Transductive Gaussian Process Regression with Automatic Model Selection

Quoc V. Le[1], Alex J. Smola[1], Thomas Gärtner[2], and Yasemin Altun[3]

[1] RSISE, Australian National University, 0200 ACT, Australia
Statistical Machine Learning Program, National ICT Australia, 0200 ACT, Australia
[2] Fraunhofer IAIS, Schloß Birlinghoven, 53754 Sankt Augustin, Germany
[3] Toyota Technological Institute at Chicago, Chicago IL 60632, USA

Abstract. In contrast to the standard *inductive inference* setting of predictive machine learning, in real world learning problems often the test instances are already available at training time. *Transductive inference* tries to improve the predictive accuracy of learning algorithms by making use of the information contained in these test instances. Although this description of transductive inference applies to predictive learning problems in general, most transductive approaches consider the case of classification only. In this paper we introduce a transductive variant of Gaussian process regression with automatic model selection, based on approximate moment matching between training and test data. Empirical results show the feasibility and competitiveness of this approach.

1 Introduction

Machine learning research mostly concentrates on estimating an underlying unknown conditional or functional dependence of a target property on some other variables. This estimate is based on a set of training instances, respecting this dependency. It is then usually applied to test instances for which the target property has not been observed. This setting is known as *supervised learning* or *inductive inference*. The downside of such algorithms is that they ignore the test data at training time even when such data is available. In this case, *transductive inference* approaches promise improved predictive accuracy as they exploit available knowledge about the test instances at training time. A related class of methods are *semi-supervised learning* algorithms that take advantage of additional unlabeled data which may or may not be used for testing purposes. See, e.g., [1] for an overview of recent results.

This work has led to a number of competitive algorithms mostly making use of the "cluster assumption", i.e., that "the decision boundary should not cross high-density regions", e.g., [2]. Although the transductive as well as the semi-supervised learning settings have no inherent restriction to classification only, there is so far very little work on transductive nor semi-supervised regression or structural prediction. Most work on transductive or semi-supervised regression is primarily concerned with designing kernel matrices such as the inverse graph Laplacian [3,4] or related matrices [5,6] on both labeled and unlabeled data. Similarly, Bayesian Committee Machines [7] can also be considered as a transductive method where the test data is incorporated in the computation of the kernel [8]. In [9] the labels for the test data are chosen to minimise the

J. Fürnkranz, T. Scheffer, and M. Spiliopoulou (Eds.): ECML 2006, LNAI 4212, pp. 306–317, 2006.

leave-one-out error of ridge regression on the joint training and test data and are constraint to be close to the inductive solution. In [10] the disagreement on unlablelled data between the hypotheses and an origin function is minimised and in [11] the disagreement on unlablelled data between hypotheses from different views is minimised.

In this paper we introduce a transductive algorithm for Gaussian process regression. The algorithm is based on the idea that the moments on training and test set, i.e., mean and variance, should match approximately. This is a realistic assumption in many real-world datasets and theoretically justified by the assumption of iid data and the observation that scalar quantities are much more concentrated around their mean than, say, the distribution of maximum a posteriori estimators. More precisely, let $\{y_1, \ldots, y_m\}$ denote the labels on the training data. We make sure that the observed mean and variance on the training set match the predicted mean and variance on the test set $\mathbf{E}[y]$ or $\mathbf{E}\left[y^2\right]$. In the present paper we achieve this by directly modifying the prior such that only parameters which are consistent between training and test set are considered in the inference procedure. The algorithm has the potential of being combined with previous approaches based on modifying the kernel or on minimising the disagreement between hypotheses.

In this fashion our setting draws on [12] which studies transductive classification based on a similar principle, namely that the predicted conditional class probabilities should match the observed counts on the training set. While we frame our approach in terms of a homoscedastic Gaussian Process estimator [13] it is readily extensible to heteroscedastic estimation [14], albeit at the expense of additional complication in the notation.

This paper is organized as follows: Section 2 introduces Gaussian process regression and model selection strategies for Gaussian processes. Our general approach to transductive Gaussian processes with automatic model selection is described in Section 3 and the optimization details are laid out in Section 4. Finally, Section 5 contains our empirical findings and Section 6 concludes.

2 Gaussian Process Regression

2.1 Setting

We begin with a very brief overview over Gaussian Process (GP) Regression, as described, e.g., in [15,13]. Denote by $\mathcal{X} \times \mathcal{Y}$ the domain of patterns and labels respectively from which m pairs (x_i, y_i) are drawn independently and identically distributed (iid). For regression assume that $\mathcal{Y} \subseteq \mathbb{R}$. Moreover assume that there exists a Gaussian process on $\mathcal{X}$ with covariance kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and mean $\mu : \mathcal{X} \to \mathbb{R}$. For notational convenience we set $\mu(x) = 0$, i.e., we ignore the offset for the remainder of the paper.

The key assumption of GP regression is that y is given by $y = t + \xi$ where $\xi \sim \mathcal{N}(0, \sigma^2)$ is an iid Gaussian random variable and that t is drawn from the Gaussian process on $\mathcal{X}$ specified by k. That is,

$$Y = (y_1, \ldots, y_m) \sim \mathcal{N}(0, K + \sigma^2 \mathbf{I})$$

where $K_{ij} = k(x_i, x_j)$ and $\mathbf{I}$ is the identity matrix.

2.2 Regression

Since $Y|X \sim \mathcal{N}(0, K + \sigma^2\mathbf{I})$ is normal, so is the conditional distribution of test labels given training and test data $p(Y_{\text{test}}|Y_{\text{train}}, X_{\text{train}}, X_{\text{test}})$. We have $Y_{\text{test}}|Y_{\text{train}}, X_{\text{train}}, X_{\text{test}} \sim \mathcal{N}(\mu, \Sigma)$ where

$$\mu = K_{\text{test,train}}(K_{\text{train,train}} + \sigma^2\mathbf{I})^{-1}Y_{\text{train}} \tag{1}$$

$$\Sigma = K_{\text{test,test}} + \sigma^2\mathbf{I} - K_{\text{test,train}}(K_{\text{train,train}} + \sigma^2\mathbf{I})^{-1}K_{\text{train,test}} \,. \tag{2}$$

Here $K_{\text{train,train}}$ is the covariance matrix computed on the training set $X_{\text{train}} = \{x_1, \ldots, x_m\}$, $K_{\text{test,test}}$ is the corresponding part computed on $X_{\text{test}} = \{x'_1, \ldots, x'_{m'}\}$, $K_{\text{test,train}}, K_{\text{train,test}}$ contain the cross-terms, and Y_{train} is the vector of training labels y_i. Eq. (2) contains the Schur complement arising from conditioning on a subset of random variables.

Note that the distribution of $Y_{\text{test}}|Y_{\text{train}}, X_{\text{train}}, X_{\text{test}}$ may differ significantly from the distribution of observed Y_{train}. In particular, there is no guarantee that any of the moments of the conditional distribution match that of the observed data. This is the key weakness of the model which we will address in Section 3.

2.3 Model Selection

If we knew the correct k and σ^2, Equations (1) and (2) would be all we need for inference. In reality, the kernel k and the degree of noise σ^2 need to be adjusted. By Bayes rule this leads to

$$p(Y_{\text{train}}, \sigma^2, k|X_{\text{train}}) \propto\ p(Y_{\text{train}}|X_{\text{train}}, \sigma^2, k)p(\sigma^2, k|X_{\text{train}}) \,.$$

Inference is then carried out either by sampling from the posterior or by maximum a posteriori (MAP) estimation with respect to (k, σ^2). For the purpose of this paper we focus on the latter due to its superior computational efficiency. Lacking further knowledge about the prior, one typically assumes that $p(\sigma^2, k|X_{\text{train}}) = p(\sigma^2)p(k)$ factorizes. Typically $p(k)$ is non-zero only for a parameterised family of kernels. For instance, the liberty in choosing k might relate to the width and scaling in a Gaussian RBF kernel, leading to parameter scaling in a fashion similar to Automatic Relevance Determination [16]. We then need the derivatives with respect to these parameters (see Section 4.4).

This leads to the minimization of the negative log-posterior $\mathcal{P}(\sigma^2, k) :=$
$-\log p(Y_{\text{train}}, \sigma^2, k|X_{\text{train}})$ which is given (up to constants) by

$$\mathcal{P}(\sigma^2, k) = \frac{1}{2}\log|K + \sigma^2\mathbf{I}| - \log p(\sigma^2) - \log p(k) + \frac{1}{2}y^\top(K + \sigma^2\mathbf{I})^{-1}y \,. \tag{3}$$

Here we skipped the "train" subscripts on K and Y for a more compact notation. Using tr to denote the trace, the derivatives of $\mathcal{P}$ with respect to σ^2 and k are readily obtained via:

$$\begin{aligned}
\partial_{\sigma^2}\mathcal{P} =& \frac{1}{2}\operatorname{tr}(K + \sigma^2\mathbf{I})^{-1} - \partial_{\sigma^2}\log p(\sigma^2) - \frac{1}{2}\left\|(K + \sigma^2)^{-1}y\right\|^2 \\
\partial_k\mathcal{P} =& \frac{1}{2}\operatorname{tr}(K + \sigma^2\mathbf{I})^{-1}\left[\partial_k K\right] - \partial_k\log p(k) \\
& - \frac{1}{2}y^\top(K + \sigma^2\mathbf{I})^{-1}\left[\partial_k K\right](K + \sigma^2\mathbf{I})^{-1}y \,.
\end{aligned} \tag{4}$$

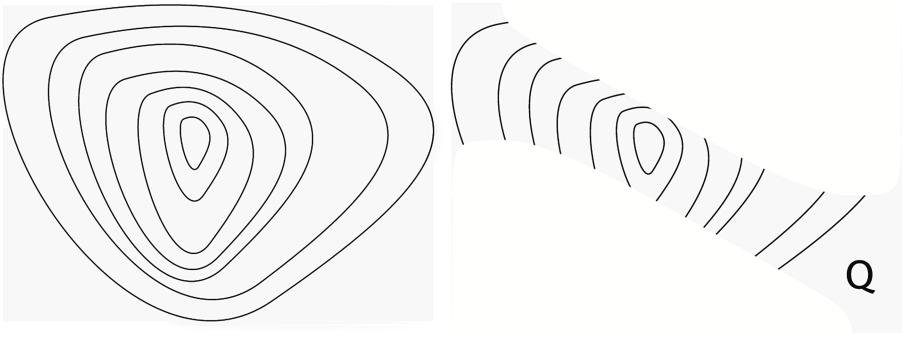

Fig. 1. Left: unrestricted prior, e.g., $p(\sigma^2, k)$, with contour lines indicating equal prior probability; Right: effective prior as restricted by $\mathcal{Q}_\epsilon$ to a sub-domain of (σ^2, k) which satisfy the marginal constraints on the test set. The order of hypotheses within the feasible set remains unchanged by the intersection with $\mathcal{Q}_\epsilon$. However, the normalization changes due to the restriction of the domain.

Minimization is achieved, e.g., by gradient descent. In terms of computation, the key cost is to deal with the inverse of $(K + \sigma^2 \mathbf{I})$.

3 Transduction and Empirical Bayes

When viewing the negative log-posterior (3) it is obvious that X_{test} does not enter the discussion. This is perfectly reasonable provided that our prior on k and σ^2 is well specified. In reality, however, we can rarely be sure that the prior is sufficiently accurate. We address this issue in the following by a semi-empirical construction of the prior on σ^2, k.

3.1 Restricting the Prior

We begin with a prior $p(k, \sigma^2)$ which denotes our (so far observation independent) knowledge of the estimation problem. We would like to modify the prior such that it only contains values of k and σ^2 such that $Y_{\text{test}}|Y_{\text{train}}, X$ has a distribution similar to that of Y_{train}. In particular, we consider distributions from the family $\mathcal{Q}_\epsilon$ which on the test set Y_{test} has mean and variance close to the observed values on the training set:

$$\mathcal{Q}_\epsilon := \{q| \, \|\mathbf{E}_{Y_{\text{test}} \sim q} [\phi(y)] - \bar{\mu}\| \leq \epsilon\} \ . \tag{5}$$

Here $\phi(y) := \left(y, -\frac{1}{2}y^2\right)$ are the sufficient statistics of the normal distribution and $\bar{\mu} = m^{-1} \sum_{i=1}^{m} \phi(y_i)$ is the empirical statistics of y on the training set Y_{train}.

We could now simply perform inference by minimizing $\mathcal{P}(\sigma^2, k)$ subject to the constraint that $p(Y_{\text{test}}|Y_{\text{train}}, X, \sigma^2, k) \in \mathcal{Q}_\epsilon$ (See Figure 1 for an example). However, there is no guarantee that any (σ^2, k) satisfies the constraint on the distribution. Hence we relax the conditions in the following sections to also include distributions close to $\mathcal{Q}_\epsilon$.

3.2 Unconstrained Minimization

Denote by $D(p\|q)$ the Kullback-Leibler (KL) divergence between two distributions

$$D(q\|p) := \int \log \frac{q(x)}{p(x)} dq(x)$$

and denote by $D(\mathcal{Q}\|p) := \inf_{q\in\mathcal{Q}} D(q\|p)$ the KL-divergence between a distribution p and a subset of distributions $\mathcal{Q}$. Since the KL divergence vanishes only for equivalent distributions, $D(\mathcal{Q}\|p) = 0$ is equivalent to $p \in \mathcal{Q}$. Moreover $D(\mathcal{Q}\|p) \geq 0$ for all p.

This provides us with a *barrier function* to ensure that $p \in \mathcal{Q}$ whenever possible and a measure for the distance between $\mathcal{Q}$ and some $p \notin \mathcal{Q}$. Instead of minimizing $\mathcal{P}(\sigma^2, k)$ we modify the negative log-likelihood and minimize now:

$$\mathcal{P}(\sigma^2, k) + \lambda D(\mathcal{Q}\|p(Y_{\text{test}}|Y_{\text{train}}, X, \sigma^2, k)) \,, \tag{6}$$

where $\lambda \geq 0$. For $\lambda \to \infty$ we obtain the optimization problem with hard constraints on (σ^2, k). For $\lambda \to 0$ we recover the unrestricted problem.

Similar to variational methods the objective function can be rewritten in terms of the entropy of the closest distribution in $\mathcal{Q}$ and an effective likelihood term in p. The problem of minimising (6) can then be rewritten as a joint minimization over (σ^2, k, q) as

$$\inf_{q\in\mathcal{Q},\sigma^2,k} \mathcal{P}(\sigma^2, k) + \lambda D(q(Y_{\text{test}})\|p(Y_{\text{test}}|Y_{\text{train}}, X, \sigma^2, k)) \,.$$

Decomposing the KL-divergence we have

$$\begin{aligned} \inf_{q\in\mathcal{Q},\sigma^2,k} & -\log p(Y_{\text{train}}|X_{\text{train}}, \sigma^2, k) - \log p(\sigma^2) - \log p(k) \\ & - \lambda \mathbf{E}_{Y_{\text{test}}\sim q}\left[\log p(Y_{\text{test}}|Y_{\text{train}}, X, \sigma^2, k)\right] - \lambda H(q) \end{aligned} \tag{7}$$

where H denotes the entropy ($H(q) = -\int \log q(x) dq(x)$).

This decomposition closely resembles variational methods for estimation, where an intractable model is replaced by a tractable approximation, see e.g., [17].

The joint minimization problem over q and (σ^2, k) can be solved, e.g., by subspace descent. The advantage of this approach is that while the objective function (7) is jointly *nonconvex* in the parameters, the resulting subproblems may be more amenable to minimization. For instance, the problem of finding a minimizer in q for fixed (σ^2, k) can be recast as a convex problem for certain $\mathcal{Q}$. We have the following algorithm:

1. For fixed q minimize

$$-\log p(Y_{\text{train}}, \sigma^2, k|X_{\text{train}}) - \lambda \mathbf{E}_{Y_{\text{test}}\sim q}\left[\log p(Y_{\text{test}}|Y_{\text{train}}, X, \sigma^2, k)\right] \tag{8}$$

   with respect to (σ^2, k).
2. For fixed (σ^2, k) minimize

$$D(q(Y_{\text{test}})\|p(Y_{\text{test}}|Y_{\text{train}}, X, \sigma^2, k))$$

   with respect to q, where $q \in \mathcal{Q}$.

In the following section we discuss both steps in greater detail for the case of regression. We begin with Step 2.

4 Minimizing the Effective Posterior

4.1 A Duality Theorem for q

Recall the definition of $\mathcal{Q}_\epsilon$ as in (5). There we required that q evaluated on Y_{test} has approximate mean $\bar{\mu}$ with regard to the statistic $\phi(y)$. The following theorem, which follows immediately from [18] is a generalization of the well-known duality between maximum likelihood estimation and entropy maximization with moment matching constraints. It states the connection between maximum a posteriori estimation and entropy maximization with *approximate moment matching constraints*,

Theorem 1 (Approximate KL Minimization). *Denote by $\mathcal{X}$ a domain and let p, q be distributions on $\mathcal{X}$. Moreover, let $\phi(x) : \mathcal{X} \to \mathcal{B}$ be a map from $\mathcal{X}$ to a Banach space $\mathcal{B}$. Then for any $\epsilon \geq 0$ the problem*

$$\min_q \; D(q\|p) \text{ subject to } \|\mathbf{E}_{x\sim q}\left[\phi(x)\right] - \bar{\mu}\| \leq \epsilon$$

has the solution

$$q_\theta(x) = p(x)\exp\left(\langle\phi(x), \theta\rangle - g(\theta)\right)$$

where θ is an element of the dual space of $\mathcal{B}$. Here $g(\theta)$ ensures that $q(x)$ is normalized to 1. Moreover θ is found as solution of the maximum a posteriori estimation problem

$$\min_\theta \; g(\theta) - \langle\bar{\mu}, \theta\rangle + \epsilon\|\theta\| \; . \tag{9}$$

Equivalently for every feasible ϵ there exists some $\Lambda \geq 0$ such that the minimum of $g(\theta) - \langle\bar{\mu}, \theta\rangle + \frac{\Lambda}{2}\|\theta\|^2$ minimizes (9).

The quadratic formulation in $\|\theta\|^2$ is preferable in terms of optimization as it is always feasible. In terms of the transductive regression estimation problem this means that

$$q(Y_{\text{test}}) = p(Y_{\text{test}}|Y_{\text{train}}, X, \sigma^2, k)\exp\left(\langle\phi(Y_{\text{test}}), \theta\rangle - g(\theta)\right)$$

where $\phi(Y_{\text{test}}) = \frac{1}{m'}\sum_{i=1}^{m'}\left(y_i', \frac{1}{2}{y_i'}^2\right)$ for m' test instances. Since p is a normal distribution and $\phi(Y_{\text{test}})$ only contains linear and quadratic terms in Y_{test}, the overall distribution $q(Y_{\text{test}})$ will also be normal. This greatly simplifies the calculation of $g(\theta)$ and its derivatives:

$$\partial_\theta g(\theta) = \mathbf{E}\left[\phi(Y_{\text{test}})\right] \text{ and } \partial_\theta^2 g(\theta) = \text{Cov}\left[\phi(Y_{\text{test}})\right] \; .$$

4.2 Minimizing with Respect to q

Let $\mathbf{1}$ denote the all one vector and $\mathbf{I}$ the identity matrix. The linear and quadratic terms in $-\log q(Y_{\text{test}})$, as a function of λ and the mean and variance (μ and Σ) of $p(Y_{\text{test}}|Y_{\text{train}}, X, \sigma^2, k)$ are then given by

$$\frac{1}{2}(Y_{\text{test}} - \mu)^\top\Sigma^{-1}(Y_{\text{test}} - \mu) - \theta_1\mathbf{1}^\top Y_{\text{test}} + \frac{1}{2}\theta_2\|Y_{\text{test}}\|^2$$

which corresponds to a normal distribution with variance and mean

$$\Sigma_q^{-1} = \Sigma^{-1} + \theta_2 \mathbf{I}$$
$$\mu_q = (\Sigma^{-1} + \theta_2 \mathbf{I})^{-1}(\Sigma^{-1}\mu + \theta_1 \mathbf{1}) .$$

The latter can be seen by some tedious but very straightforward algebra matching up linear and quadratic terms in the expansion in Y_{test}. It also allows us to compute the expected value of $\phi(Y_{\text{test}})$ as follows:

$$\begin{aligned}
\mathbf{E}\left[\phi_1(Y_{\text{test}})\right] =& \frac{1}{m'}\mathbf{1}^\top \mu_q = \frac{1}{m'}\mathbf{1}^\top(\Sigma^{-1} + \theta_2\mathbf{I})^{-1}(\Sigma^{-1}\mu + \theta_1\mathbf{1}) \\
\mathbf{E}\left[\phi_2(Y_{\text{test}})\right] =& \frac{1}{m'}\left[\operatorname{tr}\Sigma_q + \|\mu_q\|^2\right] \\
=& \frac{1}{m'}\operatorname{tr}\left(\Sigma^{-1} + \theta_2\mathbf{I}\right)^{-1} + \frac{1}{m'}\left\|(\Sigma^{-1} + \theta_2\mathbf{I})^{-1}(\Sigma^{-1}\mu + \theta_1\mathbf{1})\right\|^2 .
\end{aligned}$$

Putting everything together we obtain the conditions for finding the optimal value of q in transductive regression:

$$\partial_\theta\left[-\log q(\bar{\mu}) + \tfrac{\Lambda}{2}\|\theta\|^2\right] = 0 \iff \mathbf{E}\left[\phi(Y_{\text{test}})\right] - \bar{\mu} + \Lambda\theta = 0 . \tag{10}$$

Moreover, the solution is unique and the problem can be solved by the Newton method or conjugate gradient descent as the Jacobian of the LHS of (10) is positive definite.

4.3 Minimizing with Respect to p

We now describe how to perform the optimization in Step 1. With regard to the minimization in p we already accomplished a significant part of the calculations in (4). What remains is to deal with the expected log-likelihood of $p(Y_{\text{test}}|Y_{\text{train}}, X, \sigma^2, k)$ with respect to q. We use the following simple lemma:

Lemma 1. *Let $\Sigma, \Sigma_q \succeq 0$ be covariance matrices in $\mathbb{R}^{n\times n}$ and let $\mu, \mu_q \in \mathbb{R}^n$ be corresponding means. In this case*

$$\mathbf{E}_{x\sim\mathcal{N}(\mu_q,\Sigma_q)}\left[(x-\mu)^\top\Sigma^{-1}(x-\mu)\right] = \operatorname{tr}\Sigma^{-1}\Sigma_q + (\mu_q-\mu)^\top\Sigma^{-1}(\mu_q-\mu) .$$

Proof (Sketch only). By the trace formula $\mathbf{E}\left[x^\top\Sigma^{-1}x\right] = \operatorname{tr}\mathbf{E}\left[xx^\top\right]\Sigma^{-1}$. Expanding $(x-\mu) = (x-\mu_q) + (\mu_q-\mu)$ and direct calculation yields the desired result.

Consequently we can expand the expected log-likelihood (up to constants) as

$$\begin{aligned}
\mathcal{T}(\sigma^2, k, q) :=& -\mathbf{E}_{Y_{\text{test}}\sim\mathcal{N}(\mu_q,\Sigma_q)}\log p(Y_{\text{test}}|Y_{\text{train}}, X, k, \sigma^2) \\
=& \frac{1}{2}\log|\Sigma| + \frac{1}{2}\operatorname{tr}\Sigma^{-1}\Sigma_q + \frac{1}{2}(\mu_q-\mu)^\top\Sigma^{-1}(\mu_q-\mu)
\end{aligned}$$

where μ, Σ are given by (1) and (2) respectively. The last step is to take derivatives with respect to those parameters in analogy to $\mathcal{P}$. By standard matrix calculus [19] we obtain

$$\begin{aligned}
\partial_k\mathcal{T}(\sigma^2, k, q) =& \frac{1}{2}\operatorname{tr}\Sigma^{-1}\left[\partial_k\Sigma\right] - \frac{1}{2}\operatorname{tr}\Sigma^{-1}\left[\partial_k\Sigma\right]\Sigma^{-1}\Sigma_q \\
&+ (\mu-\mu_q)^\top\Sigma^{-1}\left[\partial_k\mu\right] - \frac{1}{2}(\mu_q-\mu)^\top\Sigma^{-1}\left[\partial_k\Sigma\right]\Sigma^{-1}(\mu_q-\mu) .
\end{aligned}$$

The terms arising from $\partial_{\sigma^2}\mathcal{T}$ are analogous. Finally, the derivatives of Σ and μ with respect to k and σ^2 are given by

$$\begin{aligned}
\partial_{\sigma^2}\mu =& -K_{\text{test,train}}(K_{\text{train,train}}+\sigma^2\mathbf{I})^{-2}Y_{\text{train}}\\
\partial_k\mu =& -K_{\text{test,train}}(K_{\text{train,train}}+\sigma^2\mathbf{I})^{-1}\partial_k K_{\text{train,train}}(K_{\text{train,train}}+\sigma^2\mathbf{I})^{-1}Y_{\text{train}}\\
\partial_{\sigma^2}\Sigma =& \mathbf{I}-K_{\text{test,train}}(K_{\text{train,train}}+\sigma^2\mathbf{I})^{-2}K_{\text{train,test}}\\
\partial_k\Sigma =& \partial_k K_{\text{test,test}}-\partial_k K_{\text{test,train}}(K_{\text{train,train}}+\sigma^2\mathbf{I})^{-1}K_{\text{train,test}}\\
&-K_{\text{test,train}}(K_{\text{train,train}}+\sigma^2\mathbf{I})^{-1}\partial_k K_{\text{train,test}}\\
&+K_{\text{test,train}}(K_{\text{train,train}}+\sigma^2\mathbf{I})^{-1}\\
&+\partial_k K_{\text{train,train}}(K_{\text{train,train}}+\sigma^2\mathbf{I})^{-1}K_{\text{train,test}}\,.
\end{aligned}$$

Finally, the derivatives of the restricted log-posterior given in (8) are given by summing over the terms $\mathcal{P}(\sigma^2,k)+\lambda\mathcal{T}(\sigma^2,k,q)$. Standard optimization methods for choosing adequate parameters in k and σ^2 can subsequently be applied to the problem.

4.4 Application to Automatic Relevance Determination

ARD [16] is a means of determining the scale of random variables. This gives us a principled method for choosing the appropriate parameters k and σ^2. In the context of Gaussian processes, we can parameterize the kernel k by

$$k_\Theta(x,x') := k(\Theta x,\Theta x')$$

where Θ is a diagonal matrix which ensures proper scaling of x in different coordinates. For Gaussian RBF kernels, we have $k(x,x')=\exp\left(-\|\Theta(x-x')\|^2\right)$, whose derivative is given by

$$\partial_\Theta k_\Theta(x,x') = -2((x_1-x_1')^2\Theta_1,\ldots,(x_n-x_n')^2\Theta_n)^\top\exp\left(-\|\Theta(x-x')\|^2\right)\,.$$

A suitable choice of a prior on the coefficients $\Theta_i\in\mathbb{R}$ can ensure that many of them will vanish. In particular we choose a factorizing gamma prior, for which

$$-\log p(\Theta)=\sum_{i=1}^{n}-a\log\Theta_i+b\Theta_i+\text{const}\,.$$

Similarly we choose a gamma prior for the additive noise σ^2.

5 Experimental Results

5.1 Regression Datasets

Dataset Facts. For experimental evaluation we decided to use the same datasets and preprocesing as in [20]. There, 23 regression datasets from UCI [21] and the R [22]

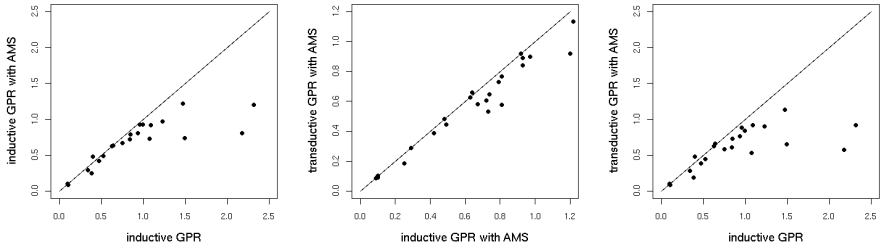

Fig. 2. Mean root mean squared errors of the different approaches on all used datasets. Note the different scaling of the figures

packages `mlbench, quantreg, alr3` and `MASS` were picked[1]. No datasets with missing values were used. In some cases where the target variable was not obvious, it was selected arbitrarily. The sample sizes vary from $m = 43$ to $m = 1375$ and the lengths of input vectors vary from $n = 1$ to $n = 60$. Finally, some datasets were standardized to have zero mean and unit variance (the datasets were also used in this form in [20]).

Overview of Results. We compared transductive and inductive GP regression in 10-fold cross-validations. For inductive GP regression the kernel bandwidth and the additive noise level are chosen via cross validation within the training sample as this is the common practice in many other papers. For transductive GP regression Λ and λ are chosen via cross validation within the training sample. The automatic relevance determination parameters are held constant throughout all experiments ($a = 1, b = 0.5$). To compare the two models we used the root mean squared error over 10-fold cross validations.

The results are illustrated in Figure 2 and full details are given in Table 1. The last three columns are the mean $\pm$ standard deviation of the root mean square errors. In the large majority of the test cases, the transductive GP regression outperforms the inductive GP regression in terms of root mean square error (20 wins/ 3 losses). However, not in all cases the difference is significant.

Statistical Comparison. To verify that transductive Gaussian processes with automatic model selection (AMS) significantly outperform inductive Gaussian processes (with AMS) over all datasets, we need to perform a proper statistical test with the null hypothesis that the algorithms perform equally well. As suggested recently [23] we used the Wilcoxon signed ranks test.

The Wilcoxon signed ranks test is a nonparametric test to detect shifts in populations given a number of paired samples. The underlying idea is that under the null hypothesis the distribution of differences between the two populations is symmetric about 0. It proceeds as follows: (i) compute the differences between the pairs, (ii) determine the ranking of the absolute differences, and (iii) sum over all ranks with positive and negative difference to obtain W_+ and W_-, respectively. The null hypothesis can be rejected

[1] Descriptions are available at http://cran.r-project.org/src/contrib/PACKAGES.html

Table 1. Dataset facts (number of instances, number of attributes, class attribute, dropped attributes, standardized (Yes, No)) and regression results (root mean squared error of inductive, inductive Gaussian processes with AMS, and transductive Gaussian processes with AMS, respectively). Bold numbers denote smaller error.

Data set	#Inst	#Att	Class	Dropped	Std	Inductive	Ind. (AMS)	Transd (AMS)
diabetes	43	3	c_peptide	-	N	$\mathbf{0.64 \pm 0.66}$	$\mathbf{0.64 \pm 0.66}$	0.66 ± 0.45
triazines	186	61	activity	-	N	0.10 ± 0.12	0.10 ± 0.10	$\mathbf{0.09 \pm 0.17}$
pyrimidines	74	28	activity	-	N	0.10 ± 0.09	$\mathbf{0.09 \pm 0.05}$	$\mathbf{0.09 \pm 0.05}$
BigMac2003	69	10	BigMac	City	Y	1.08 ± 0.95	0.73 ± 0.85	$\mathbf{0.53 \pm 0.55}$
UN3	125	7	Purban	Locality	Y	0.84 ± 0.44	0.72 ± 0.42	$\mathbf{0.61 \pm 0.38}$
topo	52	3	z	-	Y	1.49 ± 2.75	0.74 ± 1.05	$\mathbf{0.65 \pm 0.40}$
mcycle	133	2	accel	-	Y	1.23 ± 0.38	0.97 ± 0.20	$\mathbf{0.9 \pm 0.25}$
CobarOre	38	3	z	-	Y	1.47 ± 1.32	1.22 ± 0.92	$\mathbf{1.14 \pm 0.61}$
highway	39	12	Rate	-	Y	1.00 ± 0.84	0.93 ± 0.68	$\mathbf{0.84 \pm 0.66}$
sniffer	125	5	Y	-	Y	0.75 ± 0.37	0.67 ± 0.33	$\mathbf{0.58 \pm 0.31}$
caution	100	3	y	-	Y	1.03 ± 0.86	0.92 ± 0.54	$\mathbf{0.91 \pm 0.55}$
gilgais	365	9	e80	-	Y	0.85 ± 0.62	0.79 ± 0.6	$\mathbf{0.73 \pm 0.55}$
ftcollinssnow	93	2	Late	YR1	Y	2.31 ± 3.45	1.20 ± 0.95	$\mathbf{0.92 \pm 0.51}$
crabs	200	7	CW	index	Y	0.34 ± 0.26	0.29 ± 0.21	$\mathbf{0.29 \pm 0.21}$
BostonHousing	506	14	medv	-	Y	0.47 ± 0.35	0.42 ± 0.29	$\mathbf{0.39 \pm 0.27}$
engel	235	2	y	-	Y	2.18 ± 4.05	0.81 ± 0.75	$\mathbf{0.58 \pm 0.50}$
heights	1375	2	Dheight	-	Y	$\mathbf{0.10 \pm 0.10}$	$\mathbf{0.10 \pm 0.10}$	$\mathbf{0.10 \pm 0.09}$
snowgeese	45	5	photo	-	Y	0.53 ± 0.44	0.49 ± 0.43	$\mathbf{0.44 \pm 0.49}$
ufc	372	5	Height	-	Y	$\mathbf{0.63 \pm 0.39}$	$\mathbf{0.63 \pm 0.31}$	$\mathbf{0.63 \pm 0.31}$
birthwt	189	8	bwt	ftv, low	Y	0.38 ± 0.55	0.25 ± 0.51	$\mathbf{0.19 \pm 0.22}$
GAGurine	314	2	GAG	-	Y	0.94 ± 0.72	0.81 ± 0.79	$\mathbf{0.77 \pm 0.82}$
geyser	299	2	waiting	-	Y	0.96 ± 0.61	0.93 ± 0.65	$\mathbf{0.89 \pm 0.48}$
cpus	209	8	estperf	name	Y	$\mathbf{0.40 \pm 0.46}$	0.48 ± 0.78	0.48 ± 0.78

if W_+ (or $\min(W_+, W_-)$, respectively) is located in the tail of the null distribution which has sufficiently small probability.

The critical value of the one-sided Wilcoxon signed ranks test for 23 samples on a 0.5% significance level is 55. On this significance level we can reject the null hypotheses.

6 Outlook and Future Work

We presented a new transductive GP regression method, where the prior distribution on model selection parameters is modified for approximate moment matching between training and test set. Experimental results show the competitiveness of our approach. Note that significant improvements were achieved in cases where the size of the unlabelled data is only $1/10$-th of the training data. We expect even larger improvements over inductive GP when more unlabelled data is used. We also would like to emphasize that this method is in fact orthogonal to other transductive methods, eg. one can use a semi-supervised kernel function as well as the moment matching constraints.

It is important to note the generality of this method. The approximate moment matching constraints have been applied to classification problems and can easily be extend to

structural learning: All we need to do is impose the constraints that the expectations of singleton labels as well as the expectations of the neighboring label clusters over the test set should approximately match the statistics of the training data. That is, we impose moment matching conditions on the class marginals. Note that if we have a large amount of data at our disposition, imposing only moment constraints may be wasteful. That is, we have information not only about the class marginals globally but also *locally*. This leads to an interesting crossover of inductive and transductive estimation, which is subject of current research.

Finally note the similarity of our setup to empirical Bayes estimation insofar as we adjust the prior over the hypothesis space *after* seeing the data. While this clearly runs counter to proper Bayesian procedure, it still produces convincingly better results. It would be interesting to see whether it is possible to obtain statistical confidence bounds for our estimator: From [18] it follows immediately that the expected log-likelihood is well concentrated. This, however, is not our aim — we would like to obtain bounds that are *better* than the conventional uniform convergence bounds taking advantage of the fact that we have additional test data.

The transductive Gaussian process regression with automatic relevance determination can help us to determine which feature is important in regression. This information can be useful in many fields, for example in bio-informatics, where the knowledge of which genes play important roles is valuable.

Acknowledgements

The authors wish to thank the reviewers for valuable comments. The National ICT Australia is partially funded through the Australian Government's *Baking Australia's Ability* initiative and the Australian Research Council. Part of this work is also funded by the German Science Foundation DFG under grant WR40/2-2.

References

1. Zhu, X.: Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison (2005) http://www.cs.wisc.edu/∼jerryzhu/pub/ssl_survey.pdf.
2. Chapelle, O., Zien, A.: Semi-supervised classification by low density separation. In: Tenth International Workshop on Artificial Intelligence and Statistics. (2005)
3. Zhu, X., Lafferty, J., Ghahramani, Z.: Semi-supervised learning using gaussian fields and harmonic functions. In: Proc. Intl. Conf. Machine Learning. (2003)
4. Smola, A.J., Kondor, I.R.: Kernels and regularization on graphs. In Schölkopf, B., Warmuth, M.K., eds.: Proc. Annual Conf. Computational Learning Theory. Lecture Notes in Computer Science, Springer (2003) 144–158
5. Pozdnoukhov, A., Bengio, S.: Semi-supervised kernel methods for regression estimation. In: IEEE International Conference on Acoustic, Speech, and Signal Processing. (2006)
6. Verbeek, J., Vlassis, N.: Gaussian fields for semi-supervised regression and correspondence learning. Pattern Recognition, special issue on similarity based pattern recognition (2006)
7. Tresp, V.: A Bayesian committee machine. Neural Computation **12**(11) (2000) 2719–2741

8. Schwaighofer, A., Tresp, V.: Transductive and inductive methods for approximate guassian process regression. In: Neural Information Processing Systems, MIT Press (2003)
9. Chapelle, O., Vapnik, V., Weston, J.: Transductive inference for estimating values of functions. In: Advances in Neural Information Processing Systems. (1999)
10. Schuurmans, D., Southey, F., Wilkinson, D., Guo, Y.: Metric-based approaches for semi-supervised regression and classification. In: Semi-Supervised Learning. MIT Press (2006)
11. Brefeld, U., Gärtner, T., Scheffer, T., Wrobel, S.: Efficient co-regularised least squares regression. In: 23rd International Conference on Machine Learning. (2006)
12. Gärtner, T., Le, Q., Burton, S., Smola, A.J., Vishwanathan, S.V.N.: Large-scale multiclass transduction. In Weiss, Y., Schölkopf, B., Platt, J., eds.: Advances in Neural Information Processing Systems 18, Cambride, MA, MIT Press (2006) 411 – 418
13. Neal, R.M.: Monte Carlo implementation of Gaussian process models for Bayesian regression and classification. Technical Report Technical Report 9702, Dept. of Statistics (1997)
14. Le, Q.V., Smola, A.J., Canu, S.: Heteroscedastic gaussian process regression. In: Proc. Intl. Conf. Machine Learning. (2005)
15. Williams, C.K.I.: Prediction with Gaussian processes: From linear regression to linear prediction and beyond. In Jordan, M.I., ed.: Learning and Inference in Graphical Models. Kluwer Academic (1998) 599–621
16. Neal, R.M.: Assessing relevance determination methods using delve. In: Neural Networks and Machine Learning, Springer (1998) 97–129
17. Jordan, M.I., Gharamani, Z., Jaakkola, T.S., Saul, L.K.: An introduction to variational methods for graphical models. In Jordan, M.I., ed.: Learning in Graphical Models. Kluwer Academic (1998) 105–162
18. Altun, Y., Smola, A.: Divergence minimization and convex duality. In: Proc. Annual Conf. Computational Learning Theory. (2006) to appear.
19. Lütkepohl, H.: Handbook of Matrices. John Wiley and Sons, Chichester (1996)
20. Takeuchi, I., Le, Q., Sears, T., Smola, A.: Nonparametric quantile estimation. Journal of Machine Learning Research (2006) To appear and available at http://sml.nicta.com.au/∼quocle.
21. Newman, D., Hettich, S., Blake, C., Merz, C.: UCI repository of machine learning databases (1998)
22. R Development Core Team: R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. (2005) ISBN 3-900051-07-0.
23. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research **7(1)** (2006)