Generalized Gaussian Process Models

Antoni B. Chan Daxiang Dong
Department of Computer Science
City University of Hong Kong
abchan@cityu.edu.hk, daxidong@cityu.edu.hk

Abstract

We propose a generalized Gaussian process model (GGPM), which is a unifying framework that encompasses many existing Gaussian process (GP) models, such as GP regression, classification, and counting. In the GGPM framework, the observation likelihood of the GP model is itself parameterized using the exponential family distribution. By deriving approximate inference algorithms for the generalized GP model, we are able to easily apply the same algorithm to all other GP models. Novel GP models are created by changing the parameterization of the likelihood function, which greatly simplifies their creation for task-specific output domains. We also derive a closed-form efficient Taylor approximation for inference on the model, and draw interesting connections with other model-specific closed-form approximations. Finally, using the GGPM, we create several new GP models and show their efficacy in building task-specific GP models for computer vision.

1. Introduction

In recent years, Gaussian processes (GPs) [1], a non-parametric Bayesian approach to regression and classification, have been gaining popularity in computer vision. For example, recent work [2] has demonstrated promising results on object classification using GP classification and active learning. GPs have several properties that are desirable for solving computer vision tasks. First, due to the Bayesian formulation, GPs can be learned robustly from small training sets, which is important in tasks where the amount of training data is sparse compared to the dimension of the model (e.g., large-scale object recognition, tracking, 3d human pose modeling). Second, the GP regression produces a predictive distribution, not just a single predicted value, thus providing a probabilistic approach to judging confidence in the predictions, e.g., for active learning. Third, GPs are based on kernel functions between the input examples, which allows for both a diverse set of image representations (e.g., bag-of-words, local-feature descriptors), and incorporation of prior knowledge about the computer vision task (e.g., modeling object structure). Finally, in the GP framework, the kernel hyperparameters can be learned by maximizing the marginal likelihood, or evidence, of the training data. This is typically more efficient than standard cross-validation (which requires a grid search), and allows for more expressive kernels, e.g., compound kernels that model different trends in the data, or multiple kernel learning, where features are optimally combined by adjusting the weights on each feature's kernel function.

Because of these advantages, GP regression and classification have been applied to many computer vision problems, such as object classification [2], human action recognition [3], age estimation [4], eye-gaze recognition [5], tracking [6], counting people [7, 8], crowd flow modeling [9], anomaly detection [10], stereo vision [11, 12], interpolation of range data [13] non-rigid shape recovery [14], 3d human pose recovery [15–18], and latent-space models of 3d human pose [19–21]. However, despite their successes, many of these methods attempt to "shoe-horn" their computer vision task into the standard GP regression framework. In particular, while the standard GP regresses a continuous *real-valued* function, it is often used to predict *discrete* non-negative integers (crowd counts [7] or age [4]), non-negative real numbers (disparity [11, 12] or depth [13]), real numbers on a fixed interval (pose angles [15–18] or squashed optical flow [10]), and coordinate pairs (bounding boxes [5]). Hence, heuristics are required to convert the real-valued GP prediction to a *valid task-specific output*, which is not optimal in the Bayesian setting. For example in [7], the real-valued GP prediction must be truncated and rounded to generate a proper count prediction, and it is not obvious how the predictive distribution over real-values can be converted to one over counts.

Currently, to develop a new GP model for each of the above regression tasks requires first finding a suitable distribution for the output variable (e.g., Poisson distribution for counting numbers, or a Gamma distribution for positive real values). Approximate inference is usually needed, due to the lack of conjugacy between the GP prior and the observation likelihood. As a result, developing a new GP model typically requires lengthy derivations of approximate inference for each particular likelihood function. What is currently lacking is a *general* framework that unifies the existing GP models, thus simplifying the creation of new GP models for different computer vision tasks.

In this paper, we propose a *unifying framework* that encompasses many existing GP models (e.g., regression, classification, and counting), which we call a *generalized Gaussian process model* (GGPM). In the GGPM framework, the

observation likelihood of the GP model is itself parameterized. Hence, existing GP models are simply instances of the GGPM with certain parameters. By deriving approximate inference for the generalized likelihood function of the GGPM, we are able to apply the same algorithm (which was previously derived for one model) to all other GP models. Within the framework, novel GP models are created by simply changing the likelihood function through its parameterization. *This greatly eases the creation of new GP models for task-specific output domains.*

The contributions of this paper are 3-fold: 1) we propose a generalized Gaussian process model (GGPM) based on the single-parameter exponential family distribution, creating a principled regression framework that can be easily adapted to specific output domains; 2) we derive a novel efficient approximate inference algorithm for GGPM based on a Taylor approximation, and show interesting connections to model-specific closed-form approximations from [2, 8]; 3) using the GGPM framework, we create several *new* GP models and demonstrate their efficacy on several computer vision tasks. The remainder of the paper is organized as follows. In Section 2, we first discuss related work. In Sections 3 and 4, we introduce the GGPM framework, while in Section 5 we derive an efficient approximate inference algorithm. Finally, in Section 6, we present several examples and experiments using GGPM.

2. Related work

Gaussian process regression (GPR) [1] is a Bayesian approach to predicting a real-valued function $f(\mathbf{x})$ of an input vector $\mathbf{x} \in \mathbb{R}^d$ (also known as the regressor or explanatory variable). The function value is observed through a noisy observation (or measurement or output) $y \in \mathbb{R}$, with zero-mean additive Gaussian noise, i.e. $p(y|f) = \mathcal{N}\left(y\middle|f, \sigma_n^2\right)$, where σ_n^2 is the observation noise. A zero-mean *Gaussian process* prior is placed on the function, $f \sim \mathcal{GP}(\mathbf{0}, k(\mathbf{x}, \mathbf{x}'))$, where $k(\mathbf{x}, \mathbf{x}')$ is the covariance function that specifies the class of functions that f will model (e.g., linear, polynomial, etc). GPR inference can be computed in closed-form, due to the conjugacy between the Gaussian observation likelihood and Gaussian prior.

For Gaussian process classification (GPC) [1, 22], a GP prior is again placed on the function f, which is then "squashed" through a sigmoid function to obtain the probability of the class $y \in \{0, 1\}$, i.e., $p(y = 1|f(\mathbf{x})) = \sigma(f(\mathbf{x}))$, where $\sigma(f)$ is the logistic or probit sigmoid functions. However, since the observation likelihood is no longer Gaussian, inference is no longer analytically tractable. This has led to the development of several approximate inference algorithms for GPC, such as Markov-chain Monte Carlo (MCMC) [22], variational bounds [23], Laplace's method [24], and expectation propagation [1, 25]. As an alternative to approximate inference, the classification task itself can be approximated as a GP *regression* problem, where the observations are set to $y \in \{-1, +1\}$. This is a computationally efficient alternative called *label regression* (or *least-square classification*) in [1, 22], and has shown promising results in object recognition [2].

GPR has been extended in several ways for different regression tasks. [26] proposes GP ordinal regression (i.e., ranking) using a multi-probit likelihood, while multiclass classification is obtained using a probit [27] or softmax [24] sigmoid function. Replacing the Gaussian observation likelihood with the Laplace or Cauchy likelihood leads to robust GP regression [28], while [8, 29, 30] develop counting regression using a Poisson observation likelihood and a GP prior. The goal of this paper is to generalize many of these models into a unified framework, thus allowing approximate inference algorithms derived for each specific model to be applied to the other models.

3. Generalized Gaussian process models

In this section, we introduce the generalized Gaussian process model, a non-parametric Bayesian regression model that encompasses many existing GP models.

3.1. Exponential family distributions

We first note that different GP models are obtained by changing the form of the observation likelihood $p(y|f)$. The standard GPR assumes a Gaussian observation likelihood, while GPC is obtained with a Bernoulli distribution, and [8] uses a Poisson likelihood for counting. These likelihood functions are all instances of the single-parameter *exponential family distribution* [31], with likelihood given by

$$p(y|\theta, \phi) = h(y, \phi) \exp\left\{\frac{1}{a(\phi)}\left[y\theta - b(\theta)\right]\right\}, \quad (1)$$

where $y \in \mathcal{Y}$ is the observation from set of possible values $\mathcal{Y}$ (e.g., real numbers, counting numbers, binary class labels). θ is the natural parameter of the exponential family distribution, and ϕ is the dispersion parameter. $a(\phi)$ and $h(y, \phi)$ are known functions, and $b(\theta)$ is the log-partition function, which normalizes the distribution. The mean and variance of y are functions of $b(\theta)$ and $a(\phi)$,

$$\mu = \mathbb{E}[y] = b'(\theta), \quad \mathrm{var}(y) = b''(\theta)a(\phi), \quad (2)$$

where $b'(\theta)$ and $b''(\theta)$ are the first and second derivatives of b w.r.t. θ. The exponential family distribution generalizes a wide variety of distributions for different output domains, which suggests that a unifying framework can be created by analyzing a GP model where the likelihood takes the *generic form* of (1).

3.2. Generalized Gaussian process models

We now consider a framework for a generic Bayesian model that regresses from inputs $\mathbf{x} \in \mathbb{R}^d$ to outputs $y \in \mathcal{Y}$,

which encompasses many popular GP models. The model is composed of three components:

1. a latent function, $\eta(\mathbf{x}) \sim \mathcal{GP}(0, k(\mathbf{x}, \mathbf{x}'))$, which is a function of the inputs, modeled with a GP prior.
2. a random component, $p(y|\theta, \phi)$, that models the output as an exponential family distribution with parameter θ;
3. a link function, $\eta = g(\mu)$, that relates the *mean* of the output distribution with the latent function.

Formally, the GGPM is specified by

$$\eta(\mathbf{x}) \sim \mathcal{GP}(0, k(\mathbf{x}, \mathbf{x}')), \quad y \sim p(y|\theta, \phi), \tag{3}$$
$$g(\mathbb{E}[y|\theta]) = \eta(\mathbf{x}), \tag{4}$$

where $k(\mathbf{x}, \mathbf{x}')$ is the covariance (or kernel) function, which defines the distribution over possible functions. The mean of the output distribution is related to the latent function $\eta(\mathbf{x})$, through the inverse-link function, i.e. $\mu = g^{-1}(\eta(\mathbf{x}))$. The advantage with using a link function is that it allows us to *directly specify prior knowledge* about the relationship (trend) between the output mean and the latent function $\eta(\mathbf{x})$. On the other hand, the effect of the GP kernel function is to adaptively warp (or completely override) the link function to fit the data. While many trends can be respresented by the GP kernel function (e.g., polynomial functions), it is important to note that some functions (e.g., $\log(x)$) *cannot be naturally represented by a kernel function*, due to its positive-definite constraint. Hence, directly specifying the link function is necessary for these cases.

Substituting (2) for the mean, we have

$$\eta(\mathbf{x}) = g(\mathbb{E}[y|\theta]) = g(b'(\theta)) \tag{5}$$

and thus, the parameter θ is a function of the latent function,

$$\theta(\eta(\mathbf{x})) = [b']^{-1}(g^{-1}(\eta(\mathbf{x}))). \tag{6}$$

The model is simplified when $g(\cdot)$ is selected to be the *canonical link function*, such that $\theta(\eta(\mathbf{x})) = \eta(\mathbf{x})$, i.e. $g(\cdot) = [b']^{-1}(\cdot)$. Using (6), another form of GGPM is

$$\eta(\mathbf{x}) \sim \mathcal{GP}(0, k(\mathbf{x}, \mathbf{x}')), \quad y \sim p(y|\theta(\eta(\mathbf{x})), \phi), \tag{7}$$
$$\theta(\eta(\mathbf{x})) = [b']^{-1}(g^{-1}(\eta(\mathbf{x}))). \tag{8}$$

Given a set of training examples and a novel input, the predictive distribution is obtained by marginalizing over the posterior of the latent function $\eta(\mathbf{x})$, as with standard GPR/GPC [1]. The dispersion ϕ is treated as a hyperparameter, which can be estimated along with the kernel hyperparameters by maximizing the marginal likelihood.

3.3. Other related work

The GGPM can be interpreted as a Bayesian approach to *generalized linear models* (GLMs) [32], where a GP prior with a linear kernel is placed on the systemic component (latent function). Other Bayesian GLMs have also been proposed in the literature. These mainly focus on inducing sparsity in the latent function, e.g., [33, 34] assumes a factorial heavy-tailed prior distribution, but is not kernelizable due to the factorial assumption. [35] proposes a Bayesian kernelized GLM, using a hierarchical model with a sparse prior (a mixture of point mass and Silverman's g-prior). The GGPM can also be seen as a Bayesian version of a *generalised kernel machines* [36], which is based on kernelizing iterated-reweighted least squares estimation (IRWLS).

While the connection between GPR/GPC and GLMs has been mentioned in the literature (e.g., [37, 38]), to our knowledge, a unified GP framework *has not been studied in depth*. In particular, there are no inference algorithms for the *general form* of the exponential family distribution (there are only inference algorithms derived for *specific* likelihood functions). The goal of this paper is to parameterize the likelihood function, thus creating a "plug-and-play" aspect to GP models. We exploit this property later to create several novel GP models with very little extra work.

4. Inference and Learning for GGPMs

Inference on GGPMs is similar to that of the standard GPR/GPC [1] . Given a set of training examples, input vectors $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n]$ and corresponding observations $\mathbf{y} = [y_1, \cdots, y_n]^T$, the goal is to generate a *predictive distribution* of the output y_* corresponding to a novel input $\mathbf{x}_*$. The distribution of the latent values $\boldsymbol{\eta} = [\eta(\mathbf{x}_1), \cdots, \eta(\mathbf{x}_n)]^T$, corresponding to the training inputs $\mathbf{X}$, is jointly Gaussian, $\boldsymbol{\eta}|\mathbf{X} \sim \mathcal{N}(0, \mathbf{K})$, where $\mathbf{K}$ is the kernel matrix with entries $k(\mathbf{x}_i, \mathbf{x}_j)$. Including the training outputs $\mathbf{y}$, the posterior distribution of $\boldsymbol{\eta}$ is obtained with Bayes' rule,

$$p(\boldsymbol{\eta}|\mathbf{X}, \mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta}(\boldsymbol{\eta}))p(\boldsymbol{\eta}|\mathbf{X})}{p(\mathbf{y}|\mathbf{X})}, \tag{9}$$

where $p(\mathbf{y}|\mathbf{X})$ is the marginal likelihood, or evidence,

$$p(\mathbf{y}|\mathbf{X}) = \int p(\mathbf{y}|\boldsymbol{\theta}(\boldsymbol{\eta}))p(\boldsymbol{\eta}|\mathbf{X})d\boldsymbol{\eta}. \tag{10}$$

Given a novel input $\mathbf{x}_*$, the posterior distribution of the novel latent value $\eta_* = \eta(\mathbf{x}_*)$ is obtained by marginalizing over the posterior distribution in (9) (i.e., averaging over all possible latent functions),

$$p(\eta_*|\mathbf{X}, \mathbf{x}_*, \mathbf{y}) = \int p(\eta_*|\boldsymbol{\eta}, \mathbf{X}, \mathbf{x}_*)p(\boldsymbol{\eta}|\mathbf{X}, \mathbf{y})d\boldsymbol{\eta}, \tag{11}$$

and $p(\eta_*|\boldsymbol{\eta}, \mathbf{X}, \mathbf{x}_*) = \mathcal{N}\left(\eta_*|\mathbf{k}_*^T\mathbf{K}^{-1}\boldsymbol{\eta}, k_{**} - \mathbf{k}_*^T\mathbf{K}^{-1}\mathbf{k}_*\right)$, with $\mathbf{k}_* = [k(\mathbf{x}_*, \mathbf{x}_i)]_i$ and $k_{**} = k(\mathbf{x}_*, \mathbf{x}_*)$. Finally, the y_* predictive distribution is obtained by marginalizing η_*,

$$p(y_*|\mathbf{X}, \mathbf{x}_*, \mathbf{y}) = \int p(y_*|\theta(\eta_*))p(\eta_*|\mathbf{X}, \mathbf{x}_*, \mathbf{y})d\eta_*. \tag{12}$$

4.1. Approximate inference

For most non-Gaussian likelihoods, the posterior and predictive distributions in (9, 10, 11, 12) cannot be computed analytically in closed-form. Hence, approximate inference algorithms are required. One choice is to use MCMC to draw samples from the posterior $p(\boldsymbol{\eta}|\mathbf{X},\mathbf{y})$, which can be computationally intensive [22]. Other inference approximations work by finding a suitable Gaussian approximation $q(\boldsymbol{\eta}|\mathbf{X},\mathbf{y})$ to the true posterior [22], i.e.

$$p(\boldsymbol{\eta}|\mathbf{X},\mathbf{y}) \approx q(\boldsymbol{\eta}|\mathbf{X},\mathbf{y}) = \mathcal{N}(\boldsymbol{\eta}|\hat{\mathbf{m}},\hat{\mathbf{V}}) \tag{13}$$

where the parameters $\{\hat{\mathbf{m}},\hat{\mathbf{V}}\}$ are determined by the type of approximation. Substituting the approximation $q(\boldsymbol{\eta}|\mathbf{X},\mathbf{y})$ into (11), the approximate posterior for η_* is

$$p(\eta_*|\mathbf{X},\mathbf{x}_*,\mathbf{y}) \approx q(\eta_*|\mathbf{X},\mathbf{y}_*,\mathbf{y}) = \mathcal{N}\left(\eta_*|\hat{\mu}_\eta,\hat{\sigma}^2_\eta\right), \tag{14}$$

where the mean and variance are

$$\hat{\mu}_\eta = \mathbf{k}_*^T\mathbf{K}^{-1}\hat{\mathbf{m}}, \tag{15}$$

$$\hat{\sigma}^2_\eta = k_{**} - \mathbf{k}_*^T(\mathbf{K}^{-1} - \mathbf{K}^{-1}\hat{\mathbf{V}}\mathbf{K}^{-1})\mathbf{k}_*. \tag{16}$$

In many inference approximations, $\{\hat{\mathbf{m}},\hat{\mathbf{V}}\}$ take the form

$$\hat{\mathbf{V}} = (\mathbf{K}^{-1} + \mathbf{W}^{-1})^{-1}, \quad \hat{\mathbf{m}} = \hat{\mathbf{V}}\mathbf{W}^{-1}\mathbf{t}, \tag{17}$$

where $\mathbf{W}$ is a positive definite diagonal matrix, and $\mathbf{t}$ is a target vector. In these cases, (15) and (16) can be rewritten

$$\hat{\mu}_\eta = \mathbf{k}_*^T\left(\mathbf{K}+\mathbf{W}\right)^{-1}\mathbf{t},\ \hat{\sigma}^2_\eta = k_{**} - \mathbf{k}_*^T(\mathbf{K}+\mathbf{W})^{-1}\mathbf{k}_*.$$

Note that these are equivalent to the standard equations for GPR, but with $\mathbf{W}$ and $\mathbf{t}$ determined by the approximate inference algorithm.

4.2. Learning the Hyperparameters

As in GPR, the kernel hyperparameters $\boldsymbol{\alpha}$ and the dispersion ϕ, are estimated from the data using Type-II maximum likelihood, which maximizes the marginal likelihood [1],

$$\{\boldsymbol{\alpha}^*,\phi^*\} = \underset{\boldsymbol{\alpha},\phi}{\operatorname{argmax}} \int p(\mathbf{y}|\boldsymbol{\eta},\phi)p(\boldsymbol{\eta}|\mathbf{X},\boldsymbol{\alpha})d\boldsymbol{\eta}, \tag{18}$$

where we now note the dependence on the hyperparameters. The marginal likelihood measures the data fit, averaged over all probable latent functions. Hence, the criteria selects the kernel hyperparameters such that each probable latent function will model the data well.

5. Approximate inference for GGPMs

In this section, we derive approximate inference algorithms for GGPMs based on the *general form* of the exponential family distribution in (1), i.e., using the likelihood parameters $\{a(\phi),b(\theta),h(y,\phi),\theta(\eta)\}$. We refer the reader to the supplemental [39] for derivations.

5.1. Taylor approximation

In this section, we derive a novel closed-form approximation to inference based on a Taylor approximation of the likelihood term. We first define the following derivative functions of the observation log-likelihood,

$$\begin{aligned} u(\eta,y) &= \frac{\partial}{\partial\eta}\log p(y|\theta(\eta)) = \frac{1}{a(\phi)}\theta'(\eta)\left[y - b'(\theta(\eta))\right], \\ w(\eta,y) &= -\left[\frac{\partial^2}{\partial\eta^2}\log p(y|\theta(\eta))\right]^{-1} \\ &= a(\phi)\left\{b''(\theta(\eta))\theta'(\eta)^2 - \left[y-b'(\theta(\eta))\right]\theta''(\eta)\right\}^{-1} \end{aligned} \tag{19}$$

For the canonical link function, these simplify to

$$u(\eta,y) = \frac{1}{a(\phi)}[y - b'(\eta)], \quad w(\eta,y) = \frac{a(\phi)}{b''(\eta)}. \tag{20}$$

5.1.1 Joint approximation

The joint likelihood of the data and latent values is

$$\log p(\mathbf{y},\boldsymbol{\eta}|\mathbf{X}) = \log p(\mathbf{y}|\boldsymbol{\theta}(\boldsymbol{\eta})) + \log p(\boldsymbol{\eta}|\mathbf{X}). \tag{21}$$

Next we form a second-order Taylor expansion of the data log-likelihood term at the point $\tilde{\eta}_i$,

$$\begin{aligned} &\log p(y_i|\theta(\eta_i)) = \frac{1}{a(\phi)}\left[y_i\theta(\eta_i) - b(\theta(\eta_i))\right] + \log h(y_i,\phi) \\ &\approx \log p(y_i|\theta(\tilde{\eta}_i)) + \tilde{u}_i(\eta_i - \tilde{\eta}_i) - \frac{1}{2}\tilde{w}_i^{-1}(\eta_i - \tilde{\eta}_i)^2 \end{aligned} \tag{22}$$

where $\tilde{u}_i = u(\tilde{\eta}_i,y_i)$ and $\tilde{w}_i = w(\tilde{\eta}_i,y_i)$. Defining $\tilde{\mathbf{u}} = [\tilde{u}_1,\cdots,\tilde{u}_n]^T$ and $\tilde{\mathbf{W}} = \operatorname{diag}(\tilde{w}_1,\ldots,\tilde{w}_n)$, the joint likelihood in (21) can be approximated as

$$\begin{aligned} \log q(\mathbf{y},\boldsymbol{\eta}|\mathbf{X}) &= \log p(\mathbf{y}|\theta(\tilde{\boldsymbol{\eta}})) - \frac{1}{2}\log|\mathbf{K}| - \frac{n}{2}\log 2\pi \\ &- \frac{1}{2}\left\|\boldsymbol{\eta} - \mathbf{A}^{-1}\tilde{\mathbf{W}}^{-1}\tilde{\mathbf{t}}\right\|^2_{\mathbf{A}^{-1}} - \frac{1}{2}\left\|\tilde{\mathbf{t}}\right\|^2_{\mathbf{W}+\mathbf{K}} + \frac{1}{2}\tilde{\mathbf{u}}^T\tilde{\mathbf{W}}\tilde{\mathbf{u}} \end{aligned} \tag{23}$$

where $\mathbf{A} = \tilde{\mathbf{W}}^{-1} + \mathbf{K}^{-1}$, $\tilde{\mathbf{t}} = \tilde{\boldsymbol{\eta}} + \tilde{\mathbf{W}}\tilde{\mathbf{u}}$ is the target vector, and the individual targets are $\tilde{t}_i = \tilde{\eta}_i + w(\tilde{\eta}_i,y_i)u(\tilde{\eta}_i,y_i)$.

5.1.2 Approximate posterior

From (23) , the posterior of $\boldsymbol{\eta}$ is approximately Gaussian,

$$\log q(\boldsymbol{\eta}|\mathbf{X},\mathbf{y}) \propto -\frac{1}{2}\left\|\boldsymbol{\eta} - \mathbf{A}^{-1}\tilde{\mathbf{W}}^{-1}\tilde{\mathbf{t}}\right\|^2_{\mathbf{A}^{-1}} \tag{24}$$

$$\Rightarrow \quad q(\boldsymbol{\eta}|\mathbf{X},\mathbf{y}) = \mathcal{N}(\boldsymbol{\eta}|\hat{\mathbf{m}},\hat{\mathbf{V}}), \tag{25}$$

where, $\hat{\mathbf{V}} = (\tilde{\mathbf{W}}^{-1} + \mathbf{K}^{-1})^{-1}$, and $\hat{\mathbf{m}} = \hat{\mathbf{V}}\tilde{\mathbf{W}}^{-1}\tilde{\mathbf{t}}$. These are of the form in (17), and hence, the approximate posterior of η_* has parameters

$$\hat{\mu}_\eta = \mathbf{k}_*^T(\mathbf{K}+\tilde{\mathbf{W}})^{-1}\tilde{\mathbf{t}},\ \hat{\sigma}^2_\theta = k_{**} - \mathbf{k}_*^T(\mathbf{K}+\tilde{\mathbf{W}})^{-1}\mathbf{k}_*.$$

The Taylor approximation is a closed-form (non-iterative) approximation, that can be interpreted as performing GPR on a set of targets $\tilde{\mathbf{t}}$ with target-specific non-i.i.d. observation noise $\tilde{\mathbf{W}}$. The targets $\tilde{\mathbf{t}}$ are a function of the the expansion point $\tilde{\boldsymbol{\eta}}$, which can be chosen as a non-linear transformation of the observations $\mathbf{y}$. One advantage with this Taylor approximation is that it is an *efficient non-iterative* method with the same complexity as GPR. Instances of the closed-form Taylor approximation for different GP models are further explored in Section 6.

5.1.3 Approximate Marginal

The approximate marginal likelihood is obtained by integrating out $\boldsymbol{\eta}$ in (23), yielding

$$\log q(\mathbf{y}|\mathbf{X}) = -\frac{1}{2}\tilde{\mathbf{t}}^T(\tilde{\mathbf{W}}+\mathbf{K})^{-1}\tilde{\mathbf{t}} - \frac{1}{2}\log\left|\tilde{\mathbf{W}}+\mathbf{K}\right| + r(\phi)$$

where $r(\phi) = \log p(\mathbf{y}|\theta(\tilde{\boldsymbol{\eta}})) + \frac{1}{2}\tilde{\mathbf{u}}^T\tilde{\mathbf{W}}\tilde{\mathbf{u}} + \frac{1}{2}\log|\tilde{\mathbf{W}}|$. This marginal is similar to that of GPR, but with modified targets and noise terms. There is also an additional penalty term on the dispersion ϕ, given by $r(\phi)$.

5.2. Laplace approximation

The Laplace approximation is a Gaussian approximation of the posterior $p(\boldsymbol{\eta}|\mathbf{X},\mathbf{y})$ at its maximum (mode). Hence, the Laplace approximation is a special case of the closed-form Taylor approximation in the previous section, where the target $\tilde{\mathbf{t}}$ is set to the maximum of the true posterior,

$$\hat{\boldsymbol{\eta}} = \underset{\boldsymbol{\eta}}{\operatorname{argmax}} \log p(\boldsymbol{\eta}|\mathbf{X},\mathbf{y}). \tag{26}$$

The true posterior mode is obtained iteratively using the Newton-Raphson method. The mode is unique when the log posterior is concave, or equivalently when $\mathbf{W}^{-1}$ is positive definite, i.e.

$$\begin{aligned}&\tfrac{1}{a(\phi)}\left\{b''(\theta(\eta))\theta'(\eta)^2 - [y-b'(\theta(\eta))]\,\theta''(\eta)\right\} > 0\\ &\quad\Rightarrow b''(\theta(\eta))\theta'(\eta)^2 > [y-b'(\theta(\eta))]\,\theta''(\eta)\end{aligned}$$

For a canonical link function, this simplifies to $b''(\eta_i) > 0$, i.e., a unique maximum exists when $b(\eta)$ is convex.

5.3. Expectation propagation

Expectation propagation (EP) [25] is a general algorithm for approximate inference, which has been shown to be effective for GPC [22]. EP approximates each likelihood term $p(y_i|\theta(\eta_i))$ with an unnormalized Gaussian $t_i = \tilde{Z}_i\mathcal{N}\left(\eta_i\middle|\tilde{\mu}_i,\tilde{\sigma}_i^2\right)$ (also called a site function). The posterior approximation is

$$q(\boldsymbol{\eta}|\mathbf{X},\mathbf{y}) = \frac{1}{Z_{EP}}\prod_{i=1}^{n} t_i(\eta_i)p(\boldsymbol{\eta}|\mathbf{X}) = \mathcal{N}\left(\boldsymbol{\eta}\middle|\hat{\mathbf{m}},\hat{\mathbf{V}}\right)$$

where $\{\hat{\mathbf{m}},\hat{\mathbf{V}}\}$ are given by (17) with $\mathbf{t} = \tilde{\boldsymbol{\mu}}$ and $\mathbf{W} = \tilde{\boldsymbol{\Sigma}}$, and $Z_{EP} = q(\mathbf{y}|\mathbf{X}) = \int q(y|\theta(\boldsymbol{\eta}))p(\boldsymbol{\eta}|\mathbf{X})d\boldsymbol{\eta}$ is the EP approximation of the marginal likelihood. The parameters of the site functions are iteratively optimized, which requires computing moments (mean, variance, and normalization) of $q(\eta_i) \propto p(y_i|\theta(\eta_i))\mathcal{N}\left(\eta_i\middle|\mu_{\neg i},\sigma^2_{\neg i}\right)$, where $\{\mu_{\neg i},\sigma^2_{\neg i}\}$ are parameters of the *cavity distribution* (more details in [1, 22]). Note that these moments may not be analytically tractable (in fact, $q(\eta_i)$ is the same form as the predictive distribution), so approximate integration is usually required.

6. Examples and Experiments

In this section, we present examples of both existing and novel GP models using GGPM. By simply changing the parameters of the exponential family distribution to form a specific observation likelihood (i.e., selecting the functions $\{a(\phi),b(\theta),h(y,\phi),\theta(\eta)\}$), we can easily obtain a wide range of GP models with different types of outputs.

The GGPM was implemented in MATLAB by extending the GPML toolbox [1] to include implementations for: 1) the generic exponential family distribution using the parameters $\{a(\phi),b(\theta),h(y,\phi),\theta(\eta)\}$; 2) the closed-form Taylor approximation for inference. EP moments and the predictive distributions are computed using numerical integration. Empirically, we found that EP was sensitive to the accuracy of the approximate integrals; there were convergence problems when other approximations were used (e.g. Gaussian-Hermite quadrature). Hyperparameters (dispersion and kernel parameters) were optimized by maximizing the marginal likelihood, using the existing GPML functions.

6.1. Binomial distribution

The binomial distribution models the probability of a certain number of events occurring in N independent trials, where the event probability in an individual trial is π,

$$p(y|\pi,N) = \tbinom{N}{Ny}\pi^{Ny}(1-\pi)^{N-Ny} \tag{27}$$

where $y \in \{\frac{0}{N},\frac{1}{N},\cdots,\frac{N}{N}\}$ is the fraction of events. With $\theta = \log\frac{\pi}{1-\pi}$ and $\phi = \frac{1}{N}$, the exponential family form is

$$a(\phi) = \phi,\;\; b(\theta) = \log(1+e^{\theta}),\;\; h(y,\phi) = \tbinom{N}{Ny}. \tag{28}$$

If we assume the canonical link function, then

$$\pi = \mathbb{E}[y] = g^{-1}(\eta) = b'(\eta) = \tfrac{e^{\eta}}{1+e^{\eta}}, \tag{29}$$

and hence the mean is related to the latent space through the logistic function. For $N = 1$, the Binomial-GGPM is equivalent to the GPC model using the logistic function. For $N > 1$, the model can naturally accommodate uncertainty in the labels by using fractional y_i, e.g., for $N = 2$ there are three levels $y \in \{0,\frac{1}{2},1\}$. Furthermore, by changing the

link function to the probit function, we obtain GPC using the probit likelihood,

$$g(\mu) = \Phi^{-1}(\mu), \Rightarrow g^{-1}(\eta) = \Phi(\eta)$$

where $\Phi(\eta)$ is the cumulative distribution of a Gaussian. Substituting into the GGPM, we have

$$\theta(\eta) = \log \tfrac{\Phi(\eta)}{1-\Phi(\eta)}, \quad b(\theta(\eta)) = -\log(1-\Phi(\eta)).$$

6.1.1 Inference by Taylor approximation

We next look at the Taylor approximation for the binomial-GGPM. The derivative functions are

$$u(\eta, y) = N(y - \tfrac{e^\eta}{1+e^\eta}), \;\; w(\eta, y) = \tfrac{(1+e^\eta)^2}{Ne^\eta}.$$

Thus, the target and effective noise are

$$t_i = \tilde{\eta}_i + \tfrac{(1+e^{\tilde{\eta}_i})^2}{e^{\tilde{\eta}_i}}(y_i - \tfrac{e^{\tilde{\eta}_i}}{1+e^{\tilde{\eta}_i}}), \; w_i = \tfrac{(1+e^{\tilde{\eta}_i})^2}{Ne^{\tilde{\eta}_i}}.$$

An agnostic choice of expansion point is $\tilde{\eta}_i = 0$, which ignores the training classes, leading to

$$t_i = 4(y_i - 0.5), \quad w_i = 4/N. \tag{30}$$

Hence, the Taylor approximation for binomial-GGPM is equivalent to GPR in the latent space of the binomial model, with targets t_i scaled between $[-2, +2]$ and an effective noise term $w_i = 4/N$. When $y_i \in \{0, 1\}$, the target values are $\{-2, +2\}$, which is equivalent to label regression [1, 2, 22] (up to a scaling). Hence, *label regression can be interpreted as a Taylor approximation to GPC inference!* The scaling of the targets (± 2 or ± 1) is irrelevant if we only use the latent space, i.e. when classifying using the sign of y. However, this scaling is important if we want to compute actual label probabilities using the predictive distribution.

Method	Inference	Avg. Error
GPC (1-vs-all)	EP	0.0866
Binomial-GGPM	Taylor	**0.0631**
Nearest Neighbors	–	0.1260
SVM	–	0.0905

Table 1. Average error for traffic classification.

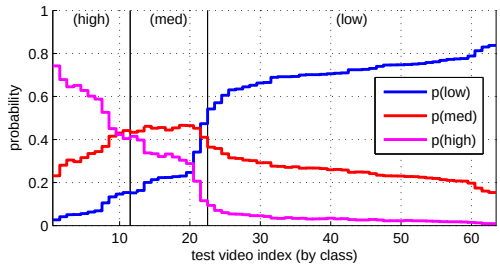

Figure 1. Probabilities of traffic classes using binomial-GGPM. Test videos are sorted by ground-truth class and $p(\text{low})$.

6.1.2 Experiments

We use the binomial-GGPM to perform ordinal classification on the highway traffic dataset from [40]. The class labels "low", "medium", and "high" traffic are assigned to the responses $y \in \{0, \frac{1}{2}, 1\}$ of the binomial model. This provides a natural ranking of the classes, that is not possible with normal 1-vs-all classification. Each video is represented with a dynamic texture, and the kernel function is the exponentiated Martin distance [40]. The experimental results are presented in Table 1, and show that binomial-GGPM outperforms the standard 1-vs-all GPC, with an average error of 0.0631 vs 0.0866. Figure 1 shows the predicted class probabilities for each test video. Unlike standard 1-vs-all classifiers, the binomial-GGPM produces class probabilities that are correlated with the ordering of the classes. The two GP models also outperform the nearest neighbors and SVM classifiers from [40] (see Table 1).

6.2. Poisson distribution

The Poisson distribution is a model for counting data,

$$p(y|\lambda) = \tfrac{1}{y!}\lambda^y e^{-\lambda}, \tag{31}$$

where $y \in \mathbb{Z}_+ = \{0, 1, \cdots\}$ are counts, and λ is the arrival-rate (mean) parameter. By setting $\theta = \log \lambda$ and $\phi = 1$, we obtain the exponential family form with

$$a(\phi) = 1, \;\; b(\theta) = e^\theta, \;\; h(y, \phi) = 1/y!. \tag{32}$$

The canonical link function is

$$\mathbb{E}[y] = g^{-1}(\eta) = e^\eta = \lambda, \;\; g(\mu) = \log \mu. \tag{33}$$

Hence, the mean of the Poisson is the exponential of the latent value. The Poisson-GGPM is a Bayesian regression model for predicting counts y from an input vector $\mathbf{x}$, and has been previously studied in [8, 29, 30].

6.2.1 Linearized mean

The canonical link function assumes that the mean is the exponential of the latent function. This may cause problems when this is not the case, as illustrated in Figure 2a, where the count actually follows a linear trend. One way to address this problem is to use a non-linear kernel function (e.g. RBF) to try to counteract the exponential link function. However, there is no kernel function for the logarithm, and hence errors occur at the extremes of the latent function.

Alternatively, the mean can be *directly linearized* by changing the link function of the Poisson-GGPM to be more linear. For this purpose, we use the logistic error function,

$$g^{-1}(\eta) = \log(1 + e^\eta) \;\Rightarrow\; g(\mu) = \log(e^\mu - 1), \mu > 0.$$

For large values of η, the link function is linear, while for negative values of η, the link approaches zero. The parameter function and new partition function are

$$\theta(\eta) = \log(\log(1 + e^\eta)), \quad b(\theta(\eta)) = \log(1 + e^\eta). \tag{34}$$

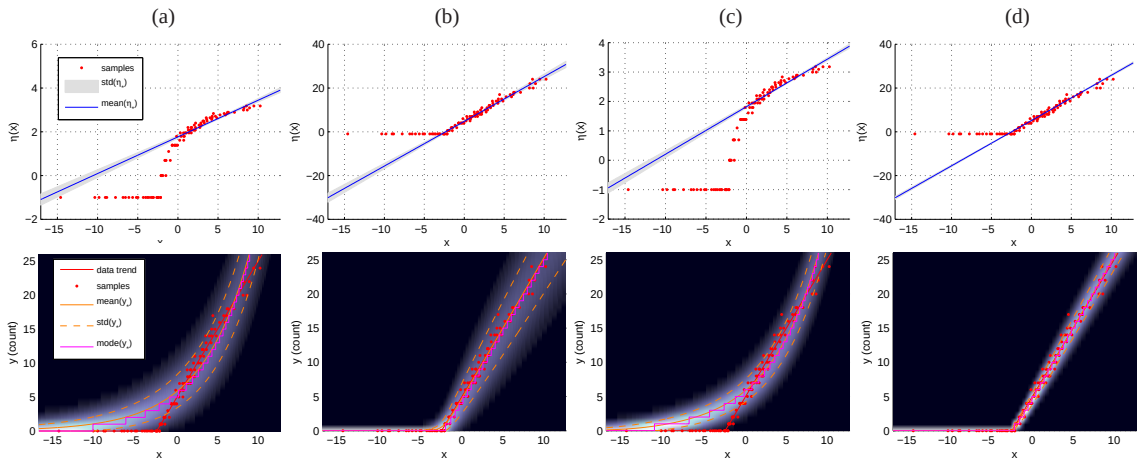

Figure 2. Examples of GGPM count regression models using different likelihood functions: a) Poisson; b) Linearized Poisson; c) COM-Poisson; d) Linearized COM-Poisson. The data follows a linear trend and is underdispersed. The top row shows the latent function learned in the latent space, while the bottom row shows the predictive distributions. The background color indicates the count probability (white most probable, black least probable)

Figures 2a and 2b illustrate the difference between the standard and linearized Poisson GGPMs. The standard Poisson-GGPM cannot correctly model the linear trend, resulting in a poor data fit at the extremes, while the linearized Poisson follows the linear trend.

6.2.2 Inference by Taylor approximation

Noting that $\eta = \log(\mathbb{E}[y])$, a reasonable choice of expansion point is $\tilde{\eta}_i = \log(y_i + c)$, where $c \geq 0$ is a constant to prevent taking the logarithm of zero, and hence

$$t_i = \log(y_i + c) - \frac{c}{y_i + c}, \quad w_i = \frac{1}{y_i + c}. \tag{35}$$

For $c = 0$, the Taylor approximation is exactly the closed-form approximation proposed for Bayesian Poisson regression in [8], which was derived in a different way using a log-gamma approximation.

6.3. Conway-Maxwell-Poisson distribution

One limitation with the Poisson distribution is that it models an equidispersed random variable, i.e. the variance is equal to the mean. However, in some cases, the actual random variable is *overdispersed* (with variance greater than the mean) or *underdispersed* (with variance less than the mean). An alternative distribution for count data, which represents different dispersion levels, is the Conway-Maxwell-Poisson (COM-Poisson) distribution [41–43],

$$p(y|\mu,\nu) = \frac{1}{S(\mu,\nu)} \left[\frac{\mu^y}{y!}\right]^\nu, \quad S(\mu,\nu) = \sum_{n=0}^{\infty} \left[\frac{\mu^n}{n!}\right]^\nu,$$

where $y \in \mathbb{Z}_+$, μ is (roughly) the mean parameter, and ν is the dispersion parameter. The COM-Poisson is a smooth interpolation between three distributions: geometric ($\nu = 0$), Poisson ($\nu = 1$), and Bernoulli ($\nu \to \infty$). The distribution is overdispersed for $\nu < 1$, and underdispersed for $\nu > 1$. Setting $\theta = \log \mu$ and $\phi = \nu$, we have

$$a(\phi) = \phi^{-1}, \; b_\phi(\theta) = \phi^{-1} \log S(e^\theta, \phi), \; h(y,\phi) = (y!)^{-\phi}.$$

Note that $b_\phi(\theta)$ is now also a function of ϕ (this only affects optimization of the dispersion ϕ (details in [39])). For the canonical link function, we set $\theta(\eta) = \eta$, and thus

$$\mathbb{E}[y] \approx e^\eta + \tfrac{1}{2\nu} - \tfrac{1}{2} = g^{-1}(\eta) \Rightarrow g(\mu) = \log(\mu - \tfrac{1}{2\nu} + \tfrac{1}{2}).$$

Alternatively the parameter function in (34) can be used to model a linear trend in the mean. The COM-Poisson GGPM includes a dispersion hyperparameter that decouples the variance of the Poisson from the mean, thus allowing more control on the observation noise of the output. Figures 2c and 2d show examples of using the COM-Poisson-GGPM on underdispersed counting data with a linear trend. Note that the variance of the prediction is much lower for the COM-Poisson models than for the Poisson models (Figures 2a and 2b), thus illustrating that the COM-Poisson GGPM can effectively estimate the dispersion of the data. A COM-Poisson GLM (with canonical link) was proposed in [43], and thus the COM-Poisson GGPM is a non-linear Bayesian extension using a GP prior on the latent function.

6.3.1 Counting experiments

We perform two counting experiments using GGPMs with Poisson-based likelihoods. In all cases, predictions are based on the mode of the distribution for GGPMs, and the rounded, truncated mean for GPR. In the first experiment, we perform crowd counting using the dataset from [7], and results are presented in Table 2. In all cases the compound linear-RBF kernel was used. On the "right" crowd, Poisson-GGPM performs the best (error 1.264), followed by the linearized Poisson (1.360). This is due to the large number of people in the "right" crowd, which leads to a more non-linear (exponential) trend in the feature space. On the other hand, the results on the "left" crowd show that the linearized COM-Poisson, linearized Poisson, and standard GPR all perform similarly, indicating a more linear trend in the data (due to smaller crowd sizes and fewer occlusions).

Method	Inference	MAE(R)	MAE(L)
Gauss	Exact	1.556	0.853
Poisson GGPM	Taylor	**1.264**	1.035
Poisson GGPM	Laplace	1.268	1.037
Poisson GGPM	EP	1.272	1.035
Linearized Poisson GGPM	Taylor	1.363	0.880
Linearized Poisson GGPM	Laplace	1.360	0.868
Linearized Poisson GGPM	EP	1.367	0.868
COM-Poisson GGPM	Taylor	1.432	1.053
COM-Poisson GGPM	Laplace	1.352	1.082
COM-Poisson GGPM	EP	1.429	1.048
Lin. COM-Poisson GGPM	Taylor	1.530	0.908
Lin. COM-Poisson GGPM	Laplace	1.523	**0.839**
Lin. COM-Poisson GGPM	EP	1.579	0.862

Table 2. Mean absolute errors for crowd counting.

In the second experiment, the GGPM is used for age estimation on the FG-NET dataset [44], where 150 facial features are extracted using active appearance models [45]. Our results are presented in Table 3, indicating that the Poisson GGPM with linearized mean performs the best among the models, with an mean absolute error of 5.824 versus 6.123 for standard GPR. Examples appear in Figure 3.

Method	Inference	MAE
GP	Exact	6.123
Warped GP [4]	Exact	6.111
Poisson GGPM	Taylor	6.444
Linearized Poisson GGPM	Taylor	5.975
Linearized Poisson GGPM	Laplace	**5.824**

Table 3. Mean absolute error for age estimation on FG-NET.

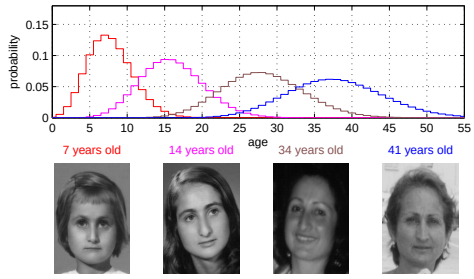

Figure 3. Examples of predicted age distributions on FG-NET.

Acknowledgements

This work was funded by CityU Hong Kong Grant 7200187. The authors thank CE Rasmussen and CKI Williams for the GPML code [1].

References

[1] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006.
[2] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell, "Gaussian processes for object categorization," *IJCV*, vol. 88, pp. 169–199, 2010.
[3] D. Han, L. Bo, and C. Sminchisescu, "Selection and context for action recognition," in *ICCV*, pp. 1933 –1940, sep. 2009.
[4] Y. Zhang and D.-Y. Yeung, "Multi-task warped gaussian process for personalized age estimation," in *CVPR*, 2010.
[5] B. Noris, K. Benmachiche, and A. G. Billard, "Calibration-free eye gaze direction detection with gaussian processes," in *VISAPP*, 2008.
[6] L. Raskin, M. Rudzsky, and E. Rivlin, "Tracking and classifying of human motions with gp annealed particle filter," in *ACCV*, 2007.
[7] A. B. Chan, Z. S. J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *CVPR*, 2008.
[8] A. B. Chan and N. Vasconcelos, "Bayesian Poisson regression for crowd counting," in *ICCV*, 2009.
[9] D. Ellis, E. Sommerlade, and I. Reid, "Modelling pedestrian trajectory patterns with gaussian processes," in *ICCV Workshops*, 2009.
[10] C. C. Loy, T. Xiang, and S. Gong, "Modelling multi-object activity by gaussian processes," in *BMVC*, 2009.
[11] O. Williams, "A switched gaussian process for estimating disparity and segmentation in binocular stereo," in *NIPS*, 2006.
[12] F. Sinz, Q. Candela, G. Bakir, C. Rasmussen, and M. Franz, "Learning depth from stereo," in *DAGM Symposium*, 2004.
[13] C. Plagemann, K. Kersting, P. Pfaff, and W. Burgard, "Gaussian beam processes: A nonparametric bayesian measurement model for range finders," in *In Proc. of Robotics: Science and Systems*, 2007.
[14] J. Zhu, S. Hoi, and M. Lyu, "Nonrigid shape recovery by gaussian process regression," in *CVPR*, pp. 1319 –1326, jun. 2009.
[15] L. Bo and C. Sminchisescu, "Twin gaussian processes for structured prediction," *IJCV*, vol. 87, pp. 28–52, 2010.
[16] R. Urtasun and T. Darrell, "Sparse probabilistic regression for activity-independent human pose inference," in *CVPR*, 2008.
[17] M. Fergie and A. Galata, "Local gaussian processes for pose recognition from noisy inputs," in *BMVC*, 2010.
[18] X. Zhao, H. Ning, Y. Liu, and T. Huang, "Discriminative estimation of 3d human pose using gaussian processes," in *ICPR*, 2008.
[19] R. Urtasun, D. Fleet, A. Hertzmann, and P. Fua, "Priors for people tracking from small training sets," in *ICCV*, 2005.
[20] J. Wang, D. Fleet, and A. Hertzmann, "Gaussian process dynamical models for human motion," *TPAMI*, vol. 30.2, pp. 283–98, 2008.
[21] J. Chen, M. Kim, Y. Wang, and Q. Ji, "Switching gaussian process dynamic models for simultaneous composite motion tracking and recognition," in *CVPR*, 2009.
[22] H. Nickisch and C. E. Rasmussen, "Approximations for binary gaussian process classification," *JMLR*, pp. 2035–78, 2008.
[23] M. Gibbs and D. J. C. Mackay, "Variational gaussian process classifiers," *IEEE TNN*, vol. 11, pp. 1458–1464, 1997.
[24] C. K. I. Williams and D. Barber, "Bayesian classification with gaussian processes," *IEEE TPAMI*, vol. 20, no. 12, pp. 1342–51, 1998.
[25] T. Minka, *A family of algorithms for approximate Bayesian inference*. PhD thesis, Massachusetts Institute of Technology, 2001.
[26] W. Chu and Z. Ghahramani, "Gaussian processes for ordinal regression," *JMLR*, pp. 1–48, 2005.
[27] M. Girolami and S. Rogers, "Variational bayesian multinomial probit regression with gaussian process priors," *Neur. Comp.*, vol. 18, 2005.
[28] M. Opper and C. Archambeau, "The variational gaussian approximation revisited," *Neur. Comp.*, vol. 21, pp. 786–92, March 2009.
[29] P. J. Diggle, J. A. Tawn, and R. A. Moyeed, "Model-based geostatistics," *Applied Statistics*, vol. 47, no. 3, pp. 299–350, 1998.
[30] A. Vehtari and J. Vanhatalo, "Sparse log gaussian processes via mcmc for spatial epidemiology," in *Wshop on GP in Practice*, 2007.
[31] R. Duda, P. Hart, and D. Stork, *Pattern Classification*. 2001.
[32] P. McCullagh and J. Nelder, *Generalized linear models*. 1999.
[33] M. Seeger, S. Gerwinn, and M. Bethge, "Bayesian inference for sparse generalized linear models," in *ECML*, 2007.
[34] H. Nickisch and M. W. Seeger, "Convex variational bayesian inference for large scale generalized linear models," in *ICML*, 2009.
[35] Z. Zhang, G. Dai, D. Wang, and M. I. Jordan, "Bayesian generalized kernel models," in *AISTATS*, vol. 9, 2010.
[36] G. C. Cawley, G. J. Janacek, and N. L. C. Talbot, "Generalised kernel machines," in *Intl. Joint Conf. on Neural Networks*, 2007.
[37] M. Seeger, "Gaussian processes for machine learning," *International Journal of Neural Systems* ., vol. 14, no. 2, pp. 69–106, 2004.
[38] V. Tresp, "The generalized bayesian committee machine," in *KDDM*.
[39] A. B. Chan and D. Dong, "Derivations for generalized gaussian process models," tech. rep., City University of Hong Kong, 2011.
[40] A. B. Chan and N. Vasconcelos, "Probabilistic kernels for the classification of auto-regressive visual processes," in *CVPR*, 2005.
[41] R. W. Conway and W. L. Maxwell, "A queuing model with state dependent service rates," *J. Industrial Eng.*, vol. 12, pp. 132–6, 1962.
[42] G. Shmueli, T. Minka, J. Kadane, S. Borle, and P. Boatwright, "A useful distribution for fitting discrete data: revival of the conway-maxwell-poisson distribution," *J. of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 54.1, pp. 127–142, 2005.
[43] S. Guikema and J. Coffelt, "A flexible count data regression model for risk analysis," *Risk Analysis*, vol. 28.1, pp. 213–223, 2008.
[44] "`http://www.fgnet.rsunit.com`."
[45] T.F.Cootes, G. Edwards, and C.J.Taylor., "Active appearance models," *IEEE TPAMI*, vol. 23.6, pp. 681–685, 2001.