



Deliverable D3.2

Metric Learning Analysis

Contract number: **FP7-216529** PinView

Personal Information Navigator Adapting Through Viewing

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under *grant agreement* n° 216529.



Identification sheet

Project ref. no.	FP7-216529
Project acronym	PinView
Status and version	Final, Revision 1.01
Contractual date of delivery	31.12.2009
Actual date of delivery	03.01.2010
Deliverable number	D3.2
Deliverable title	Metric Learning Analysis
Nature	report
Dissemination level	PU – Public
WP contributing to the deliverable	WP3 Learning the comparison metrics
Task contributing to the deliverable	Task 3.2 Criteria for metric selection
WP responsible	University College London
Task responsible	University College London
Editor	Zakria Hussain, <z.hussain@cs.ucl.ac.uk >
Editor address	Department of Computer Science, University College London, WC1E 6BT, United Kingdom
Authors in alphabetical order	Zakria Hussain and John Shawe-Taylor
EC Project Officer	Pierre-Paul Sondag
Keywords	multiple kernel learning, generalisation error bounds, linear programme boosting
Abstract	This report analyses Multiple Kernel Learning (MKL) algorithms in order to upper bound the loss of the algorithm proposed in Deliverable 3.1. We adapt a previously published bound, which shows that we will have good generalisation provided we have a large margin and a small number of kernels in the final combination. Furthermore, we propose a second generalisation error bound which motivates a new MKL algorithm using linear programme boosting (LP-Boost).

List of annexes

none

Contents

1 Overview	4
2 Introduction	5
3 Theoretical Analysis	5
3.1 Sample Compression-Margin bound	7
3.2 Rademacher Complexity bound	8
4 Algorithmic Extensions	10
4.1 Linear Programme Boosting as an MKL problem	10
5 Conclusions	12

1 Overview

This is the second Deliverable of Work Package 3 of the *Personal Information Navigator Adapting Through Viewing*, PinView, project, funded by the European Community’s Seventh Framework Programme under Grant Agreement n° 216529. The report constitutes the output of Task 3.2 *Criteria for metric selection*.

The description of work for *Task 3.2 Criteria for metric selection* is to:

“place the results of Task 3.1 in a rigorous statistical framework that can be used to analyse the factors that affect the performance of systems that learn metrics. This will be used to propose improvements and extensions to the classes of metrics and algorithms developed in Task 3.1. The new approaches will be implemented and evaluated to verify that the statistical analysis accurately captures the key features of the learning scenario.”

Therefore, the goal is to theoretically analyse the algorithm developed in D3.1 – namely the 1-norm 2-norm Multiple Kernel Learning algorithm that was developed for the one-class support vector machine. In the first section we adapt a generalisation error bound for the case when we would expect a small number of kernels in the final kernel combination. After this we develop a new analysis using Rademacher complexity – which also motivates an extension to MKL algorithms, based on applying Boosting to the MKL problem.

The involvement of TKK and MUL is ongoing. With respect to MUL, the work carried out during the visit of Alex Leung and Zakria Hussain to TKK is considered more relevant to deliverable D3.3, and so will be reported there.

2 Introduction

In this report our main goal is to analyse Multiple Kernel Learning (MKL) in terms of generalisation error. Constructing generalisation error bounds for MKL has been a topic of considerable interest of late, with several authors proposing bounds (Lanckriet et al., 2004; Srebro and Ben-David, 2006; Ying and Campbell, 2009). The bounds have used Rademacher complexity or margin theory. We start by applying a recent bounding technique (Hussain and Shawe-Taylor, 2009), used for kernel matching pursuit, in order to view the choice of kernels as a sample compression scheme (Littlestone and Warmuth, 1986). We show that in situations where we only expect to use a small number of kernels, that our bound can be tighter than the bound of Srebro and Ben-David (2006).

Next we propose a novel Rademacher bound (Bartlett and Mendelson, 2002; Shawe-Taylor and Cristianini, 2004) by making use of the idea of Rademacher complexities for Boosting. The fact that boosting constructs classifiers based on a linear combination of weak learners, suggests that by viewing each choice of kernel as a weak learner, we can adapt the LPBoost algorithm (Demiriz et al., 2002) to find a linear combination of kernels. We apply the LPBoost algorithm as opposed to AdaBoost (Freund and Schapire, 1997) due to its global convergence properties, well-defined stopping criteria and its sparsity (1-norm regularisation) in the choice of weak learners.

The motivation of our work is to show that MKL algorithms can have upper bounds for their generalisation error, putting much of the work from Deliverable 3.1 (Hussain et al., December 2008) on a sound theoretical footing. Furthermore, based on the analysis, another motivation is to propose a novel algorithm that also minimises the generalisation error bound. This is an important requirement of any learning algorithm, because in practice we would like to minimise the true error that may be incurred in the future. By minimising an *upper bound* on the true error we may hope to achieve good generalisation. This has been the goal of computational learning theory (Anthony and Bartlett, 1999) and remains an important component for the design of any new learning algorithm.

3 Theoretical Analysis

In this section we describe the MKL problem, and what we would like to analyse theoretically. Subsection 3.1 describes a method of applying sample compression bounds over margin bounds that have already been proposed for MKL. This bound is for the MKL algorithm proposed in Deliverable 3.1 (Hussain et al., December 2008). It suggests that if we find a small number of kernels that maximise the margin than we should learn well in the future. Subsection 3.2 describes a novel Rademacher bound for MKL. All proofs of our results can be found in the appendix.

Let $S = \{(x_i, y_i)\}_{i=1}^m$ be an m -sample where $x_i \in \mathcal{X} \subset \mathbb{R}^n$ and $y_i \in \mathcal{Y} = \{-1, +1\}$. Let $\mathbf{x} = \{x_1, \dots, x_m\}$ contain the inputs.

Definition 1 (Aizerman et al. (1964)). A kernel is a function K that for all $x, x' \in \mathcal{X}$ satisfies

$$K(x, x') = \langle \phi(x), \phi(x') \rangle,$$

where ϕ is a mapping from \mathcal{X} to an (inner product) Hilbert space \mathcal{H}

$$\phi : \mathcal{X} \mapsto \mathcal{H}.$$

Kernel learning algorithms make use of the $m \times m$ kernel matrix $K_{\mathbf{x}} = [K(x_i, x'_j)]_{i,j=1}^m$ defined using the training inputs \mathbf{x} . When using the kernel representation it is not always

possible to represent the weight vector w explicitly and so we can use the function f directly as the predictor:

$$f(x) = \sum_{i=1}^m \alpha_i K(x_i, x),$$

where $\alpha = (\alpha_1, \dots, \alpha_m)$ is the dual weight vector. Given a kernel K , learning can be described as finding a function from the class of functions (Srebro and Ben-David, 2006):

$$\mathcal{F}_K = \{x \mapsto \langle w, \phi(x) \rangle \mid \|w\| \leq 1, K(x, x') = \langle \phi(x), \phi(x') \rangle\}$$

minimising the hinge loss

$$\hat{h}^\gamma(f) = \frac{1}{m} \sum_{i=1}^m \max(\gamma - y_i f(x_i), 0).$$

For the generalisation error bounds we assume that the data are generated iid from a fixed but unknown probability distribution P over the joint space $\mathcal{X} \times \mathcal{Y}$. Given the *true error* of a function f :

$$\text{err}(f) = \mathbb{E}_{(x,y) \sim P}(y f(x) \leq 0),$$

the *empirical margin error* of f with margin $\gamma > 0$:

$$\hat{\text{err}}^\gamma(f) = \frac{1}{m} \sum_{i=1}^m \mathbb{I}(y_i f(x_i) < \gamma)$$

where \mathbb{I} is the indicator function, and the estimation error $\text{est}^\gamma(f)$

$$\text{est}^\gamma(f) = |\text{err}(f) - \hat{\text{err}}^\gamma(f)|,$$

we would like to find an upper bound for $\text{est}^\gamma(f)$. In the sequel we will state the bounds in standard form, where the true error $\text{err}(f)$ of a function f is upper bounded by the empirical error $\hat{\text{err}}(f)$ plus the estimation error $\text{est}(f)$:

$$\text{err}(f) \leq \hat{\text{err}}(f) + \text{est}(f). \quad (1)$$

Let $\mathcal{K} = \{K_1, \dots, K_D\}$ denote a family of kernels, where each kernel K_i is the i th *base kernel*. The following kernel families are formed using a linear or convex combination of base kernels:

$$\begin{aligned} \mathcal{K}_{\text{lin}}(K_1, \dots, K_D) &= \left\{ K^\eta = \sum_{i=1}^D \eta_i K_i \mid K^\eta \succcurlyeq 0, \sum_{i=1}^D \eta_i = 1 \right\} \\ \mathcal{K}_{\text{con}}(K_1, \dots, K_D) &= \left\{ K^\eta = \sum_{i=1}^D \eta_i K_i \mid \eta_i \geq 0, \sum_{i=1}^D \eta_i = 1 \right\}. \end{aligned}$$

These two kernel families are considered *finite dimensional*. The MKL problem can be described as finding a function f from the class:

$$\mathcal{F}_\mathcal{K} = \cup_{K \in \mathcal{K}} \mathcal{F}_K,$$

that minimises $\hat{h}^\gamma(f)$.

3.1 Sample Compression-Margin bound

In this section we use covering number bounds for learning the kernel with SVMs (Srebro and Ben-David, 2006) together with sample compression bounds (Littlestone and Warmuth, 1986; Floyd and Warmuth, 1995). We derive an upper bound for the above finite dimensional kernel families (that use 1-norm regularisation) using covering numbers, resulting in tighter bounds than Srebro and Ben-David (2006), when a small choice of base kernels is present in the final combination (*i.e.*, (Bach, 2008)). Before presenting our first result we define covering numbers (*e.g.*, see Anthony and Bartlett (1999)) for kernels and define sample compression schemes.

Definition 2 (covering number). A subset $A \subset \tilde{A}$ is an ϵ -net of A under the metric d if for any $a \in A$ there exists $\tilde{a} \in \tilde{A}$ with $d(a, \tilde{a}) \leq \epsilon$. The *covering number* $\mathcal{N}_d(A, \epsilon)$ is the size of the smallest ϵ -net of A .

Given a sample of points \mathbf{x} we can define the following ℓ_∞ metric:

$$d_\infty^{\mathbf{x}}(f_1, f_2) = \max |f_1(x_i) - f_2(x_i)|, \quad \forall i.$$

The uniform ℓ_∞ covering number $\mathcal{N}_m(\mathcal{F}, \epsilon)$ of a predictor class \mathcal{F} is given by considering all possible samples \mathbf{x} of size m :

$$\mathcal{N}_m(\mathcal{F}, \epsilon) = \sup_{|\mathbf{x}|=m} \mathcal{N}_{d_\infty^{\mathbf{x}}}(\mathcal{F}, \epsilon).$$

In the kernel learning scenario we have:

$$\begin{aligned} D_\infty^{\mathbf{x}}(K, \tilde{K}) &= \max |K(x_i, x_j) - \tilde{K}(x_i, x_j)|, \quad \forall i, j \\ &= \left| K_{\mathbf{x}} - \tilde{K}_{\mathbf{x}} \right|_\infty. \end{aligned}$$

A sample compression scheme is defined as follows.

Definition 3 (sample compression scheme). Let $\mathcal{A}(S)$ be the function output by learning algorithm \mathcal{A} on training set S . A sample compression scheme is a reconstruction function Φ mapping a compression set $\Lambda(S) \subset S$ to some set of functions \mathcal{F} such that $\mathcal{A}(S) = \Phi(\Lambda(S))$.

Sample compression bounds can be formed by simply using the cardinality $|\Lambda(S)|$ of the compression set, *e.g.*, an SVM is a compression scheme as it only requires the support vectors (compression set) in order to construct the same maximum margin classifier. Instead of a standard compression analysis, we will adapt the sample compression bounding technique to the following result of Srebro and Ben-David (2006), by using the cardinality $|\mathcal{K}|$ of the kernel family – viewing it as a “compression set”.¹

Theorem 1 (Srebro and Ben-David (2006)). *For any kernel family \mathcal{K} , bounded by $R \geq K(x, x)$ and with pseudo-dimension d , and any fixed $\gamma > 0$, with probability at least $1 - \delta$ over the choice of a random training set of size m we have:*

$$\text{err}(f) \leq \hat{\text{err}}^\gamma(f) + \sqrt{\frac{2 + d \log \frac{128em^3R}{\gamma^2d} + 256 \frac{R}{\gamma^2} \log \frac{\gamma em}{8\sqrt{R}} \log \frac{128mR}{\gamma^2} - \log \delta}{m}}.$$

We can apply a sample compression argument to this bound if our algorithm chooses $k < D$ kernels from a family of kernels $\mathcal{K} = \{K_1, \dots, K_D\}$ to get a bound in the form of Equation (1):

¹It is not a compression set in the standard sense, but we apply the technique developed by Hussain and Shawe-Taylor (2009) in order to apply compression bounds.

Theorem 2. *Let k be the number of kernels chosen. For any kernel family $\mathcal{K} = \{K_1, \dots, K_D\}$, bounded by R and any fixed $\gamma > 0$, with probability at least $1 - \delta$ over the choice of a random training set of size m we have:*

$$\text{err}(f) \leq \text{eir}^\gamma(f) + \sqrt{8 \frac{2 + k \log \frac{eD}{k} + k \log \frac{128em^3R}{\gamma^2k} + 256 \frac{R}{\gamma^2} \log \frac{\gamma em}{8\sqrt{R}} \log \frac{128mR}{\gamma^2} - \log \frac{\delta}{D}}{m}}.$$

This bound can be smaller than the bound of Srebro and Ben-David (2006) if there are a large number of kernels in the kernel family (possibly exponentially large) but only a small number of kernels chosen in the final combination. A recent algorithm by Bach (2008) has this property where he defines Hierarchical Multiple Kernel Learning, using ℓ_1 norm regularisation, which results in a sparse number of kernels being chosen in the final combination. The number of kernels used in the experiments in Bach (2008) were in the order of more than $D \geq 10^{30}$ but the algorithm chose a much small number of kernels $k = 300$ in the final solution. Also, our MKL algorithm from deliverable 3.1 will also deliver a sparse number of kernels as the normalisation parameter $\mu \rightarrow 1$. The above bound of Theorem 2 would apply in this scenario.

3.2 Rademacher Complexity bound

In this section we derive a novel Rademacher complexity bound (Bartlett and Mendelson, 2002; Shawe-Taylor and Cristianini, 2004) for MKL which does not have *multiplicative* behaviour between the margin complexity term and the dimensionality of the kernel family. Srebro and Ben-David (2006) state that it *may* not be possible to have *additive* behaviour for Rademacher bounds for MKL. However, we show that it is possible when one views the choice of kernels as weak learners. We begin by stating the following well-known concentration inequality.

Theorem 3 (McDiarmid (1989)). *Let X_1, \dots, X_n be independent random variables taking values in a set A , and assume that $f : A^n \mapsto \mathbb{R}$ satisfies*

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

Then for all $\epsilon > 0$

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

Next we first define the true and empirical Rademacher complexities (Bartlett and Mendelson, 2002).

Definition 4 (Rademacher complexity). For a sample $\mathbf{x} = \{x_1, \dots, x_m\}$ generated by a distribution \mathcal{D} on a set \mathcal{X} and a real-valued function class \mathcal{F} with domain \mathcal{X} , the *empirical Rademacher complexity* of \mathcal{F} is the random variable

$$\hat{R}_m(\mathcal{F}) = \mathbb{E}_\sigma \left[\sup_{f \in \mathcal{F}} \left| \frac{2}{m} \sum_{i=1}^m \sigma_i f(x_i) \right| \middle| x_1, \dots, x_m \right].$$

where $\sigma = (\sigma_1, \dots, \sigma_m)$ are independent uniform $\{\pm 1\}$ -valued (Rademacher) random variables. The (*true*) *Rademacher complexity* is:

$$R_m(\mathcal{F}) = \mathbb{E}_\mathbf{x} \left[\hat{R}_m(\mathcal{F}) \right] = \mathbb{E}_{\mathbf{x}\sigma} \left[\sup_{f \in \mathcal{F}} \left| \frac{2}{m} \sum_{i=1}^m \sigma_i f(x_i) \right| \right].$$

The standard Rademacher bounds for learning theory can be given as:

Theorem 4 (Bartlett and Mendelson (2002)). *Fix $\delta \in (0, 1)$ and let \mathcal{F} be a class of functions mapping from $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ to $[0, 1]$. Let $(z_i)_{i=1}^m$ be drawn independently according to a probability distribution \mathcal{D} . Then with probability $1 - \delta$ over random draws of samples of size m , every $f \in \mathcal{F}$ satisfies*

$$\begin{aligned} \mathbb{E}_{\mathcal{D}}[f(z)] &\leq \hat{\mathbb{E}}[f(z)] + R_m(\mathcal{F}) + \sqrt{\frac{\ln(1/\delta)}{2m}} \\ &\leq \hat{\mathbb{E}}[f(z)] + \hat{R}_m(\mathcal{F}) + 3\sqrt{\frac{\ln(2/\delta)}{2m}} \end{aligned}$$

These bounds are quite general and applicable to various learning algorithms if an *empirical Rademacher complexity* $\hat{R}_m(\mathcal{F})$ of the function class \mathcal{F} can be found efficiently. For kernel method algorithms a well-known result uses the trace of the kernel matrix to bound the empirical Rademacher complexity.

Theorem 5 (Bartlett and Mendelson (2002)). *If $K : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is a kernel, and $\mathbf{x} = \{x_1, \dots, x_m\}$ is a sample of points from \mathcal{X} , then the empirical Rademacher complexity of the class \mathcal{F}_B with bounded norm $\|w\|_2 \leq B$ satisfies*

$$\hat{R}_m(\mathcal{F}_B) \leq \frac{2B}{m} \sqrt{\sum_{i=1}^m K(x_i, x_i)} = \frac{2B}{m} \sqrt{\text{trace}(K_{\mathbf{x}})}.$$

The combination of the set of weak learners in boosting can be viewed as a convex hull:

$$\text{conv}_B(\mathcal{F}) = \left\{ \sum a_i f_i : f_i \in \mathcal{F}, a_i \in \mathbb{R}, a_i \geq 0, \sum a_i \leq B \right\} \quad (2)$$

We are interested in the empirical Rademacher complexity of a convex hull as given by Equation (2) (*i.e.*, boosting), because we can view the MKL problem in a similar way.

Theorem 6 (Shawe-Taylor (2009)). *The empirical Rademacher complexity of the convex hull $\text{conv}_B(\mathcal{F})$ of function class \mathcal{F} satisfies*

$$\hat{R}_m(\text{conv}_B(\mathcal{F})) \leq B \hat{R}_m(\mathcal{F}).$$

Given all of the results from above, we are now in a position to state the following theorem, which proves an upper bound for the empirical Rademacher complexity given multiple feature spaces.

Theorem 7. *Let $\mathbf{x} = \{x_1, \dots, x_m\}$ be an m -sample of points from \mathcal{X} , then the empirical Rademacher complexity $\hat{R}_m(\cup \mathcal{F}_j)$ of the class $\mathcal{F} = \mathcal{F}_1 \cup \dots \cup \mathcal{F}_k$, where each \mathcal{F}_j has rademacher complexity R_j , satisfies:*

$$\hat{R}_m(\cup \mathcal{F}_j) \leq \max_{1 \leq j \leq k} \hat{R}_m(\mathcal{F}_j) + 2\sqrt{\frac{\ln((k+1)/\delta)}{2m}}$$

Therefore we have the following generalisation error bound.

Theorem 8. *Let $\mathbf{x} = \{x_1, \dots, x_m\}$ be a randomly generated sample from distribution \mathcal{D} . Furthermore let $\mathcal{F} = \mathcal{F}_1 \cup \dots \cup \mathcal{F}_k$ be a joint feature space and K a (normalised) kernel function constructed using this joint space \mathcal{F} . Then for any $z \in \mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ and for all $f \in \mathcal{F}$, with probability at least $1 - \delta$ we have:*

$$\mathbb{E}_{\mathcal{D}}[f(z)] \leq \hat{\mathbb{E}}_{\mathcal{D}}[f(z)] + \frac{2}{m} \sqrt{\text{trace}(K_{\mathbf{x}})} + 5\sqrt{\frac{\ln((k+3)/\delta)}{2m}}.$$

We will now give details of a novel MKL algorithm that will also use this bound.

4 Algorithmic Extensions

The second important goal of the report is to use the analysis to help motivate new algorithms. The bound of Theorem 8 motivates the use of LPBoost to solve MKL problems, where the weights of the weak learners act like the weights of the kernels. The reason LPBoost is preferred to AdaBoost is because the bound is minimised when k is small, *i.e.*, when the number of chosen kernels is small. LPBoost carries out ℓ_1 norm regularisation resulting in much sparser solutions than AdaBoost, and therefore minimises the bound derived above. We begin by describing the LPBoost algorithm.

4.1 Linear Programme Boosting as an MKL problem

In the boosting framework (Freund and Schapire, 1997; Meir and Rätsch, 2003) the idea is to construct a convex combination (*i.e.*, a weighting that sums to 1) of “weak” learners – defined as classifiers that misclassify just under 50% of the time. For example, weak learners are usually defined as decision stumps, rays, *etc.* The idea is that a weighted combination of weak learners may be boosted to become a single strong learner.

Recall that $S = \{(x_i, y_i)\}_{i=1}^m$ is a training sample. Let \mathcal{H} be a class of “weak” functions, and $h_1, \dots, h_k \in \mathcal{H}$ be k weak learners. Let a_i be the weight of the i th weak learner, such that $\sum_{i=1}^k a_i = 1$. Let $u \in \mathbb{R}^m$ be the weights of the training inputs, and let $\mathcal{H}(S, u)$ be the base learning algorithm from which weak learners are chosen. After k weak learners are chosen the final hypothesis is defined as $f(x) = \sum_{i=1}^k a_i h_i(x)$. Therefore, the LPBoost objective can be written as (Demiriz et al., 2002):

$$\sum_{i=1}^k \alpha_i + C \sum_{i=1}^m \xi_i$$

where $\xi_i = \max(1 - y_i \sum_f \alpha_i f(x_i), 0)$ and $C \in \mathbb{R}$. Therefore $\xi_i > 0$ if x_i is misclassified. Let $u = (u_1, \dots, u_m)$ denote the dual variables (or boosting weights) for m data points. We can denote the restricted master problem for dual LPBoost like so (Demiriz et al., 2002):

$$\begin{aligned} \min \quad & \beta \\ \text{s.t.} \quad & \sum_{i=1}^m u_i y_i H_{ij} \leq \beta \\ & \sum_{i=1}^m u_i = 1 \end{aligned} \quad (3)$$

where $H_{ij} = h_j(x_i)$ is the classification of example x_i given weak learner h_j . Pseudocode for LPBoost solved using column generation is given in Algorithm 1

Let w_t denote the weight vector defined in the feature space $\phi_t : \mathcal{X} \mapsto \mathcal{F}_t$. From LPBoost we know the following optimisation problem needs to be fulfilled in order to add a weak learner $h_{t, w_t}(x) = \langle w_t, \phi_t(x) \rangle$:

$$\max_{t, w_t} \sum_{i=1}^m u_i y_i \langle w_t, \phi_t(x_i) \rangle > \beta. \quad (4)$$

Rewriting the lhs, by taking the summation inside the inner product we get

$$\max_{\|w_t\|=1} \left\langle w_t, \sum u_i y_i \phi_t(x_i) \right\rangle.$$

Furthermore, we know that $w_t = \sum u_i y_i \phi_t(x_i)$ for $i = 1, \dots, m$. It is clear that in order to satisfy $\|w_t\| = 1$ we need to normalise each vector $w_t = w_t / \|w_t\|$. Hence, substituting this above we get:

Algorithm 1: LPBoost via column generation (Demiriz et al., 2002)**Input:** training set S 1: initialise $n \leftarrow 0, a \leftarrow 0, \beta \leftarrow 0, u \leftarrow (\frac{1}{m}, \dots, \frac{1}{m})$ 2: **repeat**3: $n \leftarrow n + 1$ 4: $h_n \leftarrow \operatorname{argmax}_{h_j \in \mathcal{H}(S, u)} \sum_{i=1}^m u_i y_i h_j(x_i) > \beta$

5: solve restricted master problem:

$$(u, \beta) \leftarrow \begin{array}{ll} \operatorname{argmin} & \beta \\ \text{s.t.} & \sum_{i=1}^m u_i y_i h_p(x_i) \leq \beta \\ & p = 1, \dots, n \\ & \sum_{i=1}^m u_i = 1 \\ & 0 \leq u_i \leq D \end{array}$$

6: **until** $\sum_{i=1}^m u_i y_i h_j(x_i) \leq \beta$ for all $h_j \in \mathcal{H}(S, u)$ 7: $n \leftarrow n - 1$ **Output:** $a \leftarrow$ Lagrangian multipliers from last LP and $f = \sum_{p=1}^n a_p h_p$

$$\begin{aligned} & \left\langle \frac{w_t}{\|w_t\|}, \sum u_i y_i \phi_t(x_i) \right\rangle \\ &= \left\langle \frac{\sum u_i y_i \phi_t(x_i)}{\|\sum u_i y_i \phi_t(x_i)\|}, \sum u_i y_i \phi_t(x_i) \right\rangle \\ &= \frac{\sum_{i,j=1}^m u_i u_j y_i y_j K_t(x_i, x_j)}{\sqrt{\sum_{i,j=1}^m u_i u_j y_i y_j K_t(x_i, x_j)}} \\ &= \sqrt{\sum_{i,j=1}^m u_i u_j y_i y_j K_t(x_i, x_j)}. \end{aligned}$$

Hence, for the Multiple Kernel Learning problem we only need to maximise the following quantity:

$$v_{n+1} = \max_{\forall t} \sqrt{\sum_{i,j=1}^m u_i u_j y_i y_j K_t(x_i, x_j)} > \beta \quad (5)$$

and each weak learner added to the matrix H will be:

$$h_{t, w_t}(x_j) = \langle w_t, \phi_t(x_j) \rangle = \frac{1}{v_{n+1}} \sum_{i=1}^m u_i y_i K_t(x_i, x_j) \quad (6)$$

where $H_{t, n+1} = (h_{t, w_t}(x_1), \dots, h_{t, w_t}(x_m))'$. The pseudocode for this MKL algorithm is given in Algorithm 2. Furthermore, we can also have the following bound specialised to Algorithm 2, by using the result of Theorem 8.

Corollary 1. *Using all of the notation from Theorem 8 and letting f be the function output by LPBoost for MKL (Algorithm 2) using k kernels, then with probability $1 - \delta$ we have an upper bound for the true error of f as:*

$$\operatorname{err}(f) \leq \frac{1}{m} \sum_{i=1}^m \xi_i + \frac{2}{\sqrt{m}} + 5 \sqrt{\frac{\ln((k+3)/\delta)}{2m}}.$$

Algorithm 2: LPBoost for MKL

Input: training set S , set of k kernels $N = \{1, \dots, k\}$

1: initialise $n \leftarrow 0, \alpha \leftarrow 0, \beta \leftarrow 0, u \leftarrow (\frac{1}{m}, \dots, \frac{1}{m})$

2: **repeat**

3: $n \leftarrow n + 1$

4: $v_n \leftarrow \max_{t \in N} \sqrt{\sum_{i,j=1}^m u_i u_j y_i y_j K_t(x_i, x_j)} > \beta$

5: $h_n(x) \leftarrow \frac{1}{v_n} \sum_{j=1}^m u_j y_j K_t(x, x_j)$

6: solve restricted master problem:

$$(u, \beta) \leftarrow \begin{array}{ll} \operatorname{argmin} & \beta \\ \text{s.t.} & \sum_{i=1}^m u_i y_i h_p(x_i) \leq \beta \\ & p = 1, \dots, n \\ & \sum_{i=1}^m u_i = 1 \\ & 0 \leq u_i \leq D \end{array}$$

7: **until** $\sqrt{\sum_{i,j=1}^m u_i u_j y_i y_j K_t(x_i, x_j)} \leq \beta$ for all K_t

8: $n \leftarrow n - 1$

Output: $\alpha \leftarrow$ Lagrangian multipliers from last LP and $f = \sum_{p=1}^n \alpha_p K_p$

5 Conclusions

In this report we analysed MKL algorithms, and started with the proposition of a novel bound using sample compression theory (Theorem 2) applicable to the algorithm proposed in deliverable 3.1 (Hussain et al., December 2008). Furthermore, we also proposed a novel Rademacher bound (Theorem 8) for the case when we would have a function taken from a union of function classes. This inspired Algorithm 2 – a new MKL algorithm using the boosting framework. We also presented a bound specialised for this algorithm in Corollary 1.

We now briefly discuss the differences between the bounds. The Srebro and Ben-David (2006) bound is of the order:

$$\sqrt{\tilde{O} \left(\frac{d + 1/\gamma^2 - \ln \delta}{m} \right)}$$

where \tilde{O} hides logarithmic factors in its argument, d is the pseudo-dimension of the kernel class \mathcal{K} and γ is the margin. Theorem 2 is of the order:

$$\sqrt{\tilde{O} \left(\frac{k + 1/\gamma^2 - \ln(d/\delta)}{m} \right)}$$

where k are the number of kernels chosen from a possibility of d base kernels. This bound makes d extra applications of the Srebro and Ben-David (2006) result but does not require the size d of the kernel class, but a much smaller $k \ll d$ corresponding to the number of kernels chosen in the final combination of kernels. In the 1-norm case of MKL this bound would be tighter than the Srebro and Ben-David (2006) bound. Theorem 8 is of the order:

$$\mathcal{O} \left(\frac{1}{\sqrt{m}} \right) + \sqrt{\tilde{O} \left(\frac{\ln(k/\delta)}{m} \right)}.$$

This bound has an extra order of $1/\sqrt{m}$ but tighter constants in the \tilde{O} term. Also, it should be noted that this final bound does not use the margin γ , although this can be handled with Rademacher bounds (Shawe-Taylor and Cristianini, 2004) with some additional complexity.

The next step of this work would be to use the proposed Algorithm 2 as a method of finding a metric for the kernelised LinRel algorithm of Work Package 4. Furthermore, testing this algorithm in the PinView system against the MKL algorithm devised in D3.1 would also help determine which metric learning system to use in the final PinView system. This would culminate in the prototype required for the final year of the PinView project and deliverable 3.3.

Acknowledgements

We thank Jussi Kujala of TKK for his comments on this report.

Appendix

Proof. (of Theorem 2) From Anthony and Bartlett (1999) (Theorem 10.1) we have:

$$\sup_{f \in \mathcal{F}} \text{est}^\gamma(f) \leq \sqrt{8 \frac{1 + \log \mathcal{N}_{2m}(\mathcal{F}, \gamma/2) - \log \delta}{m}},$$

which is found by solving the following equation for $\epsilon > 0$:

$$2\mathcal{N}_{2m}(\mathcal{F}, \gamma/2) \exp\left(-\frac{\epsilon^2 m}{8}\right) = \delta.$$

From Theorem 1 and Lemma 3 of Srebro and Ben-David (2006) we have the following upper bound of the covering number for a family \mathcal{K} of kernels bounded by $R \geq K(x, x)$ and any $\alpha < 1$:

$$\mathcal{N}_m(\mathcal{F}_k, \alpha) \leq 2 \left(\frac{4em^3 R}{\alpha^2 k}\right)^k \left(\frac{16mR}{\alpha^2}\right)^{\frac{64R}{\alpha^2} \log\left(\frac{\alpha em}{8\sqrt{R}}\right)}. \quad (7)$$

Hence following Hussain and Shawe-Taylor (2009), and making use of the fact that we have $\binom{D}{k}$ different ways of choosing the kernels and making a further application of D we get:

$$\binom{D}{k} 2\mathcal{N}_{2n}(\mathcal{F}_k, \gamma/2) \exp\left(-\frac{\epsilon^2 n}{8}\right) = \frac{\delta}{D}.$$

Applying (7) and solving for ϵ gives the result. \square

Proof. (of Theorem 7) We know that:

$$\begin{aligned} \hat{R}_m(\cup \mathcal{F}_j) &= \mathbb{E}_\sigma \left[\sup_{f \in \cup \mathcal{F}_j} \left| \frac{2}{m} \sum_{i=1}^m \sigma_i f(x_i) \right| \right] \\ &\leq \sup_{f \in \cup \mathcal{F}_j} \left| \frac{2}{m} \sum_{i=1}^m \sigma_i f(x_i) \right| + 2\sqrt{\frac{\ln((k+1)/\delta)}{2m}} \\ &\leq \max_{1 \leq j \leq k} \sup_{f \in \mathcal{F}_j} \left| \frac{2}{m} \sum_{i=1}^m \sigma_i f(x_i) \right| + 2\sqrt{\frac{\ln((k+1)/\delta)}{2m}} \\ &\leq \max_{1 \leq j \leq k} \mathbb{E}_\sigma \sup_{f \in \mathcal{F}_j} \left| \frac{2}{m} \sum_{i=1}^m \sigma_i f(x_i) \right| + 2\sqrt{\frac{\ln((k+1)/\delta)}{2m}} \\ &= \max_{1 \leq j \leq k} \hat{R}_m(\mathcal{F}_j) + 2\sqrt{\frac{\ln((k+1)/\delta)}{2m}} \end{aligned}$$

where the second follows from an application of Theorem 3, the third line by observing that the supremum of a joint function class (*i.e.*, $\cup \mathcal{F}_j$) will always be upper bounded by the maximum function in one of the function classes, the next line by taking the expectation over σ to get the final line in terms of the empirical Rademacher complexity of a single function class \mathcal{F}_j . \square

Proof. (of Theorem 8) We view each feature space \mathcal{F}_j as the space for a new kernel. Putting together all the above results give:

$$\begin{aligned} \mathbb{E}_{\mathcal{D}}[f(z)] &\leq \hat{\mathbb{E}}[f(z)] + \hat{R}_m(\cup \mathcal{F}_j) + 3\sqrt{\frac{\ln(2/\delta)}{2m}} \\ &\leq \hat{\mathbb{E}}[f(z)] + B\hat{R}_m(\mathcal{F}_j) + 3\sqrt{\frac{\ln(2/\delta)}{2m}} \\ &\leq \hat{\mathbb{E}}[f(z)] + B \max_{1 \leq j \leq k} \hat{R}_m(\mathcal{F}_j) + 5\sqrt{\frac{\ln((k+3)/\delta)}{2m}} \\ &\leq \hat{\mathbb{E}}[f(z)] + \frac{2}{m} \sqrt{\sum_{i=1}^m K(x_i, x_i)} + 5\sqrt{\frac{\ln((k+3)/\delta)}{2m}}. \end{aligned}$$

Where the first line is given by Theorem 4, the second line comes from applying Theorem 6, the third by applying Theorem 7 and the final line by applying Theorem 5. \square

References

- M. Aizerman, E. Braverman, and L. Rozonoer. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25: 821 – 837, 1964.
- M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, Cambridge, 1999.
- F. R. Bach. Exploring large feature spaces with hierarchical multiple kernel learning. In *Advances in Neural Information Processing Systems*, 2008.
- P. L. Bartlett and S. Mendelson. Rademacher and gaussian complexities: risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.
- A. Demiriz, K. P. Bennett, and J. Shawe-Taylor. Linear programming boosting via column generation. *Machine Learning*, 46(13):225–254, 2002.
- S. Floyd and M. Warmuth. Sample compression, learnability, and the Vapnik-Chervonenkis dimension. *Machine Learning*, 21(3):269–304, 1995.
- Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55:119–139, 1997.
- Z. Hussain and J. Shawe-Taylor. Theory of matching pursuit. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 721–728. 2009.
- Z. Hussain, K. Pasupa, C. J. Saunders, and J. Shawe-Taylor. Basic metric learning. PinView FP7-216529 Project Deliverable Report D3.1, December 2008. URL <http://www.pinview.eu/deliverables.php>.

- G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5: 27–72, 2004.
- N. Littlestone and M. K. Warmuth. Relating data compression and learnability. Technical report, University of California Santa Cruz, Santa Cruz, CA, 1986.
- C. McDiarmid. On the method of bounded differences. In . L. M. S. L. N. Series, editor, *Surveys in Combinatorics 1989*, pages 148–188. Cambridge University Press, Cambridge, 1989.
- R. Meir and G. Rätsch. An introduction to boosting and leveraging. In *Advanced Lectures on Machine Learning, LNCS*, pages 119–184. Springer, 2003.
- J. Shawe-Taylor. Machine learning summer school tutorial on Learning Theory, 2009.
- J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, Cambridge, U.K., 2004.
- N. Srebro and S. Ben-David. Learning bounds for support vector machines with learned kernels. In *Computational Learning Theory*, volume 4005 of *Lecture Notes in Computer Science*, pages 169–183. Springer, 2006.
- Y. Ying and C. Campbell. Generalization Bounds for Learning the Kernel. In *Proceedings of the 22nd Annual Conference on Computational Learning Theory*. Springer, Berlin, 2009.