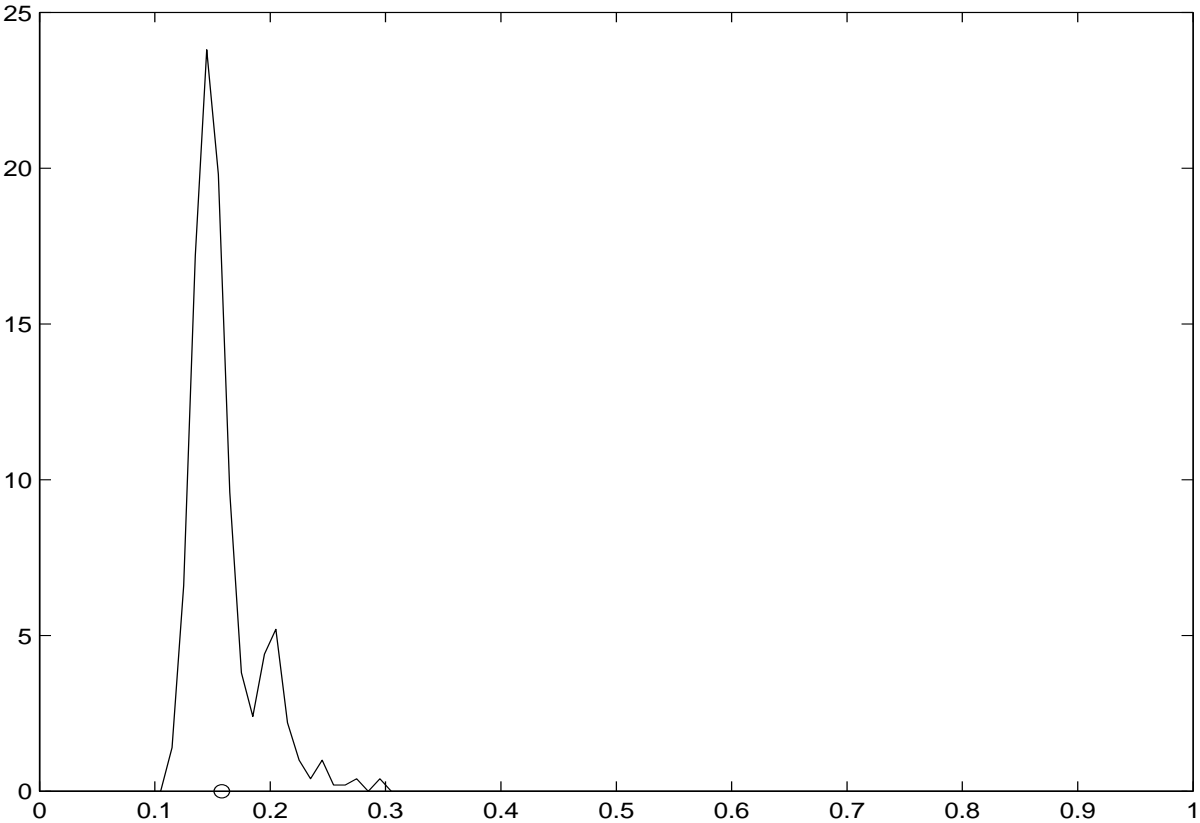
Error distribution: dataset size: 205



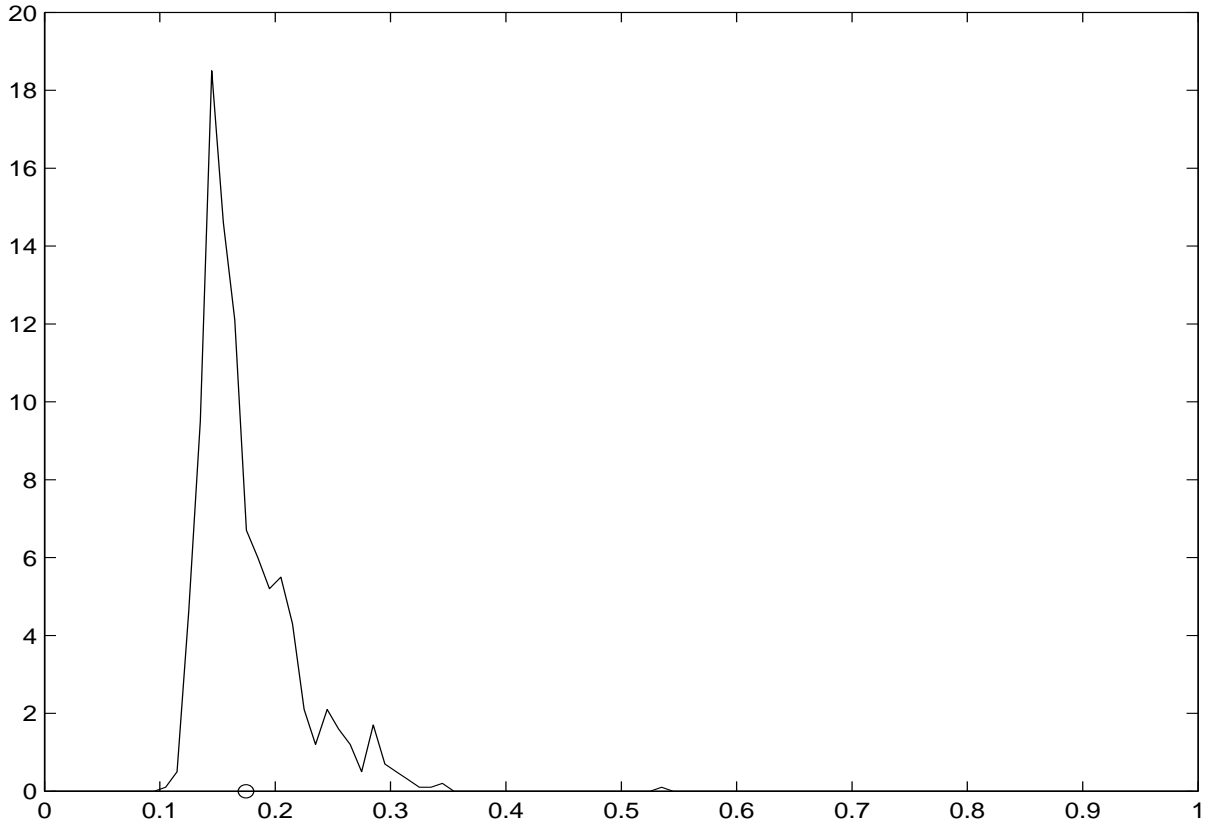
Error distribution: dataset size: 137



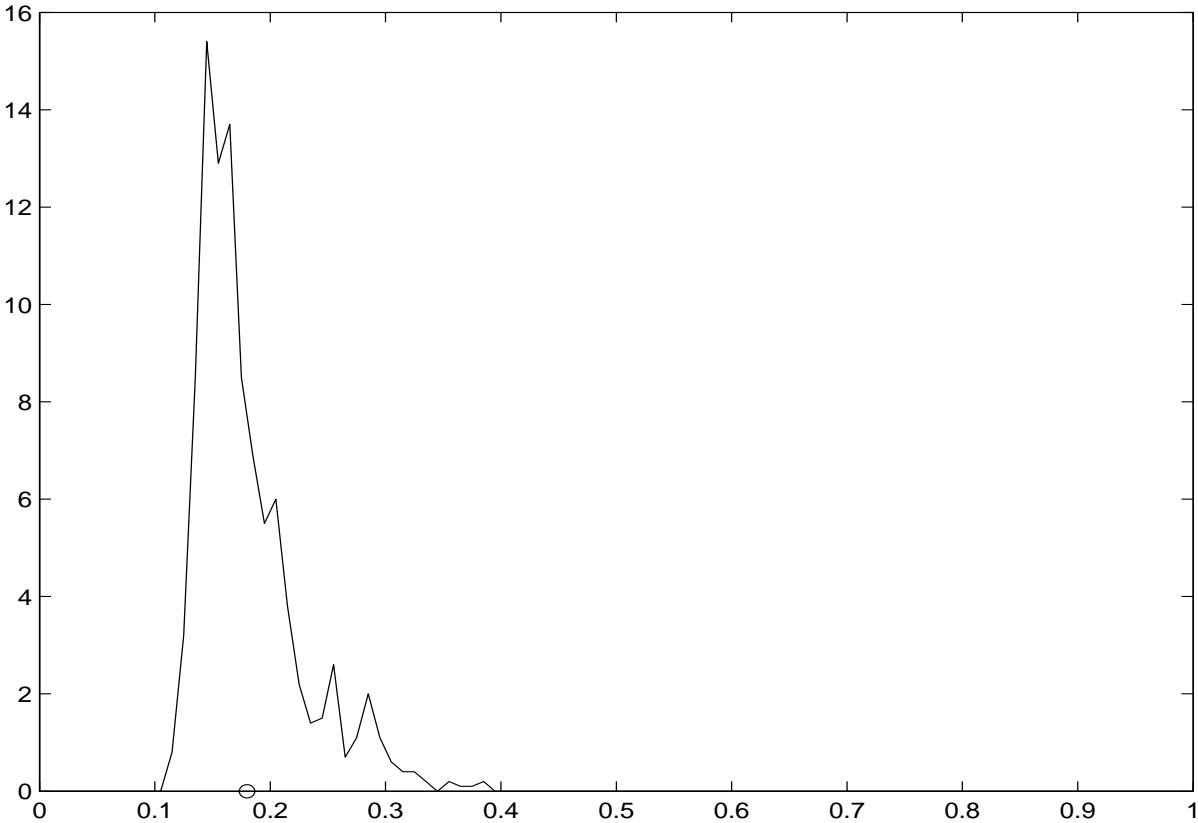
Error distribution: dataset size: 68



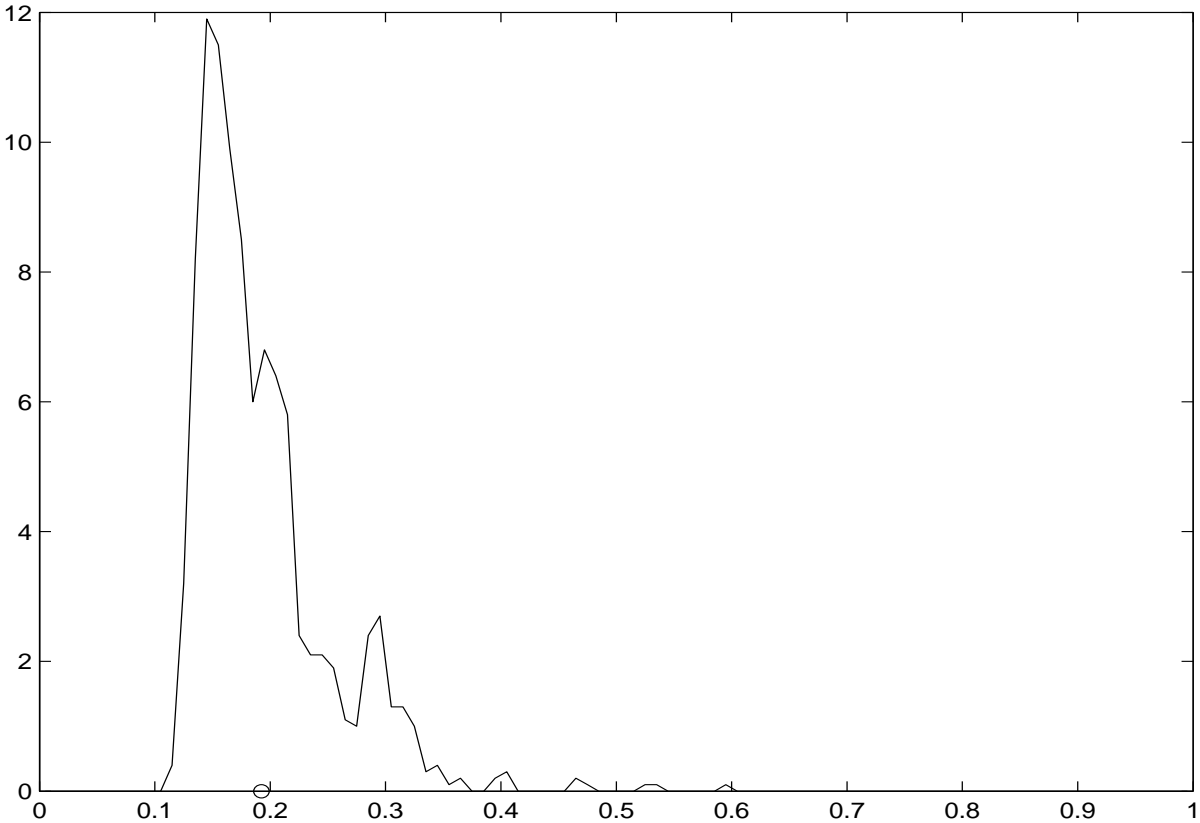
Error distribution: dataset size: 34



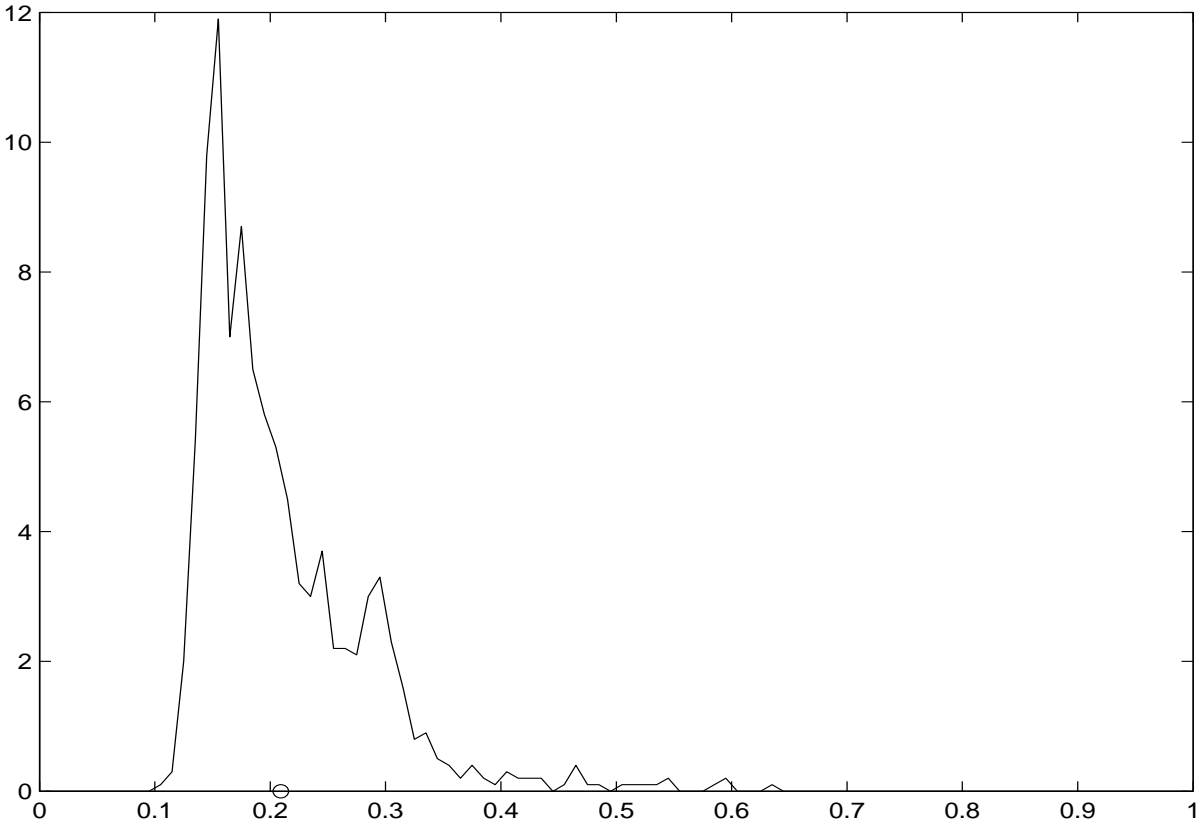
Error distribution: dataset size: 27



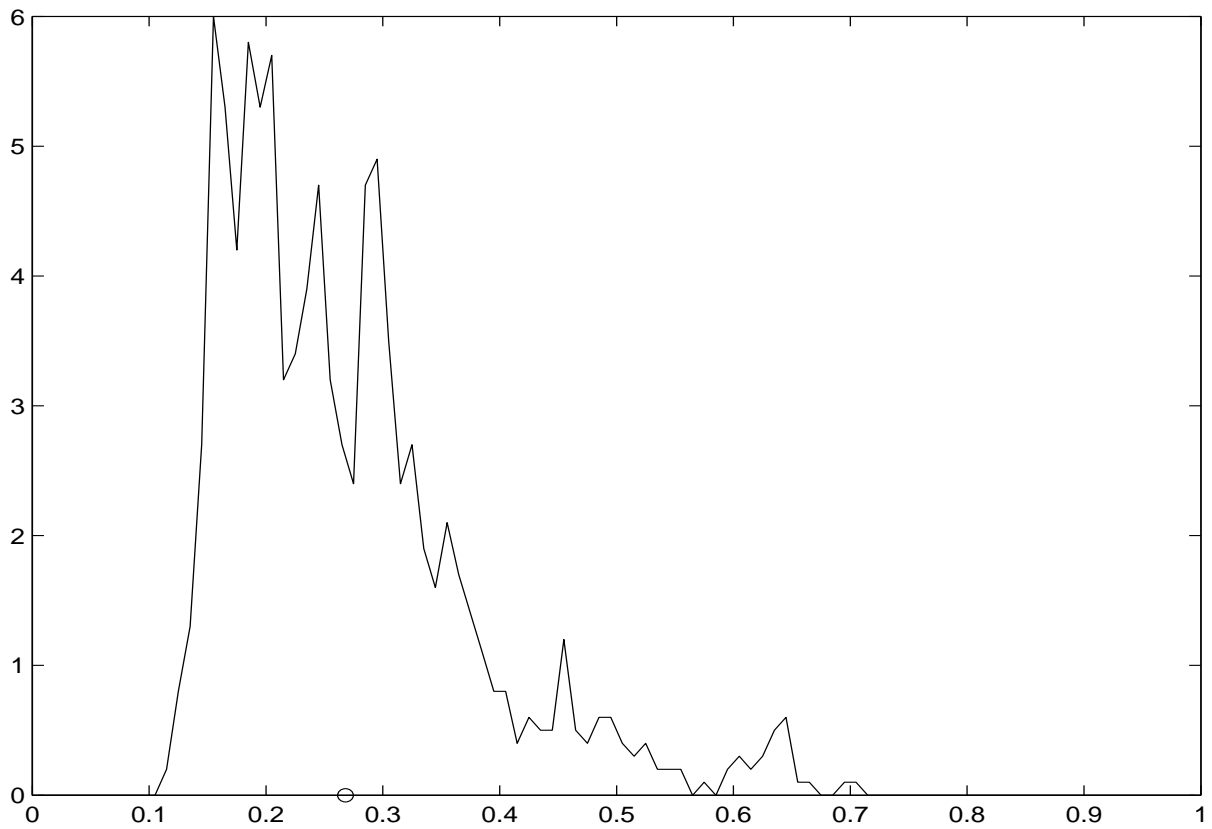
Error distribution: dataset size: 20



Error distribution: dataset size: 14



Error distribution: dataset size: 7



Observations

- Things can get bad if number of training examples small compared to dimension (in this case input dimension is 9)
- Mean can be bad predictor of true generalisation i.e. things can look okay in expectation, but still go badly wrong
- Key ingredient of learning keep flexibility high while still ensuring good generalisation

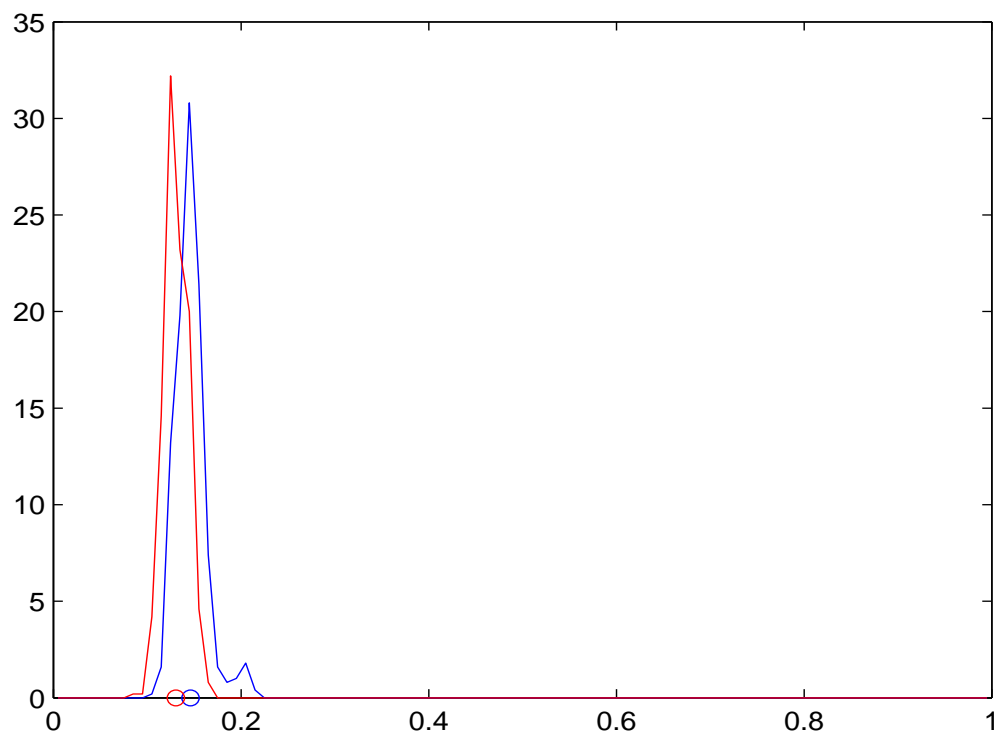
Controlling generalisation

- The critical method of controlling generalisation for classification is to force a large margin on the training data
- Equivalent to minimising the norm while keeping the separation fixed (at say ± 1)
- Support Vector Machines implement this strategy

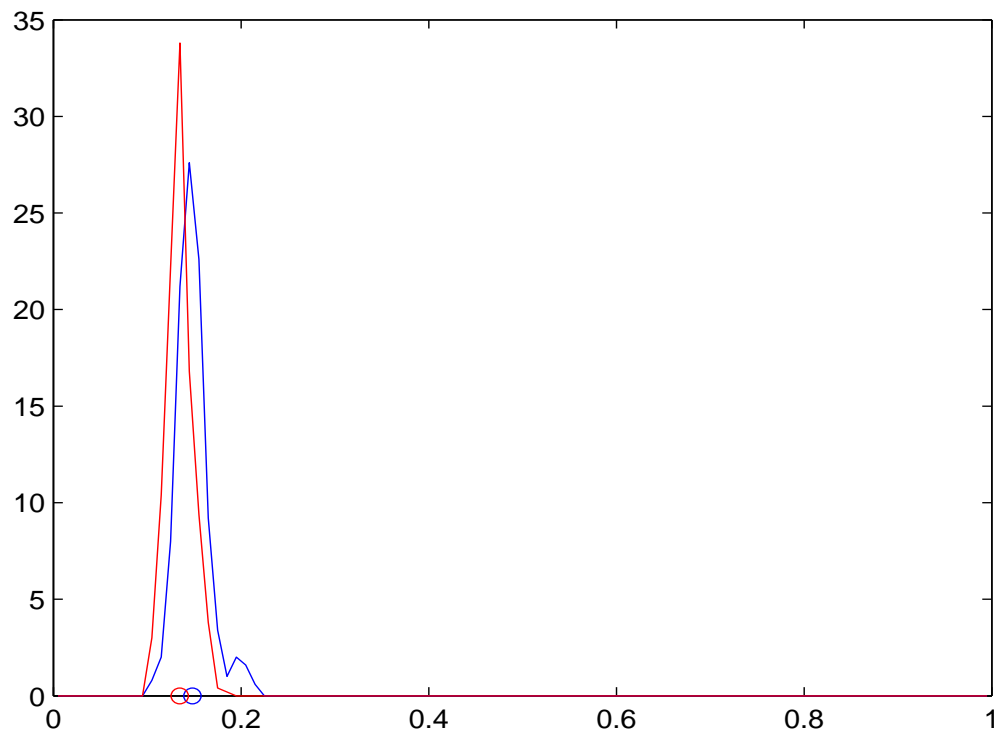
Controlling generalisation

- Now consider using an SVM on the same data and compare the distribution of generalisations
- SVM distribution in red

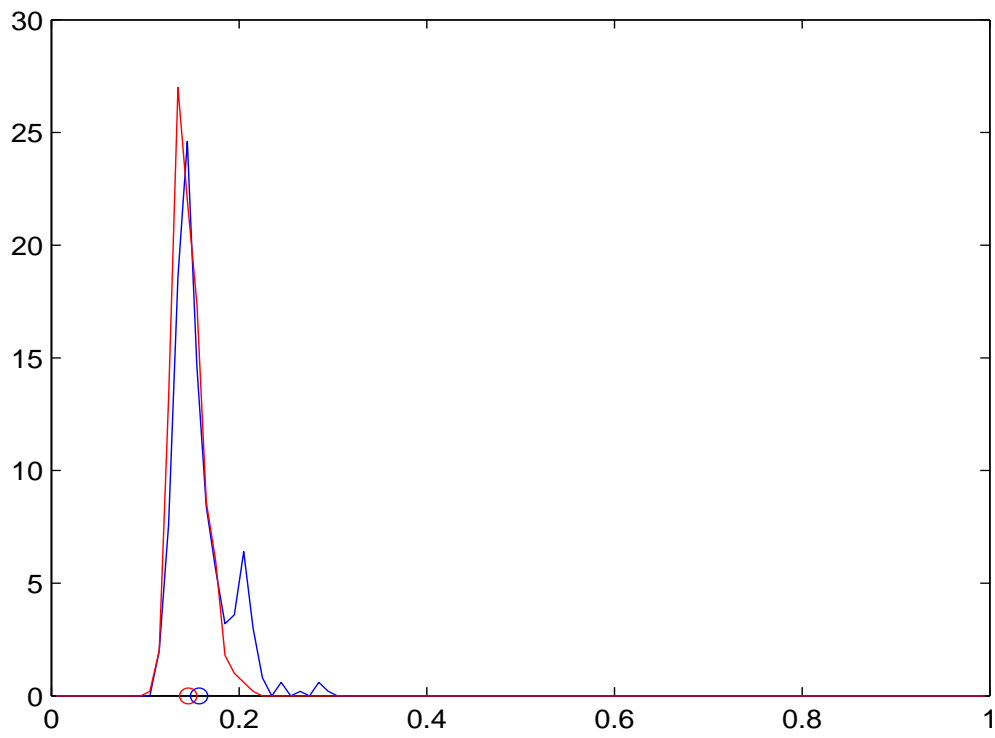
Error distribution: dataset size: 205



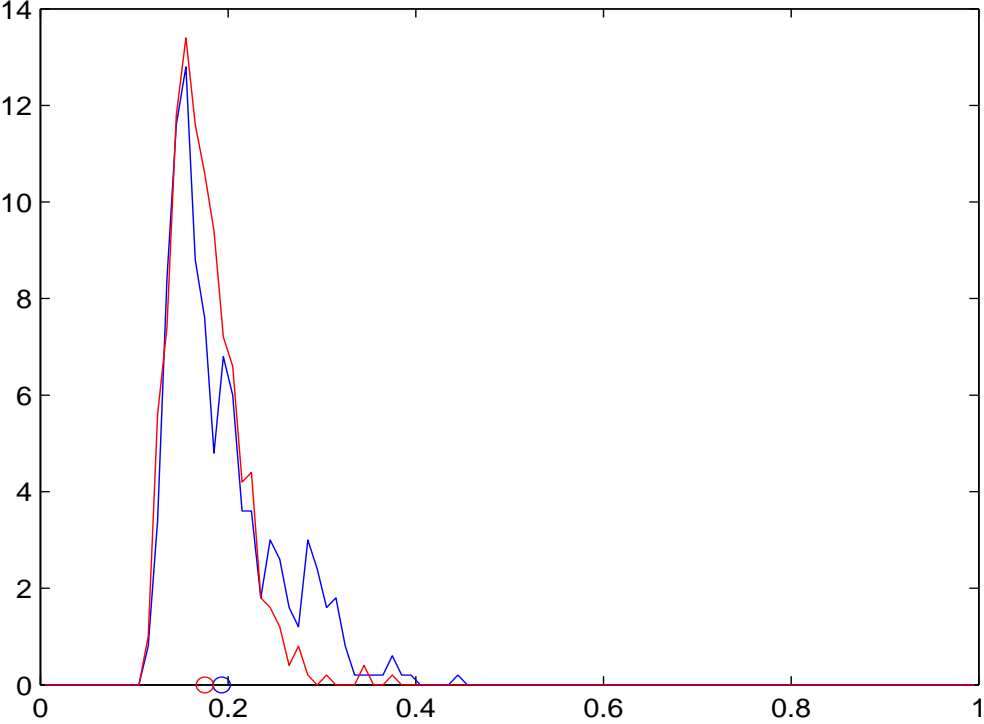
Error distribution: dataset size: 137



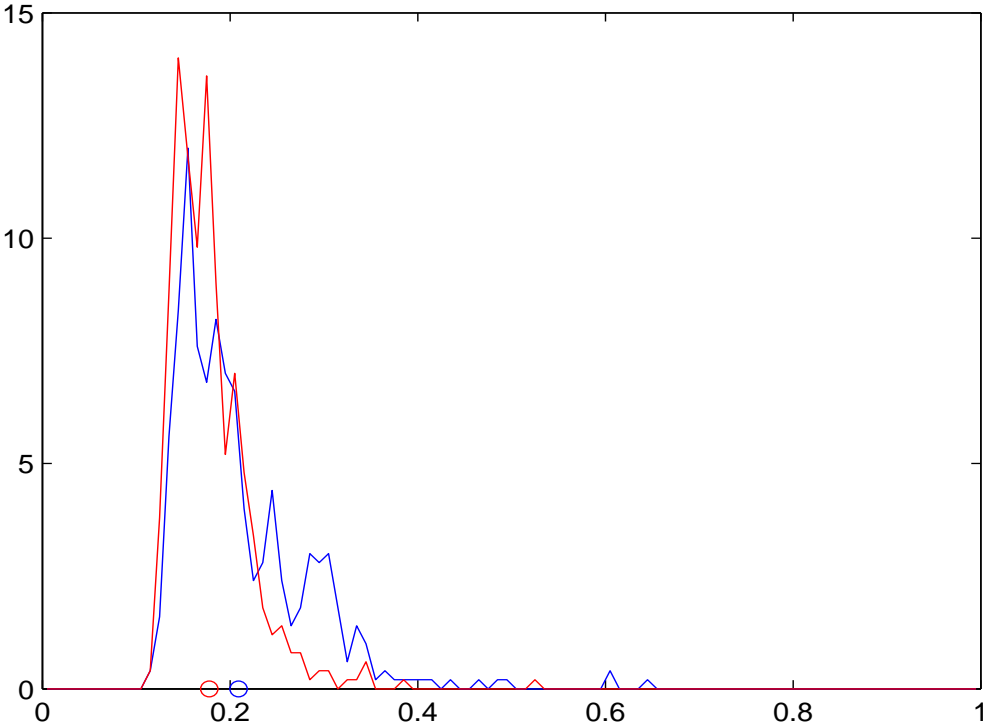
Error distribution: dataset size: 68



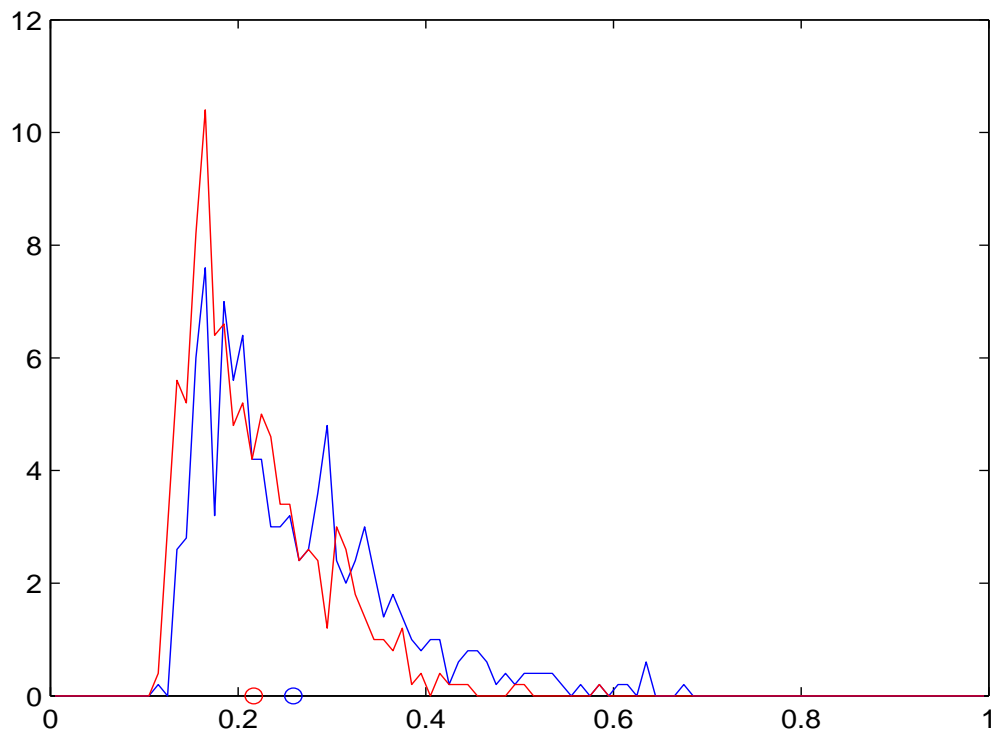
Error distribution: dataset size: 20



Error distribution: dataset size: 14



Error distribution: dataset size: 7



Expected versus confident bounds

- For a finite sample the generalisation $\epsilon(S, \mathcal{A}, \mathcal{F})$ has a distribution depending on the algorithm, function class and sample size m .
- Traditional statistics as indicated above has concentrated on the mean of this distribution – but this quantity can be misleading, eg for low fold cross-validation.

Expected versus confident bounds cont.

- Statistical learning theory has preferred to analyse the tail of the distribution, finding a bound which holds with high probability.
- This looks like a statistical test – significant at a 1% confidence means that the chances of the conclusion not being true are less than 1% over random samples of that size.
- This is also the source of the acronym PAC: probably approximately correct, the ‘confidence’ parameter δ is the probability that we have been misled by the training set.

Concentration inequalities

- Statistical Learning is concerned with the reliability or stability of inferences made from a random sample.
- Random variables with this property have been a subject of ongoing interest to probabilists and statisticians.

Concentration inequalities cont.

- As an example consider the mean of a sample of m 1-dimensional random variables X_1, \dots, X_m :

$$S_m = \frac{1}{m} \sum_{i=1}^m X_i.$$

- Hoeffding's inequality states that if $X_i \in [a_i, b_i]$

$$P\{|S_m - \mathbb{E}[S_m]| \geq \epsilon\} \leq 2 \exp\left(-\frac{2m^2\epsilon^2}{\sum_{i=1}^m (b_i - a_i)^2}\right)$$

Note how the probability falls off exponentially with the distance from the mean and with the number of variables.

Concentration for SLT

- We are now going to look at deriving SLT results from concentration inequalities.
- Perhaps the best known form is due to McDiarmid (although he was actually representing previously derived results):

McDiarmid's inequality

Theorem 1. Let X_1, \dots, X_n be independent random variables taking values in a set A , and assume that $f : A^n \rightarrow \mathbb{R}$ satisfies

$$\sup_{x_1, \dots, x_n, \hat{x}_i \in A} |f(x_1, \dots, x_n) - f(x_1, \dots, \hat{x}_i, x_{i+1}, \dots, x_n)| \leq c_i,$$

for $1 \leq i \leq n$. Then for all $\epsilon > 0$,

$$P\{f(X_1, \dots, X_n) - \mathbb{E}f(X_1, \dots, X_n) \geq \epsilon\} \leq \exp\left(\frac{-2\epsilon^2}{\sum_{i=1}^n c_i^2}\right)$$

- Hoeffding is a special case when $f(x_1, \dots, x_n) = S_n$

Using McDiarmid

- By setting the right hand side equal to δ , we can always invert McDiarmid to get a high confidence bound: with probability at least $1 - \delta$

$$f(X_1, \dots, X_n) < \mathbb{E}f(X_1, \dots, X_n) + \sqrt{\frac{\sum_{i=1}^n c_i^2}{2} \log \frac{1}{\delta}}$$

- If $c_i = c/n$ for each i this reduces to

$$f(X_1, \dots, X_n) < \mathbb{E}f(X_1, \dots, X_n) + \sqrt{\frac{c^2}{2n} \log \frac{1}{\delta}}$$

Rademacher proof beginnings

For a fixed $f \in \mathcal{F}$ we have

$$\mathbb{E}[f(\mathbf{z})] \leq \hat{\mathbb{E}}[f(\mathbf{z})] + \sup_{h \in \mathcal{F}} \left(\mathbb{E}[h] - \hat{\mathbb{E}}[h] \right).$$

where \mathcal{F} is a class of functions mapping from Z to $[0, 1]$ and $\hat{\mathbb{E}}$ denotes the sample average.

We must bound the size of the second term. First apply McDiarmid's inequality to obtain ($c_i = 1/m$ for all i) with probability at least $1 - \delta$:

$$\sup_{h \in \mathcal{F}} \left(\mathbb{E}[h] - \hat{\mathbb{E}}[h] \right) \leq \mathbb{E}_S \left[\sup_{h \in \mathcal{F}} \left(\mathbb{E}[h] - \hat{\mathbb{E}}[h] \right) \right] + \sqrt{\frac{\ln(1/\delta)}{2m}}.$$

Deriving double sample result

- We can now move to the ghost sample by simply observing that $\mathbb{E}[h] = \mathbb{E}_{\tilde{S}} [\hat{\mathbb{E}}[h]]$:

$$\mathbb{E}_S \left[\sup_{h \in \mathcal{F}} \left(\mathbb{E}[h] - \hat{\mathbb{E}}[h] \right) \right] =$$
$$\mathbb{E}_S \left[\sup_{h \in \mathcal{F}} \mathbb{E}_{\tilde{S}} \left[\frac{1}{m} \sum_{i=1}^m h(\tilde{\mathbf{z}}_i) - \frac{1}{m} \sum_{i=1}^m h(\mathbf{z}_i) \mid S \right] \right]$$

Deriving double sample result cont.

Since the sup of an expectation is less than or equal to the expectation of the sup (we can make the choice to optimise for each \tilde{S}) we have

$$\mathbb{E}_S \left[\sup_{h \in \mathcal{F}} \left(\mathbb{E}[h] - \hat{\mathbb{E}}[h] \right) \right] \leq \mathbb{E}_S \mathbb{E}_{\tilde{S}} \left[\sup_{h \in \mathcal{F}} \frac{1}{m} \sum_{i=1}^m (h(\tilde{\mathbf{z}}_i) - h(\mathbf{z}_i)) \right]$$

Adding symmetrisation

Here symmetrisation is again just swapping corresponding elements – but we can write this as multiplication by a variable σ_i which takes values ± 1 with equal probability:

$$\begin{aligned} \mathbb{E}_S \left[\sup_{h \in \mathcal{F}} \left(\mathbb{E}[h] - \hat{\mathbb{E}}[h] \right) \right] &\leq \\ &\leq \mathbb{E}_{\sigma S \tilde{S}} \left[\sup_{h \in \mathcal{F}} \frac{1}{m} \sum_{i=1}^m \sigma_i (h(\tilde{\mathbf{z}}_i) - h(\mathbf{z}_i)) \right] \\ &\leq 2 \mathbb{E}_{S \sigma} \left[\sup_{h \in \mathcal{F}} \frac{1}{m} \sum_{i=1}^m \sigma_i h(\mathbf{z}_i) \right] \\ &= R_m^*(\mathcal{F}), \end{aligned}$$

Rademacher complexity

where

$$R_m^*(\mathcal{F}) = \mathbb{E}_{S\sigma} \left[\sup_{f \in \mathcal{F}} \frac{2}{m} \sum_{i=1}^m \sigma_i f(\mathbf{z}_i) \right].$$

is known as the Rademacher (free) complexity of the function class \mathcal{F} .

Main Rademacher theorem

Putting the pieces together gives the main theorem of Rademacher complexity: with probability at least $1 - \delta$ over random samples S of size m , every $f \in \mathcal{F}$ satisfies

$$\mathbb{E} [f(\mathbf{z})] \leq \hat{\mathbb{E}} [f(\mathbf{z})] + R_m^*(\mathcal{F}) + \sqrt{\frac{\ln(1/\delta)}{2m}}$$

- Note that Rademacher complexity gives the expected value of the maximal correlation with random noise – a very natural measure of capacity.
- Note that the Rademacher complexity is distribution dependent since it involves an expectation over the choice of sample – this might seem hard to compute.

Empirical Rademacher theorem

- Since the empirical Rademacher complexity

$$\hat{R}_m^*(\mathcal{F}) = \mathbb{E}_\sigma \left[\sup_{f \in \mathcal{F}} \frac{2}{m} \sum_{i=1}^m \sigma_i f(\mathbf{z}_i) \mid \mathbf{z}_1, \dots, \mathbf{z}_m \right]$$

is concentrated, we can make a further application of McDiarmid to obtain with probability at least $1 - \delta$

$$\mathbb{E}_{\mathcal{D}} [f(\mathbf{z})] \leq \hat{\mathbb{E}} [f(\mathbf{z})] + \hat{R}_m^*(\mathcal{F}) + 3\sqrt{\frac{\ln(2/\delta)}{2m}}.$$

Application to large margin classification

- Rademacher complexity comes into its own for Boosting and SVMs.

Application to Boosting

- We can view Boosting as seeking a function from the class

$$\left\{ \sum_{h \in H} a_h h(\mathbf{x}) : \sum_{h \in H} a_h \leq B \right\} = \text{conv}_B(H)$$

by minimising some function of the margin distribution. For the 1-norm of the slack variables we arrive at Linear programming boosting that minimises

$$\sum_h a_h + C \sum_{i=1}^m \xi_i,$$

where $\xi_i = (1 - y_i \sum_h a_h h(\mathbf{x}_i))_+$.

Rademacher complexity of convex hulls

Rademacher complexity has a very nice property for convex hull classes:

$$\begin{aligned}\hat{R}_m^*(\text{conv}_B(H)) &= \frac{2}{m} \mathbb{E}_\sigma \left[\sup_{h_j \in H, \sum_j a_j \leq B} \sum_{i=1}^m \sigma_i \sum_j a_j h_j(\mathbf{x}_i) \right] \\ &\leq \frac{2}{m} \mathbb{E}_\sigma \left[\sup_{h_j \in H, \sum_j a_j \leq B} \sum_j a_j \sum_{i=1}^m \sigma_i h_j(\mathbf{x}_i) \right] \\ &\leq \frac{2}{m} \mathbb{E}_\sigma \left[\sup_{h_j \in H} B \sum_{i=1}^m \sigma_i h_j(\mathbf{x}_i) \right] \\ &\leq B \hat{R}_m^*(H).\end{aligned}$$

Rademacher complexity of convex hulls cont.

- Hence, we can move to the convex hull without incurring any complexity penalty for $B = 1$!

Rademacher complexity for SVMs

- The Rademacher complexity of a class of linear functions with bounded 2-norm:

$$\left\{ \mathbf{x} \rightarrow \sum_{i=1}^m \alpha_i \kappa(\mathbf{x}_i, \mathbf{x}) : \alpha' \mathbf{K} \alpha \leq B^2 \right\} \subseteq$$
$$\subseteq \{ \mathbf{x} \rightarrow \langle \mathbf{w}, \phi(\mathbf{x}) \rangle : \|\mathbf{w}\| \leq B \}$$
$$= \mathcal{F}_B,$$

where we assume a kernel defined feature space with

$$\langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle = \kappa(\mathbf{x}, \mathbf{z}).$$

Rademacher complexity of \mathcal{F}_B

The following derivation gives the result

$$\begin{aligned}\hat{R}_m^*(\mathcal{F}_B) &= \mathbb{E}_\sigma \left[\sup_{f \in \mathcal{F}_B} \frac{2}{m} \sum_{i=1}^m \sigma_i f(\mathbf{x}_i) \right] \\ &= \mathbb{E}_\sigma \left[\sup_{\|\mathbf{w}\| \leq B} \left\langle \mathbf{w}, \frac{2}{m} \sum_{i=1}^m \sigma_i \phi(\mathbf{x}_i) \right\rangle \right] \\ &\leq \frac{2B}{m} \mathbb{E}_\sigma \left[\left\| \sum_{i=1}^m \sigma_i \phi(\mathbf{x}_i) \right\| \right] \\ &= \frac{2B}{m} \mathbb{E}_\sigma \left[\left(\left\langle \sum_{i=1}^m \sigma_i \phi(\mathbf{x}_i), \sum_{j=1}^m \sigma_j \phi(\mathbf{x}_j) \right\rangle \right)^{1/2} \right] \\ &\leq \frac{2B}{m} \left(\mathbb{E}_\sigma \left[\sum_{i,j=1}^m \sigma_i \sigma_j \kappa(\mathbf{x}_i, \mathbf{x}_j) \right] \right)^{1/2} = \frac{2B}{m} \sqrt{\sum_{i=1}^m \kappa(\mathbf{x}_i, \mathbf{x}_i)}\end{aligned}$$

Applying to 1-norm SVMs

We take the following formulation of the 1-norm SVM:

$$\begin{aligned} \min_{\mathbf{w}, b, \gamma, \xi} \quad & -\gamma + C \sum_{i=1}^m \xi_i \\ \text{subject to} \quad & y_i (\langle \mathbf{w}, \phi(\mathbf{x}_i) \rangle + b) \geq \gamma - \xi_i, \xi_i \geq 0, \\ & i = 1, \dots, m, \text{ and } \|\mathbf{w}\|^2 = 1. \end{aligned} \tag{1}$$

Note that

$$\xi_i = (\gamma - y_i g(\mathbf{x}_i))_+,$$

where $g(\cdot) = \langle \mathbf{w}, \phi(\cdot) \rangle + b$.

- The first step is to introduce a loss function which upper bounds the discrete loss

$$P(y \neq \text{sgn}(g(\mathbf{x}))) = \mathbb{E} [\mathcal{H}(-yg(\mathbf{x}))],$$

where \mathcal{H} is the Heaviside function.

Applying the Rademacher theorem

- Consider the loss function $\mathcal{A} : \mathbb{R} \rightarrow [0, 1]$, given by

$$\mathcal{A}(a) = \begin{cases} 1, & \text{if } a > 0; \\ 1 + a/\gamma, & \text{if } -\gamma \leq a \leq 0; \\ 0, & \text{otherwise.} \end{cases}$$

- By the Rademacher Theorem and since the loss function $\mathcal{A} - 1$ dominates $\mathcal{H} - 1$, we have that

$$\begin{aligned} \mathbb{E} [\mathcal{H}(-yg(\mathbf{x})) - 1] &\leq \mathbb{E} [\mathcal{A}(-yg(\mathbf{x})) - 1] \\ &\leq \hat{\mathbb{E}} [\mathcal{A}(-yg(\mathbf{x})) - 1] + \\ &\quad \hat{R}_m((\mathcal{A} - 1) \circ \mathcal{F}) + 3\sqrt{\frac{\ln(2/\delta)}{2m}}. \end{aligned}$$

Empirical loss and slack variables

- But the function $\mathcal{A}(-y_i g(\mathbf{x}_i)) \leq \xi_i/\gamma$, for $i = 1, \dots, m$, and so

$$\mathbb{E} [\mathcal{H}(-y g(\mathbf{x}))] \leq \frac{1}{m\gamma} \sum_{i=1}^m \xi_i + \hat{R}_m^*((\mathcal{A} - 1) \circ \mathcal{F}) + 3\sqrt{\frac{\ln(2/\delta)}{2m}}.$$

- The final missing ingredient to complete the bound is to bound $\hat{R}_m^*((\mathcal{A} - 1) \circ \mathcal{F})$ in terms of $\hat{R}_m^*(\mathcal{F})$.
- This can be bounded in terms of the Lipschitz constant γ^{-1} of the function $\mathcal{A} - 1$

$$\hat{R}_m^*((\mathcal{A} - 1) \circ \mathcal{F}) \leq \gamma^{-1} \hat{R}_m^*(\mathcal{F})$$

Final SVM bound

- Assembling the result we obtain:

$$\begin{aligned} P(y \neq \text{sgn}(g(\mathbf{x}))) &= \mathbb{E}[\mathcal{H}(-yg(\mathbf{x}))] \\ &\leq \frac{1}{m\gamma} \sum_{i=1}^m \xi_i + \frac{2}{m\gamma} \sqrt{\sum_{i=1}^m \kappa(\mathbf{x}_i, \mathbf{x}_i)} + 3\sqrt{\frac{\ln(2/\delta)}{2m}} \end{aligned}$$

- Note that for the Gaussian kernel this reduces to

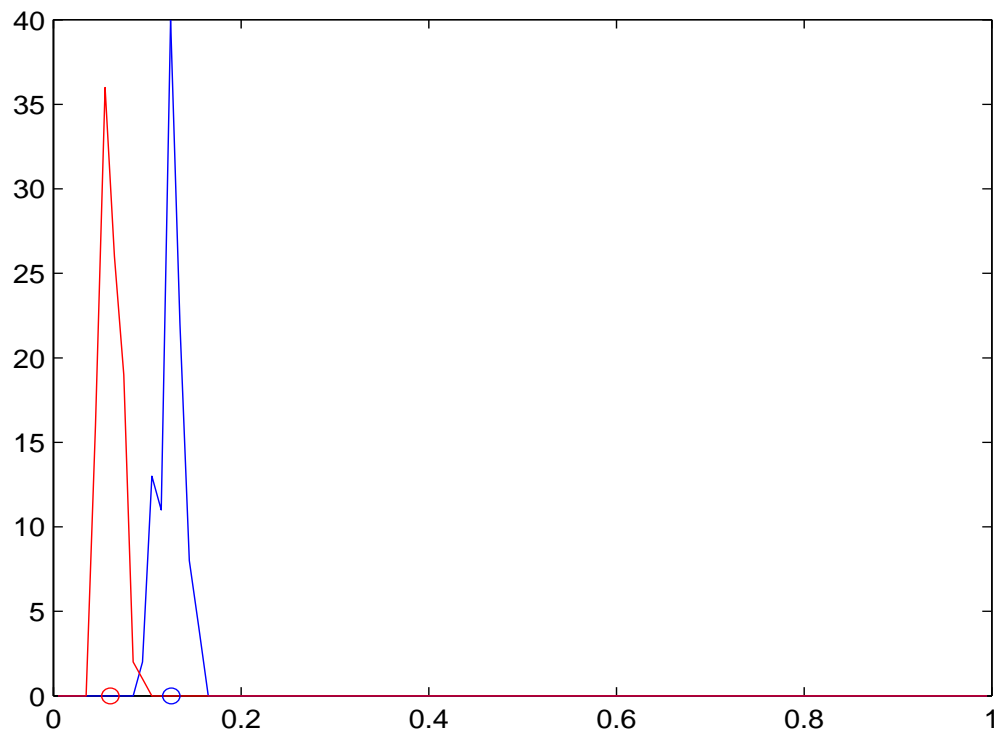
$$P(y \neq \text{sgn}(g(\mathbf{x}))) \leq \frac{1}{m\gamma} \sum_{i=1}^m \xi_i + \frac{2}{\sqrt{m}\gamma} + 3\sqrt{\frac{\ln(2/\delta)}{2m}}$$

Using a kernel

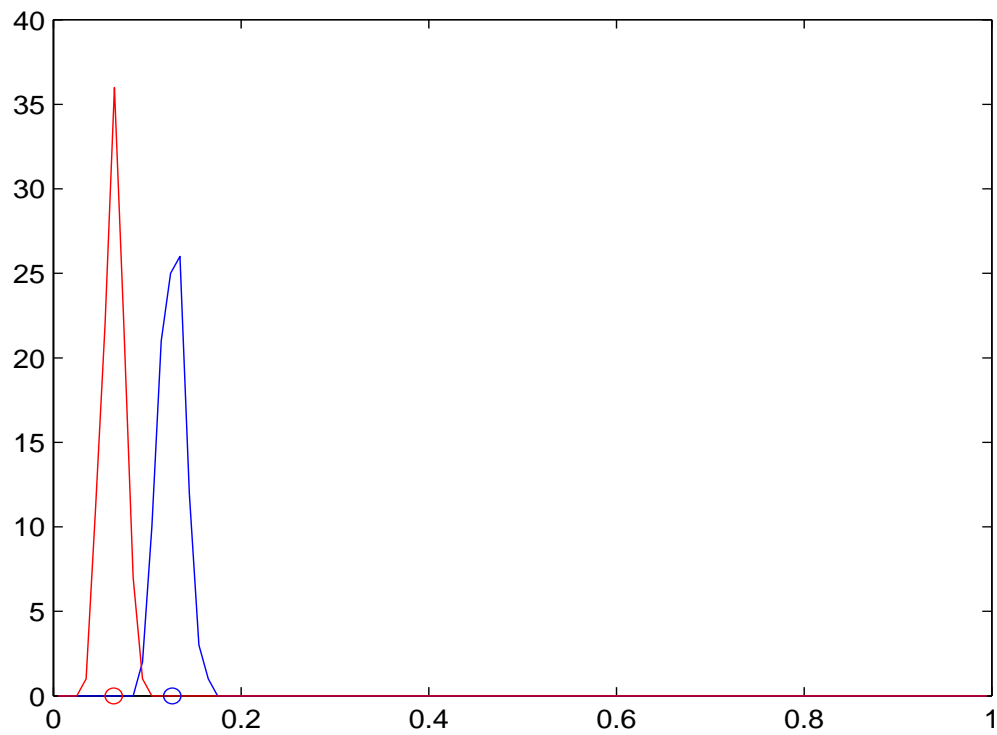
- Can consider much higher dimensional spaces using the kernel trick
- Can even work in infinite dimensional spaces, eg using the Gaussian kernel:

$$\kappa(\mathbf{x}, \mathbf{z}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{z}\|^2}{2\sigma^2}\right)$$

Error distribution: dataset size: 342



Error distribution: dataset size: 273



Multi-view learning

- In many applications we have two views of the same phenomenon:
 - A document and its translation
 - An image and its caption
 - Two feature representations of an image: e.g. interest points and patches
- We hypothesise that both views contain common information for classification, but each has different irrelevant information
- Train two SVMs but constrain the real valued outputs of the underlying linear functions to be close

SVM-2K

- Two kernels and associated constraints:

$$\begin{aligned} \min L &= \frac{1}{2} \|w_A\|^2 + \frac{1}{2} \|w_B\|^2 + C^A \sum \xi_i^A + C^B \sum \xi_i^B + D \sum \eta_i \\ \text{s.t. } & |\langle w_A, \phi_A(x_i) \rangle - \langle w_B, \phi_B(x_i) \rangle| \leq \eta_i + \epsilon \\ & y_i \langle w_A, \phi_A(x_i) \rangle \geq 1 - \xi_i^A \\ & y_i \langle w_B, \phi_B(x_i) \rangle \geq 1 - \xi_i^B \\ & \xi_i^A \geq 0 \quad \xi_i^B \geq 0 \quad \eta_i \geq 0 \end{aligned}$$

- Let \hat{w}_A, \hat{w}_B be the solution to this optimisation problem. The final decision function is then

$$\begin{aligned} f(x) &= 0.5 (\langle \hat{w}_A, \phi_A(x) \rangle + \langle \hat{w}_B, \phi_B(x) \rangle) \\ &= 0.5 (f_A(x) + f_B(x)). \end{aligned}$$

Rademacher Analysis

First observe that an application of the Rademacher bound shows that

$$\begin{aligned} \mathbb{E}_x[|f_A(x) - f_B(x)|] &\leq \mathbb{E}_x[|\langle \hat{w}_A, \phi_A(x) \rangle - \langle \hat{w}_B, \phi_B(x) \rangle|] \\ &\leq \epsilon + \frac{1}{m} \sum_{i=1}^m \eta_i + \frac{2C}{m} \sqrt{\text{tr}(K_A) + \text{tr}(K_B)} + 3\sqrt{\frac{\ln(2/\delta)}{m}} =: D \end{aligned}$$

with probability at least $1 - \delta$. Hence, the class of functions used is

$$\mathcal{F}_C = \left\{ f \mid f : x \rightarrow 0.5 \left(\sum_{i=1}^m [\alpha_A^i \kappa_A(x_i, x) + \alpha_B^i \kappa_B(x_i, x)] \right), \right. \\ \left. \alpha_A' K_A \alpha_A \leq C^2, \alpha_B' K_B \alpha_B \leq C^2, \mathbb{E}_x[|f_A(x) - f_B(x)|] \leq D \right\}$$

Rademacher bounds for SVM-2K

- Consider the function of two weight vectors w_A and w_B ,

$$D(w_A, w_B) := \mathbb{E}_x [|\langle w_A, \phi_A(x) \rangle + b_A - \langle w_B, \phi_B(x) \rangle - b_B|]$$

- Our Rademacher complexity is therefore

$$\begin{aligned} \hat{R}_{*m}(\mathcal{F}_C) &= \mathbb{E}_\sigma \left[\sup_{f \in \mathcal{F}_C} \frac{2}{m} \sum_{i=1}^m \sigma_i f(x_i) \right] \\ &= \mathbb{E}_\sigma \left[\sup_{\substack{\|w_A\| \leq C \\ \|w_B\| \leq C \\ D(w_A, w_B) \leq D}} \frac{1}{m} \sum_{i=1}^m \sigma_i [\langle w_A, \phi_A(x_i) \rangle + \langle w_B, \phi_B(x_i) \rangle] \right] \end{aligned}$$

Rademacher bounds for SVM-2K

A reverse Rademacher theorem shows that for weight vectors w_A and w_B satisfying $D(w_A, w_B) \leq D$, with probability at least $1 - \delta$ we have

$$\begin{aligned}\hat{D}(w_A, w_B) &:= \mathbb{E}_S[|\langle w_A, \phi_A(x) \rangle - \langle w_B, \phi_B(x) \rangle|] \\ &\leq D + \frac{2C}{m} \sqrt{\text{tr}(K_A) + \text{tr}(K_B)} + 3\sqrt{\frac{\ln(2/\delta)}{m}} \\ &\leq \epsilon + \frac{1}{m} \sum_{i=1}^m \eta_i + \frac{4C}{m} \sqrt{\text{tr}(K_A) + \text{tr}(K_B)} \\ &\quad + 6\sqrt{\frac{\ln(2/\delta)}{m}} =: \hat{D}\end{aligned}$$

Evaluating the bound

- By an application of McDiarmid can fix one random evaluation σ , so Rademacher complexity bounded by

$$\hat{R}_m^*(\mathcal{F}_C) \leq \sup_{\substack{\|w_A\| \leq C \\ \|w_B\| \leq C \\ \hat{D}(w_A, w_B) \leq \hat{D}}} \frac{1}{m} \sum_{i=1}^m \sigma_i [\langle w_A, \phi_A(x_i) \rangle + \langle w_B, \phi_B(x_i) \rangle] \\ + RC \sqrt{\frac{2}{m} \log \frac{1}{\delta}}$$

Evaluating this bound involves solving an optimisation problem

- In practice significant reductions in complexity are achieved with corresponding improvement in classification accuracy.

SVM-2k: results

- Results obtained classifying patents from a Japanese patent dataset with paired English translations.
- Results are average precision as percent – i.e. higher is better.
- Note that SVM-2k_j is performing crosslingual classification, i.e. only uses Japanese text – and often does better than a SVM in original language

	pSVM	kcca_SVM	SVM	SVM-2k _j	Concat	SVM-2k
1	59.4±3.9	60.3±2.8	66.6±2.8	66.1± 2.6	67.5±2.3	67.5±2.1
2	71.1±4.5	68.4±4.4	73.0±4.0	74.8±4.7	73.9±4.0	75.1±4.1
3	16.7±1.2	13.1±1.0	18.8±1.6	20.8±1.9	21.5±1.9	22.5±1.7
7	74.9±1.8	76.0±1.2	76.7±1.3	77.5±1.4	79.0±1.2	80.7±1.5
12	75.0±0.8	73.6±0.8	76.8±1.0	77.6±0.7	76.8±0.6	78.4±0.6
14	76.0±1.6	71.5±1.5	80.9±1.3	82.2±1.3	81.4±1.4	82.7±1.3

SVM-2k: results

- Results with image classification – object detection
- Two views are interest points and image patches



	Motorbike	Bicycle	People	Car
SVM 1	94.05	91.58	91.58	87.95
SVM 2	91.15	91.15	90.57	86.21
SVM 2K	94.34	93.47	92.74	90.13

SVM-2k: semi-supervised

- In the definition of the restriction placed on the two functions f_A and f_B we didn't need to use the same data as for training.
- note that we don't need labels to evaluate $\hat{D}(w_A, w_B)$ so can use unlabelled data for this
- Gives semi-supervised learning algorithm with even larger reductions in Rademacher complexity.

Semi-supervised framework

- General example of semi-supervised framework proposed by Blum and Balcan.
- Learn but restrict functions f to have some low average over test data when measured by a compatibility function χ .
- In our case functions are averaged pairs $0.5(f_A + f_B)$ with the compatibility measured by $|f_A - f_B|$.
- Example of more general ‘luckiness’, in which we posit properties such as large margin that if observed imply a much lower complexity of function space.

Conclusions

- Outline of philosophy and approach of SLT
- Particular case of SVMs
- Presentation of Rademacher complexity.
- Case of RC for classification
- SVM-2K
- Semi-supervised learning