Heteroscedastic Gaussian Process Regression

Quoc V. Le QUOC.LE@ANU.EDU.AU
Alex J. Smola ALEX.SMOLA@NICTA.COM.AU
RSISE, Australian National University, 0200 ACT, Australia
Statistical Machine Learning Program, National ICT Australia, 0200 ACT, Australia

Stéphane Canu STEPHANE.CANU@INSA-ROUEN.FR
PSI - FRE CNRS 2645, INSA de Rouen, France

Abstract

This paper presents an algorithm to estimate simultaneously both mean and variance of a non parametric regression problem. The key point is that we are able to estimate variance *locally* unlike standard Gaussian Process regression or SVMs. This means that our estimator adapts to the local noise. The problem is cast in the setting of maximum a posteriori estimation in exponential families. Unlike previous work, we obtain a convex optimization problem which can be solved via Newton's method.

1. Introduction

Regression estimation aims at finding labels $y \in \mathcal{Y}$ (where $\mathcal{Y} = \mathbb{R}$ or $\mathcal{Y} = \mathbb{R}^n$) for a given observation $x \in \mathcal{X}$ such that some measure of deviation between the estimate $\hat{y}(x)$ and the random variable $y|x$ is minimized. Various methods have been used to accomplish this task, from linear models, over neural networks, regression splines, Watson-Nadaraya estimators, to Regularization Networks, Support Vector Machines, and Gaussian Processes (Wahba, 1990; Williams, 1998).

Most regression methods rely on the assumption that the noise level is uniform throughout the domain (Williams, 1998), or at least, its functional dependency is known beforehand (Schölkopf et al., 2000). For instance, in Gaussian Process regression one treats the noise level as a hyperparameter and solves the arising nonconvex problem by iterative optimization.

The assumption of a uniform noise model, however, is not always satisfied. In other words, the amount of noise may depend on the location. In this paper, we deal with a means of addressing this issue in a Gaussian Process regression framework. More specifically, we make the noise itself a random variable which is estimated nonparametrically along with the mean of the distribution as proposed in (Goldberg et al., 1998). In this sense, our methods closely resembles (Cawley et al., 2003) and (Yuan & Wahba, 2004) which independently proposed a kernel method to deal with this problem in the context of LMS regression: they truly estimate $\mu(x) = \mathbb{E}(Y|x)$ and $\log \sigma^2(x)$ with $\sigma^2(x) = V(Y|x)$ directly. While being intuitively appealing, it has a major shortcoming: the optimization problems arising from the setting are nonconvex, potentially exhibiting several local optima. Moreover, the variance need not be log-normally distributed.

To turn this nonconvex optimization problem into a convex one, our idea is to change the parameterization of the problem. Instead of directly estimating the functional mean and variance parameters, we propose to estimate the associated natural parameter $\theta_1(x) = \mu(x)/\sigma^2(x)$ and $\theta_2(x) = 1/\sigma^2(x)$ of the exponential family representation of the normal distribution. Now, the optimization problems using the exponential family setting are convex.

The challenge to solve this problem is to obtain a statistically plausible setting while retaining attractive properties of the optimization problem, such as convexity. Secondly, in the case of kernel methods, there is the problem of choosing a suitable covariance function. We will address this issue only as far as it concerns the overall formulation of the optimization problem: automatic kernel adaptation methods, such as (Lanckriet et al., 2004) apply.

Appearing in *Proceedings of the 22nd International Conference on Machine Learning*, Bonn, Germany, 2005. Copyright 2005 by the author(s)/owner(s).

2. The Model

Assume that we have a conditionally normal random variable, that is, $y|x \sim \mathcal{N}(\mu(x), \Sigma(x))$. In this case, $p(y|x)$ is a member of the exponential family for appropriate sufficient statistics $\Phi(x, y)$. It is well known that in such cases the negative log-likelihood $-\log p(y|x;\theta)$ is a *convex* function in the natural parameter θ. As we shall see below, this can be used to estimate both *mean* and *covariance* of the process in a pointwise fashion, which leads to heteroscedastic regression estimators.

2.1. Exponential Families

We begin with basic facts about exponential families. Let $\mathcal{X}$ be the domain of x and $\langle \cdot, \cdot \rangle$ the scalar product in a Hilbert space. Denote by $\phi(x)$ the sufficient statistics of x. In exponential families the density $p(x;\theta)$ is

$$p(x;\theta) = \exp\left(\langle \phi(x), \theta \rangle - g(\theta)\right) \tag{1a}$$

$$\text{where } g(\theta) = \log \int_{\mathcal{X}} \exp(\langle \phi(x), \theta \rangle)\, dx. \tag{1b}$$

Here, θ is the natural parameter and $g(\theta)$ is the log-partition function, often also called the cumulant generating function. This setting can be extended to conditional densities via

$$p(y|x;\theta) = \exp\left(\langle \phi(x, y), \theta \rangle - g(\theta|x)\right) \tag{2}$$

In analogy to the above $\phi(x, y)$ now denotes the sufficient statistics of $y|x$ for the purpose of estimation of conditional probabilities and we will refer to $g(\theta|x)$ as the conditional log-partition function. It is well known that $g(\theta)$ and $g(\theta|x)$ are convex C^∞ functions in θ.

2.2. Estimation with Exponential Families

Assume that we observe iid (independently and identically distributed) data drawn from a distribution $p(x, y)$, i.e. $(X, Y) := \{(x_1, y_1), \ldots, (x_m, y_m)\} \subset \mathcal{X} \times \mathbb{R}^n$. One possibility to estimate θ is to maximize the likelihood $p(Y|X;\theta)$. This is equivalent to minimizing

$$-\log p(Y|X;\theta) = \sum_{i=1}^m g(\theta|x_i) - \sum_{i=1}^m \langle \phi(x_i, y_i), \theta \rangle\,. \tag{3}$$

Clearly if $\phi(x, y)$ is high dimensional, minimization of (3) will lead to overfitting. Hence we need to introduce a prior on θ. A popular choice (Altun et al., 2004) is to use a normal prior, that is $p(\theta) \propto \exp(-\frac{1}{2\lambda^2}\|\theta\|^2)$. This leads to the following negative log-posterior on θ

$$\sum_{i=1}^m g(\theta|x_i) - \sum_{i=1}^m \langle \phi(x_i, y_i), \theta \rangle + \frac{1}{2\lambda^2}\|\theta\|^2 + c \tag{4}$$

It can be minimized by convex optimization. Due to an extension of the representer theorem (Wahba, 1990) the minimizer θ^* of (4) satisfies

$$\theta^* \in \operatorname{span}\{\phi(x_i, y) \text{ where } 1 \le i \le m \text{ and } y \in \mathcal{Y}\} \tag{5}$$

Eq. (5) allows us to expand (4) in terms of scalar expansion coefficients and to use the kernels

$$k((x, y), (x', y')) = \langle \phi(x, y), \phi(x', y') \rangle\,. \tag{6}$$

This allows us to avoid evaluating $\phi(x, y)$ directly at all. Moreover, this also leads to a Gaussian Process estimation problem by defining the stochastic process

$$t(x, y) := \langle \phi(x, y), \theta \rangle \text{ where } \theta \sim \mathcal{N}(0, \lambda^2 \mathbf{1}). \tag{7}$$

It is well known that (7) leads to a Gaussian Process with covariance kernel $\lambda^2 k((x, y), (x', y'))$ (Williams, 1998; Altun et al., 2004).

2.3. Conditionally Normal Distributions

The above discussion covers a large class of exponential family estimates ranging from classification over CRFs to spatial Poisson models. We now specify the choice of $\phi(x, y)$ which will lead to regression estimators. We begin with an unconditionally normal model:

$$\begin{aligned} p(y) &= (2\pi)^{\frac{-n}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left(-\tfrac{1}{2}(y-\mu)^\top \Sigma^{-1}(y-\mu)\right) \\ &= \exp\left(\langle (y, -0.5yy^\top), (\theta_1, \theta_2) \rangle - g(\theta_1, \theta_2)\right) \end{aligned} \tag{8}$$

Here, $\theta_2 = \Sigma^{-1}$, $\theta_1 = \Sigma^{-1}\mu$ and $\phi(y) := (y, -0.5yy^\top)$. Clearly $\theta_2 \succeq 0$, as the variance - covariance matrix has to be positive definite. Moreover, the log-partition function is given by

$$g(\theta_1, \theta_2) = \frac{1}{2}\theta_1^\top \theta_2^{-1} \theta_1 + \frac{n}{2}\log 2\pi - \frac{1}{2}\log\det\theta_2 \tag{9}$$

As we are concerned with *conditional* normal distributions, that is, where $y|x$ is normal, we need to design $\phi(x, y)$ such that for fixed x, $\phi(x, y)$ only contains linear and quadratic terms in y. We have:

Lemma 1 *If $y|x$ is conditionally normal, the sufficient statistics $\phi(x, y)$ decompose into $\phi(x, y) = (\phi_1(x, y), \phi_2(x, y))$, where $\phi_1(x, y)$ is homogeneous linear and $\phi_2(x, y)$ is homogeneous quadratic in y.*

2.4. Parametric Expansion

We now cast the MAP estimation problem into one of finding scalar coefficients. From (5) we know that the mode of $p(\theta|X, Y)$ lies in the span of $\phi(x_i, y)$ for all x_i and for all $y \in \mathcal{Y}$. Naively, this would lead to

$$\theta^* = \sum_{i=1}^m \sum_{y \in \mathcal{Y}} \alpha_{iy} \phi(x_i, y). \tag{10}$$

This attempt, however, is doomed to fail, as $\mathcal{Y} = \mathbb{R}$ or $\mathcal{Y} = \mathbb{R}^n$: we would have an infinite number of terms. However, we know from Lemma 1 that $\phi(x, y)$ can be decomposed into terms $\phi_{1i}(x)$ and $\phi_{2ij}(x)$ for $1 \leq i, j \leq n$. We denote $k_{1i}(x, x') := \langle \phi_{1i}(x), \phi_{1i}(x) \rangle$ and $k_{2ij}(x, x') := \langle \phi_{2ij}(x), \phi_{2ij}(x') \rangle$ the corresponding inner products. This allows us to compute the various terms of required for (4) as follows:

$$\left[\Sigma^{-1}(x)\mu(x)\right]_i = \sum_{l=1}^{m} \alpha_{1il} k_{1i}(x_l, x) \tag{11}$$

$$\left[\Sigma^{-1}(x)\right]_{ij} = \sum_{l=1}^{m} \alpha_{2ijl} k_{2ij}(x_l, x) \tag{12}$$

$$\begin{aligned}\langle \phi(x, y), \theta \rangle = &\sum_{i=1}^{m} \sum_{u=1}^{n} \alpha_{1ui} y_u k_{1u}(x_i, x) - \\ &\frac{1}{2} \sum_{i=1}^{m} \sum_{u,v=1}^{n} y_u y_v \alpha_{2uvi} \alpha_{2uvj} k_{2u}(x_i, x).\end{aligned} \tag{13}$$

$$\begin{aligned}\|\theta\|^2 = &\sum_{i,j=1}^{m} \sum_{u=1}^{n} \alpha_{1ui} \alpha_{1uj} k_{1u}(x_i, x_j) + \\ &\sum_{i,j=1}^{m} \sum_{u,v=1}^{n} \alpha_{2uvi} \alpha_{2uvj} k_{2u}(x_i, x_j).\end{aligned} \tag{14}$$

Here, the indices $[\cdot]_i$ and $[\cdot]_{ij}$ denote the i-th coordinate of a vector and the ij-th entry of a matrix respectively, and $\alpha \in \mathbb{R}^{m \times n + m \times n \times n}$. Plugging the expansions into the above derivation leads to a regression algorithm in its fullest generality.

Unfortunately, deriving a Newton-type method for this, which requires computation of second derivatives and semidefinite matrix constraints, is a problem best left to computer algebra tools, as it involves arrays of tensors of the fourth order.

Consequently, for the rest of the paper we focus on special cases, such as one-dimensional heteroscedastic regression ($n = 1$) and rotation invariant situations. This will be sufficient to convey the fundamental ideas without obscuring the main points of the model.

3. Analysis and Special Cases

3.1. Scalar Version

Matters are much simpler in the case of scalar regression $\mathcal{Y} = \mathbb{R}$, as mean and variance are now scalars, too. Without loss of generality we set

$$\phi(x, y) = (y\phi_1(x), -0.5y^2\phi_2(x)). \tag{15}$$

This leads to the estimates for mean and variance:

$$\sigma^{-2}(x) = \langle \theta_2, \phi_2(x) \rangle \tag{16a}$$

$$\mu(x)\sigma^{-2}(x) = \langle \theta_1, \phi_1(x) \rangle \tag{16b}$$

Before going into technical details we discuss a small (but important) modification of the model of the previous section: Using a normal prior on θ implies that the mode of the prior is obtained for $\langle \phi(x, y), \theta \rangle = 0$. In other words, the prior peaks for random variables with infinite variance and zero mean. This is not an acceptable modeling assumption and we can fix it easily by introducing an offset to θ.

$$\sigma^{-2}(x) = \langle \theta_2, \phi_2(x) \rangle + \bar{\theta}_2 \tag{17}$$

where $\bar{\theta}_2 > 0$ with the constraint $\langle \theta_2, \phi_2(x) \rangle + \bar{\theta}_2 \geq 0$. If no prior knowledge is given about this offset $\bar{\theta}_2$ it can be set to the constant variance estimation of a Gaussian Process regression. The log-partition function $g(\theta|x)$ becomes

$$\begin{aligned}g(\theta|x) = &\frac{1}{2} \langle \theta_1, \phi_1(x) \rangle^2 (\langle \theta_2, \phi_2(x) \rangle + \bar{\theta}_2)^{-1} \\ &- \frac{1}{2} \log(\langle \theta_2, \phi_2(x) \rangle + \bar{\theta}_2) + c.\end{aligned} \tag{18}$$

where c is some constant offset (see equation (9)). We are now in a position to state the negative log-posterior in terms of coefficients $\alpha_1, \alpha_2 \in \mathbb{R}^m$. For this purpose, we denote K_1 the matrix obtained from $K_{1ij} := \langle \phi(x_i), \phi(x_j) \rangle$ and K_2 analogously. This yields the following expansion whose maximization is the same as maximizing $p(\theta|X, Y)$:

$$\begin{aligned}\sum_{i=1}^{m} \Big[&-y_i [K_1\alpha_1]_i + \frac{y_i^2}{2} [K_2\alpha_2]_i + \frac{[K_1\alpha_1]_i^2}{2([K_2\alpha_2]_i + \bar{\theta}_2)} \\ &- \frac{1}{2} \log([K_2\alpha_2]_i + \bar{\theta}_2) \Big] + \frac{1}{2\lambda^2} \left[\alpha_1^\top K_1 \alpha_1 + \alpha_2^\top K_2 \alpha_2 \right]\end{aligned} \tag{19a}$$

$$\text{subject to } [K_2\alpha_2]_i + \bar{\theta}_2 \geq 0 \text{ for all } i. \tag{19b}$$

Details on how to solve (19) are given in Section 4. It is a convex problem with linear constraints. Eq. (19b), however, is unnecessary as $\log([K_2\alpha_2]_i + \bar{\theta}_2)$ provides a sufficient barrier function. So as long as we start with a feasible set of variables, which can easily be obtained by α_2, and we never let the objective function diverge to ∞ we can ignore (19b).

3.2. Constant Variance

Of particular interest is the special case where the variance is assumed to be constant. There (15) can be simplified to

$$\phi(x, y) = (y\phi_1(x), -0.5y^2). \tag{20}$$

With $\theta_2 \in \mathbb{R}$ we may simplify (19) further to obtain

$$\sum_{i=1}^{m} \Big[-y_i [K_1 \alpha_1]_i + \frac{y_i^2}{2} \theta_2 + \frac{[K_1 \alpha_1]_i^2}{2(\theta_2 + \bar{\theta}_2)} \Big] \qquad (21)$$
$$- \frac{m}{2} \log(\theta_2 + \bar{\theta}_2) + \frac{1}{2\lambda^2} \left[\alpha_1^\top K_1 \alpha_1 + \theta_2^2 \right]$$

subject to $\theta_2 + \bar{\theta}_2 \geq 0$. As before, this is a convex optimization problem. For fixed θ_2 it is easy to see (by taking derivatives with respect to α_1 that the optimal choice of α_1 needs to satisfy

$$-K_1 y + \frac{1}{\theta_2 + \bar{\theta}_2} K_1^\top K_1 \alpha_1 + \frac{1}{\lambda^2} K_1 \alpha_1 = 0 \qquad (22)$$

This is solved by $\alpha_1 = (\frac{1}{\theta_2 + \bar{\theta}_2} K_1 + \frac{1}{\lambda^2} \mathbf{1})^{-1} y$ provided this regularized matrix has full rank. Taking into account that $[K_1 \alpha_1] = \mu(x) \sigma^{-2}(x)$ and that $\sigma^{-2}(x) = \theta_2 + \bar{\theta}_2$ is constant yields a solution identical to standard Gaussian Process regression.

What is typically done in GP regression is that one solves for α_1 for fixed θ_2 and subsequently an optimal θ_2 given α_1 is found, and so on iteratively until some convergence is reached. This is highly suboptimal, and in implementations we found that such a procedure may take over 100 iterations (in the heteroscedastic case) until it converges to its minimum. Eq. (20) on the other hand offers a convex joint optimization problem, which allows for fast convergence in only a few iterations. We will discuss details in Section 4.

In traditional GP regression literature θ_2 is treated as a hyperparameter, that is, as a variable extraneous to the overall regression problem. What the above reasoning shows, though, is that θ_2 is an integral part of a convex optimization problem which allows joint estimation of mean and variance. Hence the algorithmic improvements of the current paper also apply to the homoscedastic case.

3.3. Rotation Invariant Settings

We now proceed to discuss a setting for vector valued regression, i.e. $y \in \mathcal{Y} = \mathbb{R}^n$, where the equations can be stated more compactly than in Section 2.4. More specifically, we assume that

$$\phi(x, y) = (y \otimes \phi_1(x), -0.5 y y^\top \otimes \phi_2(x)). \qquad (23)$$

Here, $\otimes$ denotes the tensor product. It follows from (23) and the spherically symmetric prior on θ that every entry of the linear (and quadratic term respectively) in $\phi(x, y)$ is treated in the same fashion. This assumption implies that our model is rotation invariant in the observations. The following holds:

Theorem 2 *Denote by $\phi(x, y)$ the sufficient statistics of a conditional normal distribution $y|x$. Then if for all θ and for all rotations $O \in \mathrm{SO}(n)$ there exists $\theta' = O\theta$ such that*

$$p(y|x; \theta) = p(Oy|x; \theta') \text{ for all } y \in \mathbb{R}^n \text{ and } x \in \mathcal{X}.$$

then $\phi(x, y)$ satisfies (23) for some ϕ_1, ϕ_2.

Proof For fixed x, θ, θ', O the above condition and choosing arbitrary $y \in \mathbb{R}^n$ implies that for $\log p(y|x; \theta)$ and $\log p(Oy|x; \theta')$ all of the terms constant, linear and quadratic in y need to match.

Moreover, since this equality simultaneously also has to hold for all θ, we can infer that there must exist corresponding features in each linear and quadratic entry. This, however, is only satisfied for a Cartesian product setting, which proves the claim. ■

The above theorem implies that for most practical purposes (23) is the right choice of sufficient statistics should we wish to perform joint regression[1].In terms of implementation it means that we can drop the double and triple indices in k as required in Section 2.4. Instead, we have

$$\langle \phi(x, y), \theta \rangle = \sum_{i=1}^{m} y^\top \alpha_{1i} k_1(x_i, x) - \frac{1}{2} y^\top \alpha_{2i} y k_2(x_i, x)$$
$$\|\theta\|^2 = \sum_{i,j=1}^{m} \alpha_{1i}^\top \alpha_{1j} k_1(x_i, x_j) + \mathrm{tr}(\alpha_{2i}^\top \alpha_{2j} k_2(x_i, x_j))$$

where $\alpha_{1i} \in \mathbb{R}^n$ and $\alpha_{2i} \in \mathbb{R}^{n \times n}$. Plugging the result into the log-posterior we obtain a convex problem in complete analogy to (19). Details are omitted for the sake of brevity.

An important special case is homoscedastic vector-valued regression. In other words, $\phi(x, y) = (y \otimes \phi_1(x), -0.5 y y^\top)$. Here, we estimate both the mean and the joint noise covariance matrix simultaneously.

3.4. Predictive Variance and Uncertainty

So far, we have not made much use of the Bayesian setting of the problem besides performing maximum a posteriori estimation. In particular, we ignored the fact that a true estimate would require integrating out θ leading to

$$p(Y|X) = \int \prod_i p(y_i | x_i, \theta) p(\theta) d\theta. \qquad (24)$$

[1] Note that this is for instance the case for geographical coordinates.

Inference on novel observations is carried out by computing $p(Y_{\text{test}}|Y_{\text{train}}, X)$. In the case of GP regression with fixed variance this is possible, as Y can be seen to be the sum of two GPs: one inducing correlations between observations with covariance kernel $k(x, x')$ and an iid Gaussian noise process.

In the case of intractable integrals, conversely, one resorts to approximations such as the Maximum a Posteriori estimate. This is how the optimization problems in the previous sections were derived. However, we can obtain somewhat more information, as for fixed variance (i.e. fixed θ_2) the integral with respect to θ_1 becomes tractable. Again, as before, we only deal with the scalar case to reduce index clutter. Extensions to the vector-valued setting are straightforward.

Our derivation works as follows: first we transform the problem in θ_1 into one of dealing with random variables drawn from a Gaussian Process via

$$t(x) := \langle \phi_1(x), \theta_1 \rangle \text{ where } \theta_1 \sim \mathcal{N}(0, \lambda^2 \mathbf{1}) \tag{25}$$

It is well known (Williams, 1998) that t is a GP with mean 0 and covariance kernel $\lambda^2 \langle \phi_1(x), \phi_1(x') \rangle = \lambda^2 k_1(x, x')$. Hence, observing $T \in \mathbb{R}^m$ on X yields

$$T \sim \mathcal{N}(0, \lambda^2 K_1). \tag{26}$$

Moreover, from (16) it follows that

$$y(x) \sim \mathcal{N}(\sigma^2(x) t(x), \sigma^2(x)) \tag{27}$$

where $\sigma^2(x)$ is given by (17). As fixing θ_2 implies fixing $\sigma^2(x)$ we can now simply infer the distribution of $Y \in \mathbb{R}^m$, as it is the sum of two normal random variables. Let $S = \operatorname{diag}\{\sigma^2(x_1), \ldots, \sigma^2(x_m)\} \in \mathbb{R}^{m \times m}$. We obtain

$$Y \sim \mathcal{N}(0, S + \lambda^2 S K_1 S). \tag{28}$$

This means that having observed Y_{train} we see that $Y_{\text{test}}|Y_{\text{train}}$ is conditionally normal with

$$Y_{\text{test}}|Y_{\text{train}} \sim \mathcal{N}\left(S_{\text{test}} M, S_{\text{test}} V S_{\text{test}})\right) \tag{29}$$

where

$$\begin{aligned} M =& K_{\text{cross}}(\lambda^{-2} S_{\text{train}}^{-1} + K_{\text{train}})^{-1} S_{\text{train}}^{-1} Y_{\text{train}} && \text{(30a)} \\ V =& S_{\text{test}}^{-1} + \lambda^2 K_{\text{test}} - && \text{(30b)} \\ & \lambda^2 K_{\text{cross}}(\lambda^{-2} S_{\text{train}}^{-1} + K_{\text{train}})^{-1} K_{\text{cross}}^{\top} \end{aligned}$$

and where we decomposed S and K_1 via

$$S = \begin{bmatrix} S_{\text{train}} & 0 \\ 0 & S_{\text{test}} \end{bmatrix} \text{ and } K_1 = \begin{bmatrix} K_{\text{train}} & K_{\text{cross}} \\ K_{\text{cross}}^{\top} & K_{\text{test}} \end{bmatrix}.$$

One may check that (30a) is identical to the estimate obtained by minimizing (19) with respect to θ_1 for fixed θ_2. This means that the overall estimator is obtained by the following algorithm:

Algorithm 1 Heteroscedastic regression

```
1. Solve optimization problem (19) to obtain the
   MAP estimate for the variance and mean inherent
   to the estimation problem.
2. Compute the predictive variance according to
   (30b) by a matrix inversion.
```

3.5. Comparison to Existing Methods

In the following we compare our approach, as described in Section 2.3 to the heteroscedastic estimators of (Yuan & Wahba, 2004; Cawley et al., 2003). In both of those papers the following model is used to estimate $p(y|x, \theta)$:

$$\begin{aligned} p(y|x, \theta) &= \exp\left(-\frac{(y - \mu(x))^2}{2\sigma^2(x)} - \frac{1}{2}\log(2\pi\sigma^2(x))\right) \\ \mu(x) &= \langle \phi_1(x), \theta_1 \rangle && \text{(31)} \\ \sigma^2(x) &= \exp(\langle \phi_2(x), \theta_2 \rangle). \end{aligned}$$

Moreover, a normal prior on θ_1, θ_2 is assumed. This yields the following likelihood term:

$$\begin{aligned} -\log p(y|x, \theta) =& \frac{1}{2}(y - \langle \phi_1(x), \theta_1 \rangle)^2 \exp(-\langle \phi_2(x), \theta \rangle) \\ &+ \frac{1}{2}\langle \phi_2(x), \theta \rangle. && \text{(32)} \end{aligned}$$

While $-\log p(y|x, \theta)$ is obviously convex in θ_1 and θ_2 whenever the other parameter is fixed, one can check that this is not the case when dealing with the optimization over (θ_1, θ_2) jointly.

Note that (31) requires the estimator to allocate its capacity to equal proportions in the noisy and in the noise-free regime. This may not be very useful, as we can expect the noisy observations to be more unreliable and that they vary to a larger degree. Reparameterizing by the rescaled mean and the inverse variance, as done in Section 2.3 resolves this problem.

4. Optimization

At first sight, (19) represents a formidable problem. For m observations, we have m nonnegativity constraints, $2m$ variables and a dense matrix in $\mathbb{R}^{2m \times 2m}$. A naive approach would require in the order of $6m^2$ `double` storage, and in the order of $1000m^3$ floating point operations. This makes problems of size larger than $m = 4000$ impracticable for workstations with 1GB of memory. Instead, we design an algorithm which is *linear* in the sample size (both storage and CPU) and quadratic in the precision required.

4.1. Newton Type Approach with Line Search

Denote by $F(\alpha)$ the objective function (19a) of our problem. The first issue we need to take care of is strict feasibility of the solution set at any time. As long as the initial guess of (α_1, α_2) satisfies the linear constraints (19b), the logarithmic term in (19a) ensures that we never leave the domain of feasibility as long as we perform descent on the objective function.[2] Setting $\alpha_1 = \alpha_2 = 0$ does the trick, due to the nonzero offset $\bar{\theta}_2 > 0$. In a Newton method, the search direction is computed by

$$\Delta\alpha = -(\partial^2_\alpha F(\alpha))^{-1} \partial_\alpha F(\alpha) \tag{33}$$

In order to ensure that $\alpha + \Delta\alpha$ actually decreases the objective function, we perform line search in the direction of $\Delta\alpha$ (this can be done at only linear cost per function evaluation). This is important in particular wherever quadratic approximation of F underlying the Newton step is not very accurate, i.e. when we are far away from the optimum.

To ensure that $\alpha + \Delta\alpha$ remains feasible, we need to compute the maximum γ for which $\alpha_2 + \gamma\Delta\alpha_2$ is feasible, that is $K_2(\alpha_2 + \gamma\Delta\alpha_2) + \bar{\theta}_2 \geq 0$. We obtain this via

$$\gamma_{\max} = \min \left\{ \frac{[K_2\alpha_2]_i + \bar{\theta}_2}{-[K_2\Delta\alpha_2]_i} \text{ where } [K_2\Delta\alpha_2]_i < 0 \right\}$$

Subsequently a line search is carried out in the range

$$0 \leq \gamma \leq \min(10, 0.95\gamma_{\max}). \tag{34}$$

The uppper bound of 10 ensures that we do not waste too much time on parts of the line search which are unlikely to yield good improvements (in the quadratic convergence phase of the Newton method the step length be in the order of 1). Equally well, the bound of $0.95\gamma_{\max}$ ensures that we remain strictly feasible as the objective function diverges for $\gamma_{\max}$. This yields a first optimization algorithm (Algorithm 2).

4.2. Reduced Rank Expansion

The dominant term is to solve the linear system (33). As the Hessian may often be ill conditioned, we take advantage of this situation and use a conjugate gradient solver.

Algorithm 2 Heteroscedastic regression

```
Set $\alpha_1 = \alpha_2 = 0$
repeat
    Compute $\partial_\alpha F(\alpha)$ and $\partial^2_\alpha F(\alpha)$ and $\Delta\alpha$ using (33)
    Compute $\gamma$ using (34) and $\gamma_{\text{opt}}$ via line search.
    Update $\alpha \leftarrow \alpha + \gamma_{\text{opt}}\Delta\alpha$.
until $\|\partial_\alpha F(\alpha)\| < \epsilon_{\text{tol}}$
```

Such modifications make computation of systems up to size 7000 observations feasible (a total of 980MB for storing two kernel matrices). For further improvements we need to approximate the kernel matrices themselves, as their storage becomes the dominant part. We use positive diagonal pivoting (Fine & Scheinberg, 2001).

This factorization can be obtained at $O(mn^2)$ time and $O(mn)$ cost, where n is the dimensionality required. In particular, we can use a lower degree of resolution when it comes to estimating the variance than the mean parameter. This means that turning a standard GP regression estimator into a heteroscedastic estimator incurs not much overhead.

We have the following complexity: we need $O(mn)$ memory to store the relevant columns of the kernel matrices. Furthermore, we need $O(mn^2)$ time to determine the pivots. Computing a Hessian vector product costs $O(mn)$ time and we require c of them per Newton iteration. Assuming a constant number C of Newton steps we have a total budget of $O(mn^2 + Ccmn)$ operations, where c may be as large as n worst case.

4.3. Performance

To illustrate our reasoning, we ran our algorithm on toy data from (Yuan & Wahba, 2004) for varying sample sizes, as described in Section 5. Figure 1 shows the performance of the CG algorithm using explicit computation of the Hessian, implicit Hessian vector products and reduced rank expansions. The experiments were run on a 3GHz P4 workstation using Linux 2.6, a non-SSE3 version of ATLAS, Python 2.4 and using the CREST machine learning toolbox. The expected time requirements for our algorithm are $O(m^{2.1})$, $O(m^{1.4})$, and $O(m^{0.95})$ for explicit computation of the Hessian, implicit Hessian vector products, and reduced rank expansion respectively.

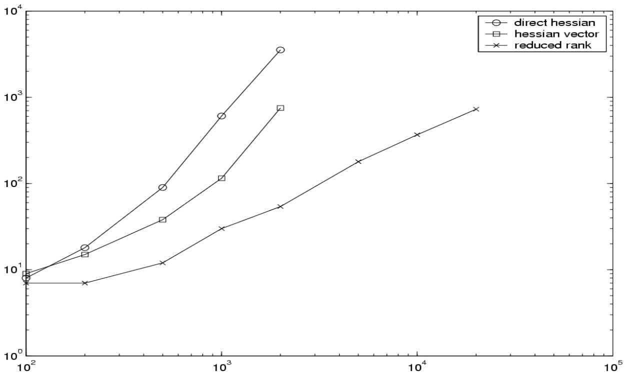
5. Experiments

Toy Data Following (Yuan & Wahba, 2004) we used the following normal distribution to generate data for illustrative purposes:

$$\mu(x_i) = 2(\exp(-30(x_i - 1/4)^2) + \sin(\pi x_i^2))$$
$$\text{and } \sigma^2(x_i) = \exp(2\sin(2\pi x_i)).$$

More specifically, we sampled m points in a regular grid on the interval $[0, 1]$ and randomly generated the corresponding y_i according to a normal law. Estima-

[2] $F(\alpha)$ is not even defined outside the feasible domain as it involves computing the logarithm of a negative number.



m	0.1	0.2	0.5	1	2	5	10	20
Direct Hessian	8	18	90	607	3551	-	-	-
Hessian vector	9	15	38	115	752	-	-	-
Reduced rank	7	7	12	30	54	179	368	727

Figure 1. Runtime of the Newton-type implementation (time in seconds vs. sample size (in 1000s) of the data set).

tion was performed using Gaussian kernels. Figure 2 shows the estimation for $m = 50$.

Spatial Interpolation Comparison 2004 The variable to be predicted is ambient radioactivity measured in Germany. The data by the German Federal Office for Radiation Protection (BfS), are gamma dose rates reported by means of the national automatic monitoring network (IMIS).[3]

The values of gamma dose rates at 200 locations measured for 10 days are used to tune the kernel hyperparameters and λ. It includes two scenarios for testing: set 1, which is a “normal” and set 2, which contains an anomaly in radiation at a specific location. As the training data for these two days was only made available to the participants, our training data was 200 points randomly picked from the 808 points from the given results.

We compared standard GP regression with our heteroscedastic GP (HGP) model. In both cases, we used Gaussian RBF kernels. To adjust the parameters, we used the 10 previous days to adapt the kernel widths and the prior variance by means of crossvalidation.

To assess performance we perform a 10 random estimates of the error for the 808 points of each of the two scenario datasets. More specifically (in keeping with the methodology of the benchmark) we use a random subset of 200 points for training and the remaining 608 points for testing. Fine-tuning of the prior variance parameters (with individual terms for k_1 and k_2 for HGP) on the subset of 200 points is done by 10-fold crossvalidation.

As can be seen in Table 1 the root mean squared error (RMSE) of GP regression and HGP regression on the normal scenario is very comparable. This is not surprising as we have no reason to believe that radioactivity would exhibit highly nonuniform behavior. On the anomaly scenario, however, GP regression is unable to cope with the increase in noise, whereas HGP adapts easily. Especially, our model predicts almost correctly the extreme values at the point of the explosion.

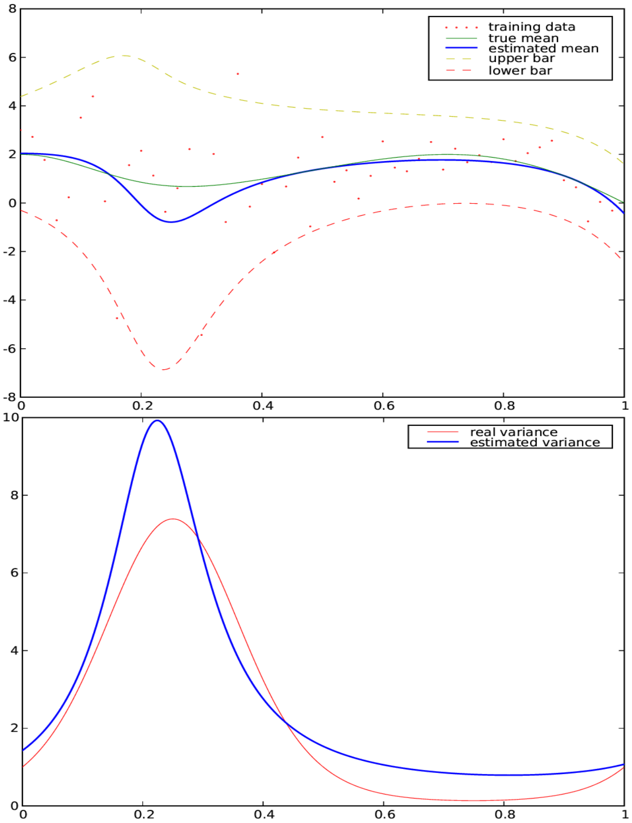


Figure 2. Top: Data, true mean, estimated mean, 95% confidence bars (dotted lines). Note that in the low-noise regions the regression is much more accurate. Bottom: variance estimate. The thin line represents the true variance while the thick line shows its estimate.

6. Summary and Outlook

We presented a scalable method for performing heteroscedastic GP regression.[4] In particular for reduced rank expansion HGP regression scales almost linearly.

More experiments and applications are needed to corroborate the algorithm further, e.g. details regarding the predictive variance, vector-valued estimation, etc. Applications are meteorological data, estimation of ore, denoising of DNA microarray measurements, and

[3] http://www.ai-geostats.org/events/sic2004/index.htm

[4] See http://sml.nicta.com.au/∼quoc.le for code.

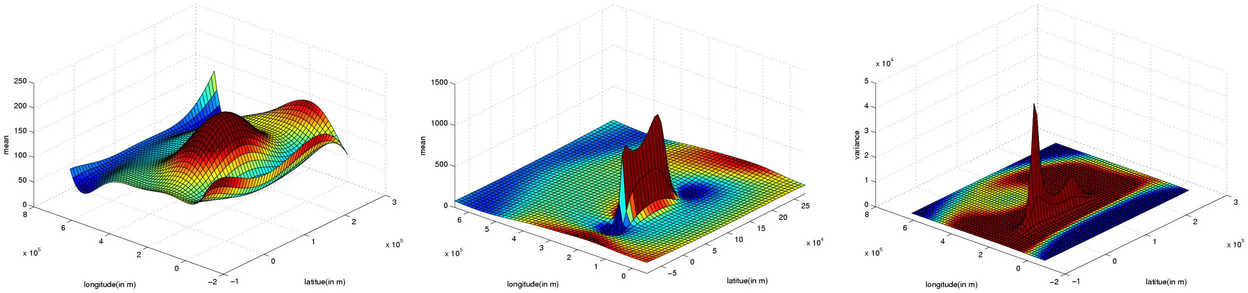


Figure 3. Spatial Interpolation Comparison 2004 Dataset (SIC2004) for set 2. Left: mean estimate for standard GP regression, Middle: mean estimate for Heteroscedastic GP regression, Right: variance estimate for Heteroscedastic GP regression. Note the difference in the mean estimate due to the fact that standard GPR is unable to adapt to the increase in noise at the location of the outbreak (GPR ends up oversmoothing).

	GP Regression (Williams, 1998)	log variance (Yuan & Wahba, 2004)	Heteroscedastic GP Regression (this paper)
Normal day	13.3 ± 0.4	12.1 ± 0.7	12.8 ± 0.6
Anomaly	86.8 ± 6.5	69.3 ± 7.2	49.4 ± 6.5

Table 1. RMSE for GPR, Heteroscedastic GPR, and log variance model

estimation of variance for MRI imaging experiments. Such work is currently ongoing and will be published separately.

Note that GP regression is not the only case where heteroscedasticity can be incorporated. For instance, in ν-SV regression one can turn the margin itself into a nonparametric function which is estimated together with the actual regression. The solution of this can be found as a quadratic program and we are currently researching this aspect.

Acknowledgements National ICT Australia is funded through the Australian Government's *Backing Australia's Ability* initiative, in part through the Australian Research Council. This work was supported by grants of the ARC and by the Pascal NoE. We thank Thomas Hofmann, Simon Burton, Cheng Soon Ong and Hao Shen for helpful discussions.

References

Altun, Y., Hofmann, T., & Smola, A. (2004). Exponential families for conditional random fields. *Uncertainty in Artificial Intelligence UAI*.

Cawley, G., Talbot, N., Foxall, R., Dorling, S., & Mandic, D. (2003). Approximately unbiased estimation of conditional variance in heteroscedastic kernel ridge regression. *European Symposium on Artificial Neural Networks* (pp. 209–214). d-side.

Fine, S., & Scheinberg, K. (2001). Efficient SVM training using low-rank kernel representations. *Journal of Machine Learning Research*, *2*, 243 – 264. http://www.jmlr.org.

Goldberg, P. W., Williams, C. K. I., & Bishop, C. M. (1998). Regression with input-dependent noise: a gaussian process treatment. *NIPS 10* (pp. 493–499). Cambridge, MA: MIT Press.

Lanckriet, G., Cristianini, N., Bartlett, P., Ghaoui, L. E., & Jordan, M. I. (2004). Learning the kernel matrix with semi-definite programming. *Journal of Machine Learning Research*, *5*, 27 – 72.

Schölkopf, B., Smola, A. J., Williamson, R. C., & Bartlett, P. L. (2000). New support vector algorithms. *Neural Computation*, *12*, 1207 – 1245.

Wahba, G. (1990). *Spline models for observational data*, vol. 59 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Philadelphia: SIAM.

Williams, C. K. I. (1998). Prediction with Gaussian processes: From linear regression to linear prediction and beyond. In M. I. Jordan (Ed.), *Learning and inference in graphical models*, 599 – 621. Kluwer Academic.

Yuan, M., & Wahba, G. (2004). *Doubly penalized likelihood estimator in heteroscedastic regression* (Technical Report 1084rr). University of Wisconsin.