Extensions of Gaussian Processes for Ranking: Semi-supervised and Active Learning

Wei Chu
Gatsby Computational Neuroscience Unit
University College London
London, WC1N 3AR, UK
chuwei@gatsby.ucl.ac.uk

Zoubin Ghahramani
Gatsby Computational Neuroscience Unit
University College London
London, WC1N 3AR, UK
zoubin@gatsby.ucl.ac.uk

Abstract

Unlabelled examples in supervised learning tasks can be optimally exploited using semi-supervised methods and active learning. We focus on ranking learning from pairwise instance preference to discuss these important extensions, semi-supervised learning and active learning, in the probabilistic framework of Gaussian processes. Numerical experiments demonstrate the capacities of these techniques.

1 Introduction

Ranking learning is an important supervised problem of learning a ranking or ordering on instances, which has attracted considerable attention in machine learning research recently. Ranking learning, in which the training data are collected or interpreted in the form of pairwise instance preference, is also known as preference learning (Fürnkranz and Hüllermeier, 2005). These learning problems frequently arise in many applications, such as decision making, collaborative filtering and drug discovery. Various learning algorithms have been developed for preference learning, e.g. large margin classifiers (Herbrich et al., 1998; Aiolli and Sperduti, 2004), constraint classification approaches (Har-Peled et al., 2002), boosting-based learning algorithms (Dekel et al., 2004), neural networks with gradient descent methods (Burges et al., 2005) and probabilistic kernel approaches based on Gaussian processes (Chu and Ghahramani, 2005).

Many applications of ranking learning involve a large number of unlabelled examples and a few labelled examples, as expensive human effort is usually required in labelling examples. The issue of effectively exploiting the information in the unlabelled instances to facilitate supervised learning has been extensively studied under the name *semi-supervised learning* (Belkin et al., 2004; Chapelle et al., 2003; Lawrence and Jordan, 2005; Zhou et al., 2004; Zhu et al., 2003). Another issue, known as *active learning* (Cohn et al., 1995; Baram et al., 2004), concerns efficiently selecting queries from the unlabelled data pool sequentially to expedite the learning process.

This paper describes a fully probabilistic approach to active and semi-supervised ranking learning from pairwise instance preference using Gaussian processes. Inspired by the attractive concept of "warped RKHS" (Sindhwani et al., 2005b), semi-supervised Gaussian processes (Sindhwani et al., 2005a) have been recently developed which provide a general framework to incorporate unlabelled data for various supervised learning tasks and a principled way for model selection. We adapt this semi-supervised Gaussian process framework for ranking learning and further discuss active learning using information-theoretic criteria.

2 Pairwise Instance Preference

Consider a set of n distinct instances $x_i \in \mathcal{R}^d$ denoted as $\mathcal{X} = \{x_i : i = 1, \ldots, n\}$, and a set of K observed pairwise preference relations on the instances, denoted as

$$\mathcal{D} = \{v_k \succ u_k : k = 1, \ldots, K\} \tag{1}$$

where $v_k \in \mathcal{X}$, $u_k \in \mathcal{X}$, and $v_k \succ u_k$ means the instance v_k is ranked higher than u_k. For example, the pair $\{v_k, u_k\}$ could be two movies, while the user may prefer v_k to u_k. Note that only a small subset of $\mathcal{X}$ is labelled in $\mathcal{D}$.

Like previous approaches (Thurstone, 1927; Elo, 1978), we assume that each training sample x_i is associated with an unobservable latent function value $f(x_i)$, and these function values $\{f(x_i)\}$ preserve the preference relations in $\mathcal{D}$. We impose a Gaussian process prior on $\{f(x_i)\}$, and employ an appropriate likelihood function to learn the user's preference from the pairwise preferences between samples.

2.1 Semi-supervised Gaussian Processes

The latent function values $\{f(x_i)\}$ are assumed to be a realization of random variables in a zero-mean Gaussian process (Williams and Rasmussen, 1996). Traditionally the covariance between the latent functions corresponding to a pair of instances x_i and x_j can be defined by any reproducing kernel. A simple example is the Gaussian kernel defined as

$$\mathcal{K}(x_i, x_j) = \kappa_o \exp\left(-\frac{\kappa}{2}\sum_{\ell=1}^{d}(x_i^\ell - x_j^\ell)^2\right) \tag{2}$$

where $\kappa_o > 0$, $\kappa > 0$ and x_i^ℓ denotes the ℓ-th element of x_i. Thus the prior probability of these latent function values $\{f(x_i)\}$ is a multivariate Gaussian

$$\mathcal{P}(\boldsymbol{f}|\mathcal{X}) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\boldsymbol{f}^T\Sigma^{-1}\boldsymbol{f}\right) \tag{3}$$

where $\boldsymbol{f} = [f(x_1), f(x_2), \ldots, f(x_n)]^T$, and Σ is the $n \times n$ covariance matrix whose ij-th element is the covariance function $\mathcal{K}(x_i, x_j)$ defined as in (2). Obviously this prior (3) does not contain any information of the abundant unlabelled data.

Semi-supervised Gaussian processes (Sindhwani et al., 2005a) provide a novel prior that exploits the localized spatial structure spanned by the given unlabelled data. A graph $\mathcal{G}$ can be defined over $\mathcal{X}$ by putting an edge between pairs of neighboring data points. For example, the neighbor set of each data point can be the k nearest neighbors simply. The resulting graph is denoted as $\mathcal{G} = (\mathcal{X}, \mathcal{E})$ where $\mathcal{E} \subset \mathcal{X} \times \mathcal{X}$ is the set of edges. Learning on the graph leads to the following (approximate) likelihood

$$\mathcal{P}(\mathcal{G}|\boldsymbol{f}) \propto \exp\left(-\frac{1}{2}\boldsymbol{f}^T L\boldsymbol{f}\right). \tag{4}$$

where L is defined to be the combinatorial Laplacian of the graph $\mathcal{G}$. For an unweighted graph the combinatorial Laplacian L is a sparse matrix where L_{ij} is equal to -1 (resp. 0) if $i \neq j$ and vertices x_i and x_j are connected (resp. disconnected), and $L_{ii} = -\sum_{j:j\neq i} L_{ij}$. The posterior distribution of $\boldsymbol{f}$ on the graph $\mathcal{G}$ is proportional to $\mathcal{P}(\mathcal{G}|\boldsymbol{f})\mathcal{P}(\boldsymbol{f})$, which is a multivariate Gaussian as follows

$$\mathcal{P}(\boldsymbol{f}|\mathcal{G}, \mathcal{X}) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma^{-1}+L|^{-\frac{1}{2}}} \exp\left(-\frac{1}{2}\boldsymbol{f}^T(\Sigma^{-1}+L)\boldsymbol{f}\right). \tag{5}$$

The posterior distribution will be used as the prior distribution for the following supervised learning problem. Note that the new prior (5) is applicable to any data points in $\mathcal{R}^d$.

2.2 Likelihood Function and Posterior Distribution

The observation that v_k is preferred to u_k can be simply preserved by an inequality relation $f(v_k) > f(u_k)$. Using the similar ideas in the Thurstonian model (Thurstone, 1927) and Árpad Élő's Chess ranking system (Elo, 1978), we define a likelihood function for noisy observations on pairwise preference relations as follows,

$$\mathcal{P}\left(v_k \succ u_k | f_k\right) = \Phi\left(z_k\right) \tag{6}$$

where $f_k = [f(v_k), f(u_k)]^T$, $z_k = \frac{f(v_k)-f(u_k)}{\sqrt{2}\sigma}$ and $\Phi(z) = \int_{-\infty}^{z} \mathcal{N}(\gamma; 0, 1) d\gamma$ is the cumulative normal. Here $\sigma > 0$ reflects the noise level in measurement. Based on Bayes' theorem, the posterior probability can then be written as

$$\mathcal{P}(\boldsymbol{f}|\mathcal{D}) = \frac{\mathcal{P}(\boldsymbol{f})}{\mathcal{P}(\mathcal{D})}\mathcal{P}(\mathcal{D}|\boldsymbol{f}) = \frac{\mathcal{P}(\boldsymbol{f})}{\mathcal{P}(\mathcal{D})}\prod_{k=1}^{K} \mathcal{P}\left(v_k \succ u_k | f_k\right) \tag{7}$$

where the prior distribution $\mathcal{P}(\boldsymbol{f})$ is as defined in (5), the likelihood function is as defined in (6), and the normalization factor $\mathcal{P}(\mathcal{D}) = \int \mathcal{P}(\mathcal{D}|\boldsymbol{f})\mathcal{P}(\boldsymbol{f}) d\boldsymbol{f}$ is known as the evidence of the model parameters $\{\kappa_o, \kappa, \sigma\}$.

2.3 Expectation Propagation

Several approximate Bayesian inference methods can be applied to approximate the posterior distribution (7) as a Gaussian. In this work, we apply the expectation propagation (EP) algorithm (Minka, 2001; Csató, 2002). EP attempts to approximate $\mathcal{P}(\mathcal{D}|\boldsymbol{f})\mathcal{P}(\boldsymbol{f})$ as a parametric product distribution in the form $\mathcal{Q}(\boldsymbol{f}) = \mathcal{P}(\boldsymbol{f})\prod_{k=1}^{K}\tilde{t}(f_k) = \mathcal{P}(\boldsymbol{f})\prod_{k=1}^{K} s_k \exp\left(-\frac{1}{2}(f_k - m_k)^T \pi_k (f_k - m_k)\right)$, where $m_k = [m_{u_k}, m_{v_k}]^T$ and π_k is a 2×2 matrix. The parameters $\{s_k, m_k, \pi_k\}$ in $\{\tilde{t}(f_k)\}$ are successively optimized by minimizing the KL divergence,

$$\tilde{t}(f_k)^{\text{new}} = \arg\min_{\tilde{t}(f_k)} \mathbf{KL}\left(\frac{\mathcal{Q}(\boldsymbol{f})}{\tilde{t}(f_k)^{\text{old}}}\mathcal{P}(v_k \succ u_k|f_k) \,\middle\|\, \frac{\mathcal{Q}(\boldsymbol{f})}{\tilde{t}(f_k)^{\text{old}}}\tilde{t}(f_k)\right). \tag{8}$$

Since $\mathcal{Q}(\boldsymbol{f})$ is in the exponential family, this minimization can be simply solved by moment matching up to the second order. A detailed updating scheme can be found in Appendix A. The formulation is particularly useful in sequential learning, though there is no guarantee of convergence. At the equilibrium of $\mathcal{Q}(\boldsymbol{f})$, we obtain an approximate evidence defined as in (11) for model selection.

3 Active Learning

Learning could be made more efficient if we can actively select salient data points. Within the Bayesian learning framework, the expected informativeness of a new observation can be measured by the change in entropy of the posterior distribution of the latent functions by the inclusion of the candidate (MacKay, 1992; Lawrence et al., 2002). The new posterior distribution with the inclusion of the *unused* preference relation $(v_k \succ u_k)$ can be approximated as a Gaussian $\mathcal{N}(\boldsymbol{f}; \mathcal{A}^{new}, \boldsymbol{h}^{new})$ as described in Appendix A. The entropy gain can be evaluated by

$$\Delta\mathbf{H}_{v_k \succ u_k} = -\frac{1}{2}\log\det(\lambda_k^{new}\lambda_k^{-1}) \tag{9}$$

where λ_k and λ_k^{new} are as defined in Appendix A. For unlabelled preference relations, we can evaluate the expected entropy gain as the criterion,

$$\langle\Delta\mathbf{H}_k\rangle = \mathcal{P}(v_k \succ u_k|\mathcal{D})\Delta\mathbf{H}_{v_k \succ u_k} + \mathcal{P}(v_k \prec u_k|\mathcal{D})\Delta\mathbf{H}_{v_k \prec u_k} \tag{10}$$

where the predictive probability $\mathcal{P}(v_k \succ u_k|\mathcal{D})$ is evaluated by $\mathcal{Z}_k$ as in Appendix A. As a much more expensive alternative, the predictive distributions of all the unlabelled data can be updated by the inclusion of the candidate and then the sum of the expected entropy gain of the updated predictive distributions over all the unlabelled data could be used as the score. The strategy is to select the sample with the highest score from the data pool.

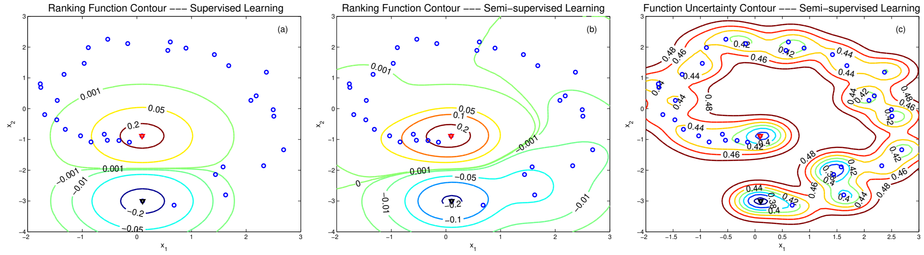

Figure 1: Contour graphs of the results on a synthetic spiral dataset. The dataset is represents by blue circles and the labelled pair is marked as triangles. The error bars of ranking functions in the posterior distribution are presented in graph (c).

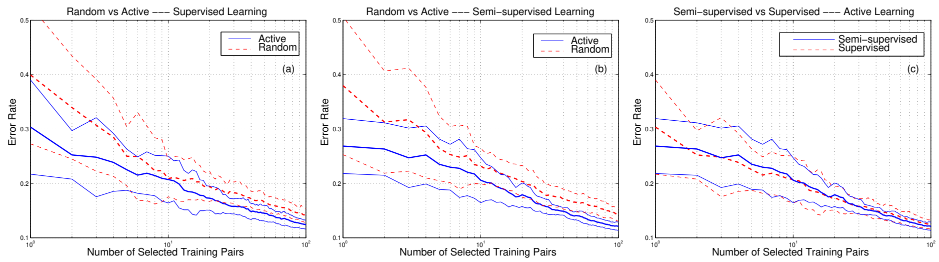

Figure 2: Performance of active learning on Boston Housing dataset. The error rate is the percent of incorrect preference predictions on 127765 pairs. The curves present the average over 20 trials along with standard deviation.

4 Demonstration

An artificial dataset was generated as shown in Figure 1, which is composed of 35 data points scattered along a spiral. Only one pair is labelled by preference. We used the Gaussian kernel defined as in (2) and the model parameters were set at $\kappa_o = 0.25$, $\kappa = 4.0$ and $\sigma^2 = 0.01$. The ranking function obtained by supervised learning is presented in Figure 1(a). We applied 3-nearest-neighbor to setup the graph $\mathcal{G}$ for semi-supervised learning and the corresponding results are presented in Figure 1(b-c). Clearly semi-supervised Gaussian processes have captured the density information that also significantly changed the ranking function and its error bars.

The Boston Housing dataset was used for active learning, which consists of 506 samples with 13 features.[1] The ranking over these samples was decided by the "housing price". We sequentially selected a pair of samples from the unlabelled data pool (1000 randomly selected unlabelled pairs) for query and then added the labelled pair into training. We used a Gaussian kernel and the model parameters were set at $\kappa_o = 0.0625$, $\kappa = 0.0625$ and $\sigma^2 = 0.001$. The graph for semi-supervised learning was constructed by 3-nearest-neighbor. We compared the performance of using the entropy criterion defined as in (10) against random selection in Figure 2(a-b). The entropy criterion works greatly better than random selection on both supervised and semi-supervised learning. We also compared active semi-supervised learning to active supervised learning in Figure 2(c). The active learning results are comparable on this case, whereas semi-supervised learning yields slightly better results than supervised learning after selecting 30 pairs.

Acknowledgment

This work was supported by the National Institutes of Health Grant Number 1 P01 GM63208. W. Chu thanks S. Sathiya Keerthi and Vikas Sindhwani for many discussions.

[1]The original data can be found in StatLib via http://lib.stat.cmu.edu/datasets/boston.

A EP Algorithm Outline

Let us define an augmented matrix Π_k for π_k, which is a $n \times n$ matrix with only four non-zero entries from π_k. Similarly we define a $n \times 1$ vector M_k for m_k. In notation of Π_k and M_k, we have $\mathcal{Q}(\boldsymbol{f}) = \mathcal{P}(\boldsymbol{f}) \prod_{k=1}^{K} s_k \exp\left(-\frac{1}{2}(\boldsymbol{f} - M_k)^T \Pi_k (\boldsymbol{f} - M_k)\right)$ equivalently, which is a Gaussian distribution $\mathcal{N}(\boldsymbol{f}; \boldsymbol{h}, \mathcal{A})$ with covariance $\mathcal{A} = (\Sigma^{-1} + L + \Pi)^{-1}$ and mean $\boldsymbol{h} = \mathcal{A}\Psi$ where $\Pi = \sum_{k=1}^{K} \Pi_k$ and $\Psi = \sum_{k=1}^{K} \Pi_k M_k$.

The initial states:

- site matrix $\pi_k = 0$, and site vector $\psi_k = \pi_k m_k = 0$, $\forall k$;
- site amplitude $s_k = 1$, posterior mean $\boldsymbol{h} = 0$, and posterior variance $\mathcal{A} = (\Sigma^{-1} + L)^{-1}$;

Looping k from 1 to K until there is no significant change in $\{\psi_k, \pi_k, s_k\}_{k=1}^{K}$:

- $\tilde{t}(f_k)^{old}$ is removed from $\mathcal{Q}(\boldsymbol{f})$, which leads to the leave-one-out distribution $\mathcal{Q}^{\setminus i}(\boldsymbol{f}) = \frac{\mathcal{Q}(\boldsymbol{f})}{\tilde{t}(f_k)^{old}}$ having (for *unused* preference relations $\lambda_k^{\setminus k} = \lambda_k$ and $h_k^{\setminus k} = h_k$)
  - covariance of f_k: $\lambda_k^{\setminus k} = (\lambda_k^{-1} - \pi_k)^{-1}$, where λ_k is a 2×2 sub-matrix of $\mathcal{A}$ with entries associated with the instances u_k and v_k;
  - mean of f_k: $h_k^{\setminus k} = h_k + \lambda_k^{\setminus k}(\pi_k h_k - \psi_k)$, where h_k is a 2×1 vector with entries of the posterior mean of $f(u_k)$ and $f(v_k)$;
- By incorporating $\mathcal{P}(v_k \succ u_k | f_k)$ into $\mathcal{Q}^{\setminus i}(\boldsymbol{f})$, the parameters in $\tilde{t}(f_k)$ can be computed as follows:
  - Let us denote $\mathbf{1}_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$ and $\mathbf{1}_2 = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$;
  - $\mathcal{Z}_k = \int \mathcal{P}(v_k \succ u_k | f_k) \mathcal{N}(f_k; h_k^{\setminus k}, \lambda_k^{\setminus k})\, df_k = \Phi(\tilde{z}_k)$, where $\tilde{z}_k = \frac{\mathbf{1}_1^T h_k^{\setminus k}}{\sigma_\star}$ and $\sigma_\star^2 = 2\sigma^2 + \mathrm{trace}\left(\lambda_k^{\setminus k} \mathbf{1}_2\right)$;
  - $\gamma_k = \frac{\partial \log \mathcal{Z}_k}{\partial h_k^{\setminus k}} = \frac{\mathcal{N}(\tilde{z}_k)}{\sigma_\star \Phi(\tilde{z}_k)} \mathbf{1}_1$, and $\beta_k = \frac{\partial \log \mathcal{Z}_k}{\partial \lambda_k^{\setminus k}} = \frac{-\tilde{z}_k \mathcal{N}(\tilde{z}_k)}{2\sigma_\star^2 \Phi(\tilde{z}_k)} \mathbf{1}_2$;
  - $\omega_k = \gamma_k \gamma_k^T - 2\beta_k$;
  - $h_k^{new} = h_k^{\setminus k} + \lambda_k^{\setminus k} \gamma_k$;
  - $\lambda_k^{new} = \lambda_k^{\setminus k} - \lambda_k^{\setminus k} \omega_k \lambda_k^{\setminus k}$;
  - $\pi_k^{new} = (\lambda_k^{new})^{-1} - (\lambda_k^{\setminus k})^{-1}$;
  - $\psi_k^{new} = (\lambda_k^{new})^{-1} \lambda_k^{\setminus k} \gamma_k + \pi_k^{new} h_k^{\setminus k}$;
  - $B_k = {h_k^{\setminus k}}^T (\lambda_k^{\setminus k})^{-1} h_k^{\setminus k} - (\psi_k^{new} + (\lambda_k^{\setminus k})^{-1} h_k^{\setminus k})^T \lambda_k^{new} (\psi_k^{new} + (\lambda_k^{\setminus k})^{-1} h_k^{\setminus k})$;
  - $s_k^{new} = \mathcal{Z}_k \left| \left((\lambda_k^{\setminus k})^{-1} + \pi_k^{new} \right) \lambda_k^{\setminus k} \right|^{\frac{1}{2}} \exp\left(\frac{1}{2} B_k\right)$; [2]
- update $\{\pi_k, \psi_k, s_k\}$, and update $\mathcal{A}$ and h as follows
  - $\mathcal{A}^{new} = \mathcal{A} + \mathbf{a}_k^T \Upsilon_k \mathbf{a}_k$, where $\Upsilon_k = \lambda_k^{-1}(\lambda_k^{new} - \lambda_k)\lambda_k^{-1}$ and $\mathbf{a}_k$ is a $2 \times n$ matrix containing the two columns of $\mathcal{A}$ associated with u_k and v_k;
  - $\boldsymbol{h}^{new} = \boldsymbol{h} + \eta \mathbf{a}_k$, where $\eta = \lambda_k^{-1} \lambda_k^{\setminus k}(\gamma_k + \pi_k h_k - \psi_k)$.

The approximate evidence at the equilibrium can be written as

$$\mathcal{P}(\mathcal{D}|\theta) \approx \frac{|\mathcal{A}|^{\frac{1}{2}}}{|\Sigma|^{\frac{1}{2}}} \exp\left(\frac{B}{2}\right) \prod_{k=1}^{K} s_k \tag{11}$$

where $B = \Psi^T \mathcal{A} \Psi$.

[2]The superfluous term $(m_k^{new})^T \pi_k^{new} m_k^{new}$ is removed from B_k.

References

Aiolli, F. and A. Sperduti. Learning preferences for multiclass problems. In *Advances in Neural Information Processing Systems 17*, 2004.

Baram, Y., R. E. Yaniv, and K. Luz. Online choice of active learning algorithms. *JMLR*, 5:255–291, 2004.

Belkin, M., P. Niyogi, and V. Sindhwani. Manifold regularization: a geometric framework for learning from examples. Computer Science Technical Report TR-2004-06, University of Chicago, 2004.

Burges, C., T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *Proceeding of the 22th International Conference on Machine Learning*, pages 89–96, 2005.

Chapelle, O., J. Weston, and B. Schölkopf. Cluster kernels for semi-supervised learning. In *NIPS*, 2003.

Chu, W. and Z. Ghahramani. Preference learning with Gaussian processes. In *Proceeding of the 22th International Conference on Machine Learning*, pages 137–144, 2005.

Cohn, D. A., Z. Ghahramani, and M. I. Jordan. Active learning with statistical models. In *Advances in Neural Information Processing Systems 7*, 1995.

Csató, L. *Gaussian Processes - Iterative Sparse Approximation*. Ph.D. thesis, Aston University, 2002.

Dekel, O., J. Keshet, and Y. Singer. Log-linear models for label ranking. In *Proceedings of the 21st International Conference on Machine Learning*, pages 209–216, 2004.

Elo, A. E. *The ratings of chess players: past and present*. London:Batsford, 1978.

Fürnkranz, J. and E. Hüllermeier. Preference learning. *Künstliche Intelligenz*, 2005. in press.

Har-Peled, S., D. Roth, and D. Zimak. Constraint classification: A new approach to multiclass classification and ranking. In *Advances in Neural Information Processing Systems 15*, 2002.

Herbrich, R., T. Graepel, P. Bollmann-Sdorra, and K. Obermayer. Learning preference relations for information retrieval. In *Proc. of Workshop Text Categorization and Machine Learning, ICML*, pages 80–84, 1998.

Lawrence, N. D. and M. I. Jordan. Semi-supervised learning via Gaussian processes. In *Advances in Neural Information Processing Systems 17*, pages 753–760, 2005.

Lawrence, N. D., M. Seeger, and R. Herbrich. Fast sparse Gaussian process methods: The informative vector machine. In Becker, S., S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, pages 609–616, 2002.

MacKay, D. J. C. Information-based objective functions for active data selection. *Neural Computation*, 4(4):589–603, 1992.

Minka, T. P. *A family of algorithms for approximate Bayesian inference*. Ph.D. thesis, Massachusetts Institute of Technology, January 2001.

Sindhwani, V., W. Chu, and S. S. Keerthi. Semi-supervised Gaussian processes. Technical report, Yahoo! Research, 2005a. in preparation.

Sindhwani, V., P. Niyogi, and M. Belkin. Beyound the point cloud: from transductive to semi-supervised learning. In *Proceedings of the 22th International Conference on Machine Learning*, pages 825–832, 2005b.

Thurstone, L. L. A law of comparative judgement. *Psychological Review*, 34:273–286, 1927.

Williams, C. K. I. and C. E. Rasmussen. Gaussian processes for regression. In Touretzky, D. S., M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 598–604, 1996. MIT Press.

Zhou, D., O. Bousquet, T. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In *Advances in Neural Information Processing Systems 18*, pages 321–328, 2004.

Zhu, X., Z. Ghahramani, and J. Lafferty. Semi-supervised learning using Gaussian fields and harmonic functions. In *Proceeding of the 20th International Conference on Machine Learning*, 2003.