TOPICS IN TWO-SAMPLE TESTING

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF STATISTICS
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Nelson C. Ray
2013

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Susan P. Holmes) Principal Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Persi W. Diaconis)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Bradley Efron)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Jerome H. Friedman)

Approved for the University Committee on Graduate Studies.

Contents

Chapter 1

Multiple Kernels

In this chapter we introduce a framework for two sample testing based on heterogeneous data based on multiple kernel learning (MKL).

1.1 Introduction

Given a set of kernels, it is possible to combine them in order to produce new kernels. This is a starting point for heterogeneous data analysis: we can define a kernel K_i for each data domain and develop a kernel K that operates on the union of the domains. Rather, we typically shall produce a parametrized family of kernels K_θ and seek the "best" choice of parameters θ.

For example, the class of kernel functions on $\mathcal{X}$ is closed under pointwise products (also known as Schur products) of kernels,

$$K(x, x') = (K_1 K_2)(x, x') = K_1(x, x') K_2(x, x'),$$

tensor products of kernels,

$$K(x, x') = (K_1 \otimes K_2)(x_1, x_2, x_1', x_2') = K_1(x_1, x_1') K_2(x_2, x_2'),$$

and conic combinations of kernels,

$$K_\theta(x, x') = (\theta_1 K_1 + \ldots + \theta_m K_m)(x, x') = \theta_1 K_1(x, x') + \ldots + \theta_m K_m(x, x').$$

An unweighted sum of kernels is equivalent to concatenating the individual feature spaces.

1.2 Multiple Kernel Learning

In a landmark paper, Lanckriet et al. [12] showed that for various SVM objective functions, the problem of finding the optimal conic combination of kernels could be posed as a convex optimization problem. Although the initial approach involved solving a computationally expensive semidefinite program, this sparked a flurry of research on similar convex approaches to MKL.

Kloft et al. [**?**] conceived a general ℓ_p-norm approach to MKL, unifying many special cases and further proposed a highly efficient algorithm. This can be seen as generalizing Problem (**??**) of Chapter **??**.

Let $L : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ be a loss function, $\Omega : \mathcal{H} \to \mathbb{R}$ be a regularizer, and $\lambda > 0$ be a tradeoff parameter.

Kloft et al. consider linear models of the form

$$h_{\tilde{w},b,\theta}(x) = \sum_{i=1}^{M} \sqrt{\theta_m} \langle \tilde{w}_m, \phi_m(x) \rangle_{\mathcal{H}_m} + b = \langle \tilde{w}, \phi_\theta(x) \rangle_{\mathcal{H}} + b,$$

where $\tilde{w} = [\tilde{w}_1^T, \ldots, \tilde{w}_m^T]^T$ and $\phi_\theta = \sqrt{\theta_1}\phi_1 \times \ldots \times \sqrt{\theta_m}\phi_m$.

The regularized risk minimization problem is the following:

$$\min_{\tilde{w},b,\theta:\theta\succeq 0} \frac{1}{n} \sum_{i=1}^{n} L\left(\sum_{m=1}^{M} \sqrt{\theta_m} \langle \tilde{w}_m, \phi_m(x_i) \rangle_{\mathcal{H}_m} + b, y_i \right) + \frac{\lambda}{2} \sum_{m=1}^{M} \|\tilde{w}_m\|_{\mathcal{H}_m}^2 + \tilde{\mu}\tilde{\Omega}(\theta), \tag{1.1}$$

for $\tilde{\mu} > 0$.

Problem (1.1) is not convex but can be transformed into a convex problem via

the substitution

$$w_m \leftarrow \sqrt{\theta_m}\tilde{w}_m.$$

Decoupling the regularization parameter from the sample size by letting $\tilde{C} = \frac{1}{n\lambda}$ and $\mu \leftarrow \frac{\tilde{\mu}}{\lambda}$, and using convex regularizers of the form $\tilde{\Omega}(\theta) = \|\theta\|_p^2$, we get

$$\min_{w,b,\theta:\theta\succeq 0} \tilde{C}\sum_{i=1}^{n} L\left(\sum_{m=1}^{M}\langle w_m, \phi_m(x_i)\rangle_{\mathcal{H}_m} + b, y_i\right) + \frac{1}{2}\sum_{m=1}^{M}\frac{\|w_m\|_{\mathcal{H}_m}^2}{\theta_m} + \mu\|\theta\|_p^2, \quad (1.2)$$

where $\frac{t}{0} = 0$ if $t = 0$ and ∞ otherwise.

Kloft et al. prove that the Tikhonov-regularized Problem (1.2) with two parameters is in fact equivalent to the following Ivanov-regularized problem with one regularization parameter, C:

$$\begin{aligned} \underset{w,b,\theta:\theta\succeq 0}{\text{minimize}} \quad & C\sum_{i=1}^{n} L\left(\sum_{m=1}^{M}\langle w_m, \phi_m(x_i)\rangle_{\mathcal{H}_m} + b, y_i\right) + \frac{1}{2}\sum_{m=1}^{M}\frac{\|w_m\|_{\mathcal{H}_m}^2}{\theta_m} \\ \text{subject to} \quad & \|\theta\|_p^2 \leq 1. \end{aligned} \quad (1.3)$$

We use Problem (1.3) as implemented in SHOGUN [**?**].

1.3 Simulations

1.3.1 Vectorial Data Mixture Distribution

Let's look at mixtures of MVN. Let

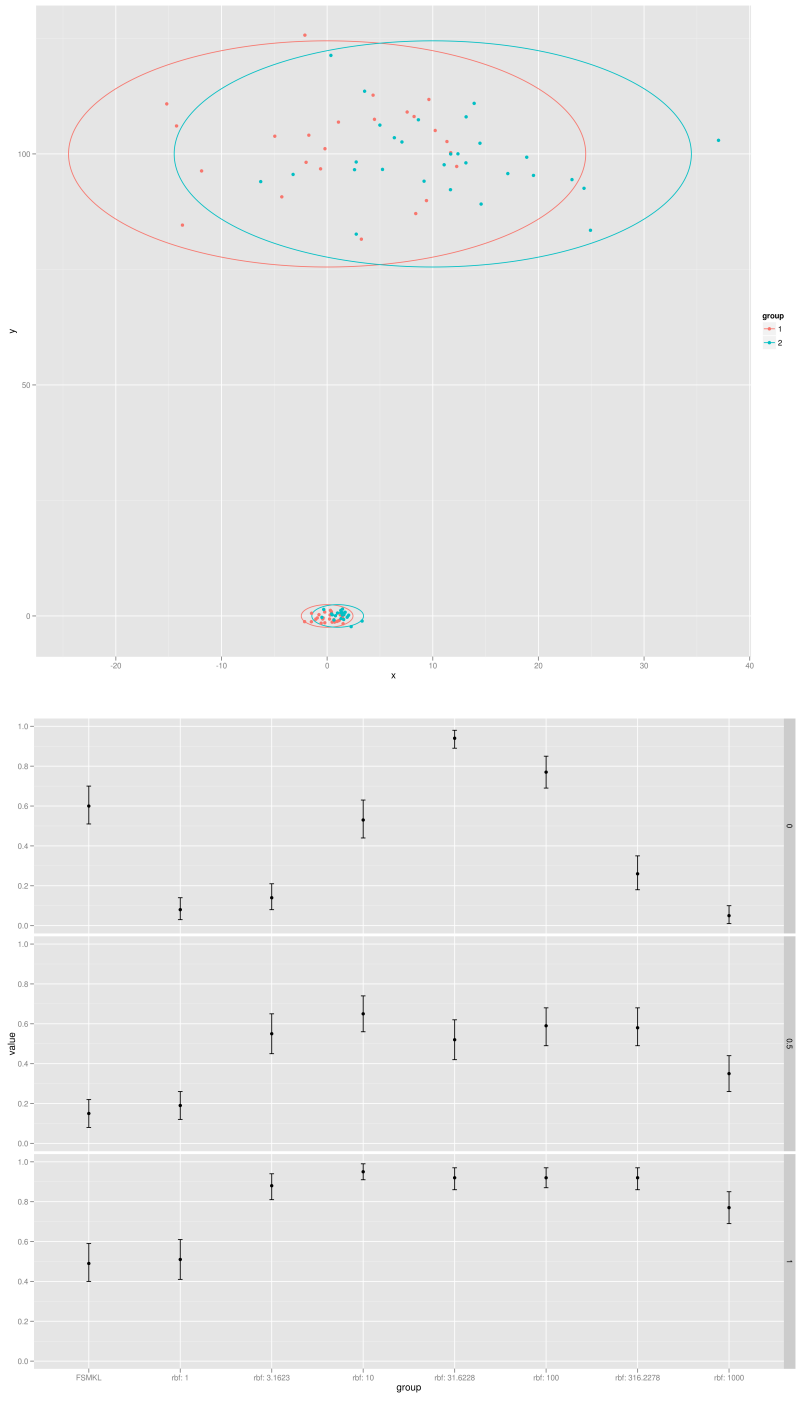
$$\Sigma_1 = \begin{bmatrix} 1^2 & 0 \\ 0 & 1^2 \end{bmatrix}$$
$$\Sigma_2 = \begin{bmatrix} 10^2 & 0 \\ 0 & 10^2 \end{bmatrix}$$
$$\mu_1(\delta_1) = [\delta_1, 0]^T$$
$$\mu_2(\delta_2) = [\delta_2, 100]^T.$$

Let d_1 be a mixture distribution of $\mathcal{N}_2([0,0]^T, \Sigma_1)$ with probability p and $\mathcal{N}_2([0,100]^T, \Sigma_2)$ with probability $1-p$. Let d_2 be a mixture distribution of $\mathcal{N}_2([1,0]^T, \Sigma_1)$ with probability p and $\mathcal{N}_2([10,100]^T, \Sigma_2)$ with probability $1-p$. Note that $\delta_1 = 1$ and $\delta_2 = 10$ were chosen to be one standard deviation away (on the x-axis, see $(\Sigma_1)_{1,1}$ and $(\Sigma_2)_{1,1}$). In all the simulations, we draw $n = 50$ samples from each mixture distribution, d_1 and d_2. We take the mixture probability $p = .5$ unless otherwise specified.

Here is a plot of the 95% confidence ellipses of the mixture distributions:

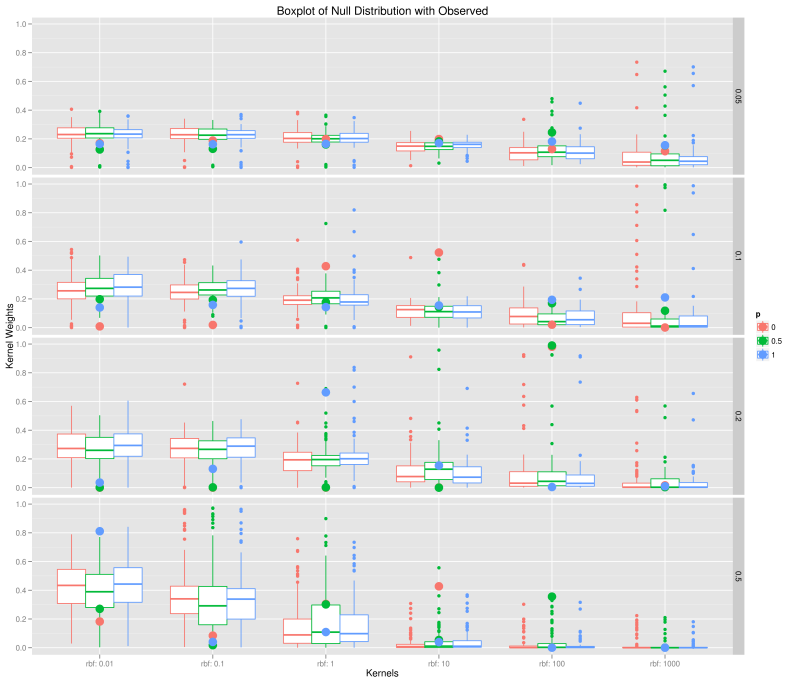
Here we plot the average power and bootstrap 95% confidence intervals (100 simulations) for each RBF kernel individually and MKL on all of them, faceted on the mixture probability. So for $p = 0$, we have all the weight on $\mathcal{N}_2(\mu_2, \Sigma_2)$, and for $p = 1$, we have all the weight on $\mathcal{N}_2(\mu_1, \Sigma_1)$. Since the latter is on a smaller scale, we expect the smaller width RBF kernels to do better. We take $C = 1$, and the widths to be from the middle run of the last section: The smaller distribution ($p = 1$) has higher power for smaller kernels, but the MKL power is about the same as compared with the $p = 0$ case. Both outperform the mixed setting.

Here are the null distributions of the weights and the observed weights:



1.3.2 Heterogeneous Data

I created a test heterogeneous dataset that has two components to it: DNA string and univariate. I created the DNA data via a Markov chain because I wanted the joint



distribution of 2-grams to be different from the product of two 1-grams. I wanted this dependence so that I could later pick out differences between two groups with a 2-spectrum kernel instead of a 1-spectrum kernel. For the first group, I randomly picked a starting string according to the stationary distribution and then proceeded via the transition probabilities. For the second group, I had independent draws from the stationary distribution.

The univariate data is simply $\mathcal{N}(\{-\mu, \mu\}, 1)$, and there are 20 samples in each group. I fixed the kernel training parameter values and trained a convex combination of 5 kernels on the data via MKL: Gaussian RBF kernels with parameter .1, .2, .5, and 1, and a 2-spectrum kernel (later I do want to test that the 2-spectrum is required in this case over the 1-spectrum because of the Markov chain construction of the data, but I had to add pre-processors and I'm still not too familiar with shogun yet).

I looked at the kernel weights (so no Friedman test yet) over $\mu = .5, .6, \ldots, 2.9, 3$. The weights on, say, the 2-spectrum kernel aren't monotonically decreasing because of sampling variance: I only ran each of these once. I have attached MKL1.png to show that the signal in the DNA data outweighs (in the sense of yielding a higher

MKL weight) the signal in the univariate part of the data until $\mu = 1.5$ or so. At that point, the RBF1 kernel takes over.

The pictures from our meeting weren't that compelling, so I've been looking for better examples. I decided a reasonable one was the Christmas star example on page 1548 of http://eprints.pascal-network.org/archive/00002269/01/sonnenburg06a.pdf

Imagine an outer star (radius 4, 5, 6, 7, 8) with an inner star (radius 4) inside. The different stars correspond to different labelings in the classification problem. I looked at 5 kernels individually (RBF with width .01, .1, 1, 10, 1000) and the MKL combination of all of them).

Here's a heterogeneous data example. I added string (DNA) data to the Christmas star example from last week. So each point is (l_i, x_i, y_i, s_i) = (label, Christmas star x-coordinate, Christmas star y-coordinate, DNA sequence). I generated the DNA by first picking a random (Poisson) length, sampling the starting point from the stationary distribution (all 1/4) of the Markov chain, and then picking according to the transition matrix M, where $M_{i,i} = s$ (for self transition probability) and $M_{i,j} = \frac{(1-s)}{3}$ for $i \neq j$.

Here are the MKL weights:

And the power:

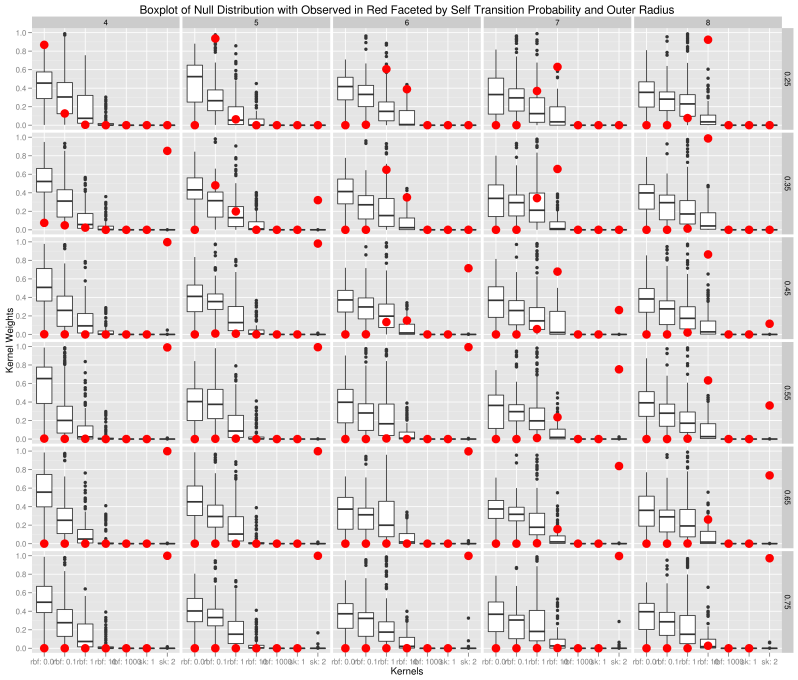
The effect of C:

C clearly has an effect, especially if you get it very wrong (>.8). I'm a little disappointed in the performance of MKL but pleased that it does pick out the right structure in the data. I'd like to say that if you know the structure of the data a priori and use that in the kernel, you will obviously get the best performance. It seems like you give up a lot of performance using MKL (maybe too much to justify its convenience), only doing better than the worst choices of kernels given.

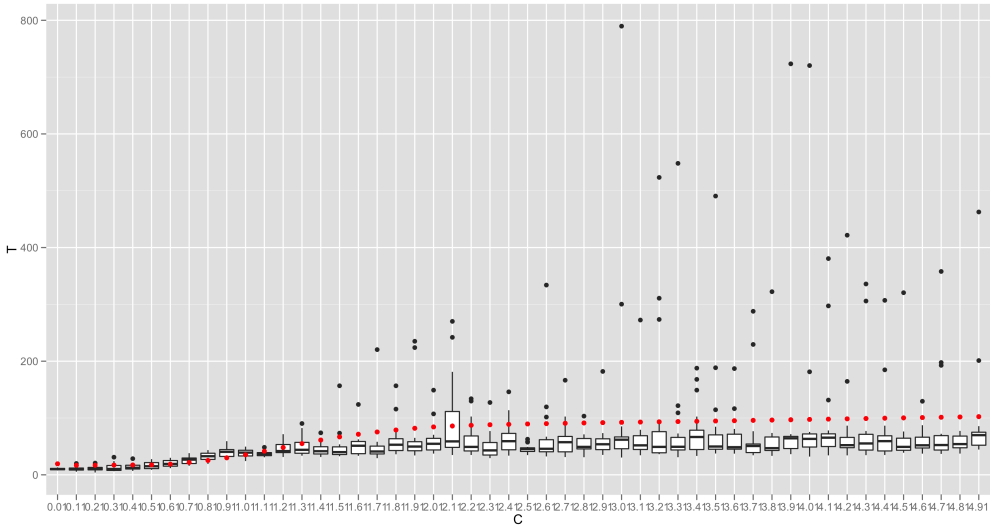
MKL can pick out the structure of the data:

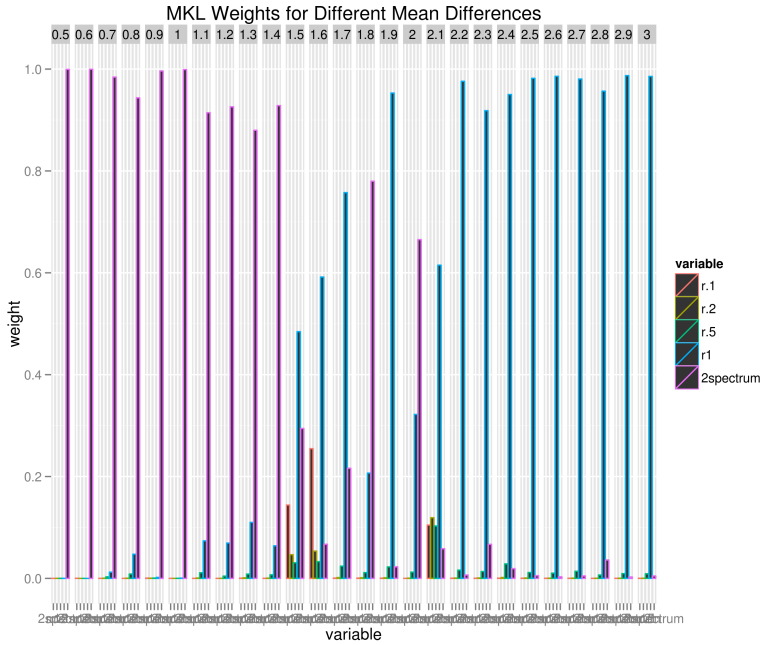
1.4 Wine Example

TODO


Boxplot of Null Distribution with Observed in Red Faceted by Self Transition Probability and Outer Radius
Kernel Weights
Kernels


Power (Christmas Star + DNA Example, C = .1, n = 50)
Power
Kernel
group
FSMKL
rbf: 1
rbf: 10
sk: 1
sk: 2


T
C


MKL Weights for Different Mean Differences
weight
variable
variable
r.1
r.2
r.5
r1
2spectrum

References

[1] A. Anonymous and B. Anonymous. A rate of convergence bound for the randomization t-distribution using stein's method. *Unpublished Technical Report*.

[2] K.M. Borgwardt, A. Gretton, M.J. Rasch, H.P. Kriegel, B. Schölkopf, and A.J. Smola. Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22(14):e49–e57, 2006.

[3] L.H.Y. Chen, L. Goldstein, and Q.M. Shao. *Normal Approximation by Stein's Method*. Springer Verlag, 2010.

[4] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, 2007.

[5] J.H. Friedman. On Multivariate Goodness–of–Fit and Two–Sample Testing. *Proceedings of Phystat2003, http://www.slac.stanford.edu/econf/C*, 30908, 2003.

[6] Jeff Gentry. *twitteR: R based Twitter client*, 2011. R package version 0.99.6.

[7] M. Gönen and E. Alpaydın. Multiple kernel learning algorithms. *Journal of Machine Learning Research*, 12:2211–2268, 2011.

[8] A. Gretton, KM Borgwardt, M. Rasch, B. Schölkopf, and AJ Smola. A kernel method for the two-sample-problem. *Advances in Neural Information Processing Systems*, 19:513–520, 2007.

[9] A. Gretton, K.M. Borgwardt, M.J. Rasch, B. Schölkopf, and A. Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13:723–773, 2012.

[10] A. Gretton, K. Fukumizu, Z. Harchaoui, and B.K. Sriperumbudur. A fast, consistent kernel two-sample test. *Advances in Neural Information Processing Systems*, 22:673–681, 2010.

[11] Alexandros Karatzoglou, Alex Smola, Kurt Hornik, and Achim Zeileis. kernlab – an S4 package for kernel methods in R. *Journal of Statistical Software*, 11(9):1–20, 2004.

[12] G.R.G. Lanckriet, N. Cristianini, P. Bartlett, L.E. Ghaoui, and M.I. Jordan. Learning the kernel matrix with semidefinite programming. *The Journal of Machine Learning Research*, 5:27–72, 2004.

[13] E.L. Lehmann. *Elements of large-sample theory*. Springer Verlag, 1999.

[14] E.L. Lehmann and J.P. Romano. *Testing Statistical Hypotheses*. Springer Verlag, 2005.

[15] C. Leslie, E. Eskin, and W.S. Noble. The spectrum kernel: A string kernel for SVM protein classification. In *Proceedings of the Pacific Symposium on Biocomputing*, volume 7, pages 566–575. Hawaii, USA., 2002.

[16] H.T. Lin, C.J. Lin, and R.C. Weng. A note on platt's probabilistic outputs for support vector machines. *Machine learning*, 68(3):267–276, 2007.

[17] M. Panov, A. Tatarchuk, V. Mottl, and D. Windridge. A modified neutral point method for kernel-based fusion of pattern-recognition modalities with incomplete data sets. *Multiple Classifier Systems*, pages 126–136, 2011.

[18] J. Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

[19] N. Poh, D. Windridge, V. Mottl, A. Tatarchuk, and A. Eliseyev. Addressing missing values in kernel-based multimodal biometric fusion using neutral point substitution. *Information Forensics and Security, IEEE Transactions on*, 5(3):461–469, 2010.

[20] S. Qiu and T. Lane. Multiple kernel learning for support vector regression. *Computer Science Department, The University of New Mexico, Albuquerque, NM, USA, Tech. Rep*, 2005.

[21] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2011. ISBN 3-900051-07-0.

[22] B. Schölkopf and A.J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. the MIT Press, 2002.

[23] C. Stein. Approximate computation of expectations. *Lecture Notes-Monograph Series*, 7, 1986.

[24] G. Wahba. Multivariate function and operator estimation, based on smoothing splines and reproducing kernels. In *SANTA FE INSTITUTE STUDIES IN THE SCIENCES OF COMPLEXITY-PROCEEDINGS VOLUME-*, volume 12, pages 95–95. ADDISON-WESLEY PUBLISHING CO, 1992.

[25] G. Wahba et al. Support vector machines, reproducing kernel hilbert spaces and the randomized gacv. *Advances in Kernel Methods-Support Vector Learning*, 6:69–87, 1999.