Noname manuscript No.
(will be inserted by the editor)

Varying-Coefficient Models for Geospatial Transfer Learning

Matthias Bussas[1] · **Christoph Sawade**[2] · **Nicolas Kühn**[3] · **Tobias Scheffer**[4] · **Niels Landwehr**[4]

the date of receipt and acceptance should be inserted later

Abstract We study prediction problems in which the conditional distribution of the output given the input varies as a function of task variables which, in our applications, represent space and time. In varying-coefficient models, the coefficients of this conditional are allowed to change smoothly in space and time; the strength of the correlations between neighboring points is determined by the data. This is achieved by placing a Gaussian process (GP) prior on the coefficients. Bayesian inference in varying-coefficient models is generally intractable. We show that with an isotropic GP prior, inference in varying-coefficient models resolves to standard inference for a GP that can be solved efficiently. MAP inference in this model resolves to multitask learning using task and instance kernels. We clarify the relationship between varying-coefficient models and the hierarchical Bayesian multitask model and show that inference for hierarchical Bayesian multitask models can be carried out efficiently using graph-Laplacian kernels. We explore the model empirically for the problems of predicting rent and real-estate prices, and predicting the ground motion during seismic events. We find that varying-coefficient models with GP priors excel at predicting rents and real-estate prices. The ground-motion model predicts seismic hazards in the State of California more accurately than the previous state of the art.

1 Introduction

In standard settings of learning from independent and identically distributed *(iid)* data, labels y of training and test instances $\mathbf{x}$ are drawn independently and are governed by a fixed conditional distribution $p(y|\mathbf{x})$. Problem settings that relax this assumption are widely referred to as *transfer learning*. We study a transfer-learning setting in which the conditional $p(y|\mathbf{x})$ is assumed to vary as a function

[1]University College London, Department of Statistical Modeling, matthias.bussas.14@ucl.ac.uk · [2]SoundCloud Ltd., christoph@soundcloud.com · [3]Berkeley University, Pacific Earthquake Engineering Research Center, kuehn@berkeley.edu · [4]University of Potsdam, Department of Computer Science, {scheffer, landwehr}@cs.uni-potsdam.de

of additional observable variables $\mathbf{t}$. The variables $\mathbf{t}$ can identify a specific domain that an observation was drawn from (as in *multitask learning*), or can be continuous attributes that describe, for instance, the time or location at which an observation was made (sometimes called *concept drift*). We focus on applications in which $\mathbf{t}$ represents a geographic location, or both a location and a point in time.

A natural model for this setting is to assume a conditional $p(y|\mathbf{x};\mathbf{w})$ with parameters $\mathbf{w}$ that vary with $\mathbf{t}$. Such models are known as *varying-coefficient models* (*e.g.,* Hastie and Tibshirani, 1993; Gelfand et al., 2003). In *iid* learning, it is common to assume an isotropic Gaussian prior $p(\mathbf{w})$ over model parameters. When the parameters vary as a function of a task variable $\mathbf{t}$, it is natural to instead assume a Gaussian-process (GP) prior over functions that map values of $\mathbf{t}$ to values of $\mathbf{w}$. A Gaussian process implements a prior $p(\boldsymbol{\omega})$ over functions $\boldsymbol{\omega} : \mathcal{T} \to \mathbb{R}^m$ that couple parameters $\mathbf{w} \in \mathbb{R}^m$ for different values of $\mathbf{t} \in \mathcal{T}$ and make it possible to generalize over different domains, time, or space. While this model allows to extend Bayesian inference naturally to a variety of transfer-learning problems, inference in these varying-coefficient models for large problems is often impractical: It involves Kronecker products that result in matrices of size $nm \times nm$, with n the number of instances and m the number of attributes of $\mathbf{x}$ (Gelfand et al., 2003; Wheeler and Calder, 2006).

Alternatively, varying-coefficient models can be derived in a regularized risk-minimization framework. Such models infer point estimates of parameters $\mathbf{w}$ for different observed values of $\mathbf{t}$ under some model that expresses how $\mathbf{w}$ changes smoothly with $\mathbf{t}$. At test time, point estimates of $\mathbf{w}$ are required for all $\mathbf{t}$ observed at the test data points. This is again computationally challenging because typically a separate optimization problem needs to be solved for each test instance. Most prominent are estimation techniques based on kernel-local smoothing (Fan and Zhang, 2008; Wu and Chiang, 2000; Fan and Huang, 2005).

Logistic and ridge regression, among other discriminative models for *iid* data, rely on an isotropic Gaussian prior $p(\mathbf{w})$. By this assumption, $p(\mathbf{w})$ is a product of Gaussians; taking the logarithm results in the standard ℓ_2 regularization term $\mathbf{w}^\top \mathbf{w}$. However, since discriminative models do not involve the likelihood $p(\mathbf{x}|y)$ of the input variables, an isotropy assumption on $\mathbf{w}$ does not amount to the assumption that the dimensions of $\mathbf{x}$ are independent. In analogy, we explore Bayesian varying-coefficient models in conjunction with isotropic GP priors. Our main theoretical result is that Bayesian inference in varying-coefficient models with isotropic GP priors is equal to Bayesian inference in a standard Gaussian process with a specific product kernel. The main practical implication of this result is that inference for varying-coefficient models becomes practical by using standard GP tools.

Our theoretical result also leads to insights regarding existing transfer learning methods: First, we identify the exact modeling assumptions under which Bayesian inference amounts to multitask learning using a Gaussian process with task kernels and instance kernels (Bonilla et al., 2007). Secondly, we show that hierarchical Bayesian multitask models (*e.g.,* Gelman et al., 1995; Finkel and Manning, 2009) can be represented as Gaussian process priors; inference then resolves to inference in standard Gaussian processes with multitask kernels based on graph Laplacians (Evgeniou et al., 2005; Álvarez et al., 2011).

Predicting real-estate prices is an economically relevant problem that has previously been addressed using varying-coefficient models (Gelfand et al., 2003). Due to both the limited scalability of known inference methods for varying-coefficient

models and limited availability of real-estate transaction data, previous studies have been carried out on small-scale data collections. Substantial amounts of real-estate transactions have been disclosed in the course of recent open data initiatives. We compile a data set of real-estate transaction from the State of New York and rent prices from the States of New York and California and explore varying-coeffient models on this data set.

In probabilistic seismic-hazard analysis, a ground-motion model estimates the expected future distribution of a ground-motion parameter of interest at a specific site, depending on the event's origin and site-related parameters such as magnitude and distance. A typical ground-motion parameter is the *peak ground acceleration* that a site experiences during a seismic event. Accurate ground-motion models are important to establish building codes and determine insurance risks. Traditional ground-motion models are ergodic; that is, the conditional distribution of the ground-motion parameter of interest at a given site is identical to the conditional distribution at any other site, given the same magnitude, distance, and site conditions (Anderson and Brune, 1999). These ergodic models compete with specialized regional models—*e.g.,* for Greece (Danciu and Tselentis, 2007), Italy (Bindi et al., 2011), the Eastern Alps (Bragato and Slejko, 2005), and Turkey (Akkar and Cagnan, 2010). These regional models suffer from smaller numbers of data points.

In order to weaken the assumption of ergodicity in ground-motion models, Gianniotis et al. (2014) and Stafford (2014) estimate ground-motion models from a larger data set and constrain the coefficients to be similar across each region. Regional adjustments can be broken down into smaller geographical units (Al-Atik et al., 2010; Lin et al., 2011). This approach, however, relies on the availability of sufficiently many observations in each geographical compartment.

Our theoretical findings allow us to derive a ground-motion model in which the coefficients of the model can vary smoothly with geographical location and for which inference is computationally tractable. The model is developed and evaluated on a subset of the NGA West 2 dataset (Ancheta et al., 2014), based on Californian data used by Abrahamson et al. (2014). In California, regional differences between Northern California and Southern California have been found previously (Atkinson and Morrison, 2009; Chiou et al., 2010), though the recent NGA West 2 models treat California as a whole.

The rest of this paper is structured as follows. Section 2 describes the problem setting and the varying-coefficient model. Section 3 studies Bayesian inference and presents our main results. Section 4 presents experiments on prediction of real estate prices and seismic-hazard analysis; Section 5 discusses related work and concludes.

2 Problem Setting and Model

In multi-task learning, one or several of the available observable variables are singled out and treated as *task* variables $\mathbf{t}$—for instance, the identity of a speaker in speech recognition or the location in a geospatial prediction problem. This modeling decision reflects the assumption that $p(y|\mathbf{x}, \mathbf{t}, \mathbf{w})$ is similar across the values of $\mathbf{t}$. If the underlying application satisfies this assumption, it allows for better predictions; for instance, for values of $\mathbf{t}$ that are poorly covered by the training

data. This section describes a stochastic process that models applications which are characterized by a conditional distribution $p(y|\mathbf{x}, \boldsymbol{\omega}(\mathbf{t}))$ whose parameterization $\boldsymbol{\omega}(\mathbf{t})$ varies smoothly as a function of $\mathbf{t}$.

A fixed set of instances $\mathbf{x}_1, \ldots, \mathbf{x}_n$ with $\mathbf{x}_i \in \mathcal{X} \subseteq \mathbb{R}^m$ is observable, along with values $\mathbf{t}_1, \ldots, \mathbf{t}_n \in \mathcal{T}$ of a *task variable*. The stochastic process starts by drawing a function $\boldsymbol{\omega} : \mathcal{T} \to \mathbb{R}^m$ according to a prior $p(\boldsymbol{\omega})$. The function $\boldsymbol{\omega}$ associates any task variable $\mathbf{t} \in \mathcal{T}$ with a corresponding parameter vector $\boldsymbol{\omega}(\mathbf{t}) \in \mathbb{R}^m$ that defines the conditional distribution $p(y|\mathbf{x}, \boldsymbol{\omega}(\mathbf{t}))$ for task $\mathbf{t} \in \mathcal{T}$. The domain $\mathcal{T}$ of the task variable depends on the application at hand. In the case of *multitask learning*, $\mathcal{T} = \{1, \ldots, k\}$ is a set of task identifiers. In hierarchical Bayesian multitask models, a tree $\mathcal{G} = (\mathcal{T}, \mathbf{A})$ over the tasks $\mathcal{T} = \{1, \ldots, k\}$ reflects how tasks are related; we represent this tree by its adjacency matrix $\mathbf{A} \in \mathbb{R}^{k \times k}$. We focus on geospatial transfer-learning problems in which the conditional distribution of y given $\mathbf{x}$ varies smoothly in the task variables $\mathbf{t}$ that represent spatial coordinates, or both space and time. In this case, $\mathcal{T} \subset \mathbb{R}^d$ is a continuous-valued space.

We model $p(\boldsymbol{\omega})$ using a zero-mean Gaussian process

$$\boldsymbol{\omega} \sim \mathcal{GP}(\mathbf{0}, \boldsymbol{\kappa}) \tag{1}$$

that generates vector-valued functions $\boldsymbol{\omega} : \mathcal{T} \to \mathbb{R}^m$. The process is specified by a matrix-valued kernel function $\boldsymbol{\kappa} : \mathcal{T} \times \mathcal{T} \to \mathbb{R}^{m \times m}$ that reflects closeness in $\mathcal{T}$. Here, $\boldsymbol{\kappa}(\mathbf{t}, \mathbf{t}') \in \mathbb{R}^{m \times m}$ is the matrix of covariances between dimensions of the vectors $\boldsymbol{\omega}(\mathbf{t})$ and $\boldsymbol{\omega}(\mathbf{t}')$.

In discriminative machine-learning models, one usually assumes that the dimensions of model-parameter vector $\mathbf{w}$ are generated independently of one another. For instance, one often assumes an isotropic Gaussian prior $p(\mathbf{w}) = \prod_{j=1}^{m} \mathcal{N}[0, \sigma^2](w_j)$; the negative log-posterior then resolves to an ℓ_2-regularized loss function. In analogy, we assume that the dimensions of $\boldsymbol{\omega}$ are generated by independent Gaussian processes; that is, $\omega_j \sim \mathcal{GP}(\mathbf{0}, k_j)$, where $k_j(\mathbf{t}, \mathbf{t}')$ is the covariance between coefficients $\omega_j(\mathbf{t})$ and $\omega_j(\mathbf{t}')$. A special case of this is an isotropic GP; here, kernel functions $k_j(\mathbf{t}, \mathbf{t}')$ are identical for all dimensions j. Therefore, for all $\mathbf{t}, \mathbf{t}' \in \mathcal{T}$, covariance matrix $\boldsymbol{\kappa}(\mathbf{t}, \mathbf{t}')$ is a diagonal matrix with diagonal elements $k_j(\mathbf{t}, \mathbf{t}')$, where the $k_j : \mathcal{T} \times \mathcal{T} \to \mathbb{R}$ are scalar-valued, positive semidefinite kernel functions. Note that we do not make any assumptions about $p(\mathbf{x}|y, \boldsymbol{\omega}(\mathbf{t}))$, and the independence of the dimensions of $\boldsymbol{\omega}$ does not imply that the dimensions of the input vector $\mathbf{x}$ are independent. Also note that isotropy of $p(\boldsymbol{\omega}(\mathbf{t}))$ is different from the assumption of an *isotropic kernel*, meaning a kernel that is uniform in all directions of the input space.

The process evaluates function $\boldsymbol{\omega}$ for all $\mathbf{t}_i$ to create parameter vectors $\mathbf{w}_1 = \boldsymbol{\omega}(\mathbf{t}_1), \ldots, \mathbf{w}_n = \boldsymbol{\omega}(\mathbf{t}_n)$. The process concludes by generating labels y_i from an observation model,

$$y_i \sim p(y|\mathbf{x}_i, \mathbf{w}_i); \tag{2}$$

for instance, a standard linear model with Gaussian noise for regression or a logistic function of the inner product of $\mathbf{w}_i$ and $\mathbf{x}_i$ for classification.

The prediction problem is to infer the distribution of the label $y_\star$ for a new observation $\mathbf{x}_\star$ with task variable $\mathbf{t}_\star$. For notational convenience, we aggregate the training instances into matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$, task variables into matrix $\mathbf{T} \in \mathbb{R}^{n \times d}$, the parameter vectors associated with training observations into matrix $\mathbf{W} \in \mathbb{R}^{n \times m}$ with row vectors $\mathbf{w}_1^\mathsf{T}, \ldots, \mathbf{w}_n^\mathsf{T}$, and the labels $y_1, \ldots, y_n$ into vector $\mathbf{y} \in \mathcal{Y}^n$.

In this model, the GP prior $p(\boldsymbol{\omega})$ over functions $\boldsymbol{\omega} : \mathcal{T} \to \mathbb{R}^m$ couples parameter vectors $\boldsymbol{\omega}(\mathbf{t})$ for different values $\mathbf{t}$ of the task variable. The hierarchical Bayesian model of multitask learning assumes a coupling of parameters based on a hierarchical Bayesian prior (*e.g.,* Gelman et al., 1995; Finkel and Manning, 2009). We will now show that the varying-coefficient model with isotropic GP prior subsumes hierarchical Bayesian multitask models by choice of an appropriate kernel function $\boldsymbol{\kappa}$ of the Gaussian process that defines $p(\boldsymbol{\omega})$. Together with results on inference presented in Section 3, this result shows how inference for hierarchical Bayesian multitask models can be carried out using a Gaussian process. The following definition formalizes the hierarchical Bayesian multitask model.

Definition 1 (Hierarchical Bayesian Multitask Model) Let $\mathcal{G} = (\mathcal{T}, \mathbf{A})$ denote a tree structure over a set of tasks $\mathcal{T} = \{1, \ldots, k\}$ given by an adjacency matrix $\mathbf{A}$, with $1 \in \mathcal{T}$ the root node. Let $\boldsymbol{\sigma} \in \mathbb{R}^k$ denote a vector with entries $\sigma_1, \ldots, \sigma_k$. The following process generates the distribution $p(\mathbf{y}|\mathbf{X}, \mathbf{T}; \mathcal{G}, \boldsymbol{\sigma})$ over labels $\mathbf{y} \in \mathcal{Y}^n$ given instances $\mathbf{X}$, task variables $\mathbf{T}$, the task hierarchy $\mathcal{G}$, and variances $\boldsymbol{\sigma}$: The process first samples parameter vectors $\bar{\mathbf{w}}_1, \ldots, \bar{\mathbf{w}}_k \in \mathbb{R}^m$ according to

$$\bar{\mathbf{w}}_1 \sim \mathcal{N}(\bar{\mathbf{w}}|\mathbf{0}, \sigma_1^2 \mathbf{I}_{m \times m}) \tag{3}$$

$$\bar{\mathbf{w}}_l \sim \mathcal{N}(\bar{\mathbf{w}}|\bar{\mathbf{w}}_{pa(l)}, \sigma_l^2 \mathbf{I}_{m \times m}) \qquad 2 \leq l \leq k \tag{4}$$

where $pa(l) \in \mathcal{T}$ is a unique node with $\mathbf{A}_{pa(l),l} = 1$ for each $l \in \mathcal{T}$. Then, the process generates labels $y_i \sim p(y|\mathbf{x}_i, \bar{\mathbf{w}}_i)$, where $p(y|\mathbf{x}_i, \bar{\mathbf{w}}_i)$ is the same conditional distribution over labels given an instance and a parameter vector as was chosen for the varying-coefficient model in Equation 2. This process defines the *hierarchical Bayesian multitask model.*

The following proposition shows that the varying-coefficient model presented in Section 2 subsumes the hierarchical Bayesian multitask model.

Proposition 1 *Let $\mathcal{G} = (\mathcal{T}, \mathbf{A})$ denote a tree structure over a set of $\mathcal{T} = \{1, \ldots, k\}$ given by an adjacency matrix $\mathbf{A}$. Let $\boldsymbol{\sigma} \in \mathbb{R}^k$ be a vector with entries $\sigma_1, \ldots, \sigma_k$. Let $k_{\mathbf{A},\boldsymbol{\sigma}} : \mathcal{T} \times \mathcal{T} \to \mathbb{R}$ be given by $k_{\mathbf{A},\boldsymbol{\sigma}}(t, t') = G_{t,t'}$, with $G_{i,j}$ the entry at row i and column j of the matrix*

$$\mathbf{G} = (\mathbf{I}_{k \times k} - \mathbf{A})^{-1} \mathbf{S} \left(\mathbf{I}_{k \times k} - \mathbf{A}^\mathsf{T} \right)^{-1},$$

and $\mathbf{S} \in \mathbb{R}^{k \times k}$ denotes the diagonal matrix with entries $\sigma_1^2, \ldots, \sigma_k^2$. Let $\boldsymbol{\kappa} : \mathcal{T} \times \mathcal{T} \to \mathbb{R}^{m \times m}$ be given by $\boldsymbol{\kappa}(t, t') = k_{\mathbf{A},\boldsymbol{\sigma}}(t, t') \mathbf{I}_{m \times m}$ and let $p(\mathbf{y}|\mathbf{X}, \mathbf{T}; \boldsymbol{\kappa}) = \int p(\mathbf{y}|\mathbf{W}, \mathbf{X}) p(\mathbf{W}|\mathbf{T}; \boldsymbol{\kappa}) \mathrm{d}\mathbf{W}$ be the marginal distribution over labels given instances and task variables defined by the varying-coefficient model. Then it holds that $p(\mathbf{y}|\mathbf{X}, \mathbf{T}; \boldsymbol{\kappa}) = p(\mathbf{y}|\mathbf{X}, \mathbf{T}; \mathcal{G}, \boldsymbol{\sigma})$.

Proposition 1 implies that Bayesian prediction in the varying-coefficient model with the specified kernel function is identical to Bayesian inference in the hierarchical Bayesian multitask model. The proof is included in the appendix. In Proposition 1, entries $G_{t,t'}$ of $\mathbf{G}$ represent a task similarity derived from the tree structure $\mathcal{G}$. Instead of a tree structure over tasks, feature vectors describing individual tasks may also be given (Bonilla et al., 2007; Yan and Zhang, 2009). In this case, $\boldsymbol{\kappa}(t, t')$ can be computed from the task features; the varying-coefficient model

then subsumes existing approaches for multitask learning with task features (see Section 3.4).

Note that Equation 3 of Daumé III (2009) is a special case of our model for learning with two tasks and a task kernel $\boldsymbol{\kappa}$ that is $\boldsymbol{\kappa}(t,t) = 2\mathbf{I}_{m\times m}$ for identical tasks and $\boldsymbol{\kappa}(t,t') = \mathbf{I}_{m\times m}$ for differing tasks $t \neq t'$.

3 Inference

We now address the problem of inferring predictions $y_\star$ for instances $\mathbf{x}_\star$ and task variables $\mathbf{t}_\star$. Section 3.1 presents exact Bayesian solutions for regression; Section 3.2 discusses approximate inference for classification. Section 3.4 derives existing multitask models as special cases.

3.1 Regression

This subsection studies linear regression models of the form $p(y|\mathbf{x},\mathbf{w}) = \mathcal{N}(y|\mathbf{x}^\mathsf{T}\mathbf{w},\tau^2)$. Note that by substituting for the slightly heavier notation $p(y|\mathbf{x},\mathbf{w}) = \mathcal{N}(y|\Phi(\mathbf{x})^\mathsf{T}\mathbf{w},\tau^2)$, this treatment also covers finite-dimensional feature maps. The predictive distribution for test instance $\mathbf{x}_\star$ with task variable $\mathbf{t}_\star$ is obtained by integrating over the possible parameter values $\mathbf{w}_\star$ of the conditional distribution that has generated value $y_\star$:

$$p(y_\star|\mathbf{X},\mathbf{y},\mathbf{T},\mathbf{x}_\star,\mathbf{t}_\star) = \int p(y_\star|\mathbf{x}_\star,\mathbf{w}_\star)p(\mathbf{w}_\star|\mathbf{X},\mathbf{y},\mathbf{T},\mathbf{t}_\star)\mathrm{d}\mathbf{w}_\star, \tag{5}$$

where the posterior over $\mathbf{w}_\star$ is obtained by integrating over the joint parameter values $\mathbf{W}$ that have generated the labels $\mathbf{y}$ for instances $\mathbf{X}$ and task variables $\mathbf{T}$:

$$p(\mathbf{w}_\star|\mathbf{X},\mathbf{y},\mathbf{T},\mathbf{t}_\star) = \int p(\mathbf{w}_\star|\mathbf{W},\mathbf{T},\mathbf{t}_\star)p(\mathbf{W}|\mathbf{X},\mathbf{y},\mathbf{T})\mathrm{d}\mathbf{W}. \tag{6}$$

Posterior distribution $p(\mathbf{W}|\mathbf{X},\mathbf{y},\mathbf{T})$ in Equation 6 depends on the likelihood function—the linear model—and the GP prior $p(\boldsymbol{\omega})$. The extrapolated posterior $p(\mathbf{w}_\star|\mathbf{W},\mathbf{T},\mathbf{t}_\star)$ for test instance $\mathbf{x}_\star$ with task variable $\mathbf{t}_\star$ depends on the Gaussian process. The following theorem states how the predictive distribution given by Equation 5 can be computed.

Theorem 1 (Bayesian Predictive Distribution) *Let $\mathcal{Y} = \mathbb{R}$ and $p(y|\mathbf{x},\mathbf{w}) = \mathcal{N}(y|\mathbf{x}^\mathsf{T}\mathbf{w},\tau^2)$. For all attributes $j \in \{1,\ldots,m\}$, let $k_j(\mathbf{t},\mathbf{t}')$ a positive definite kernel function and let the task-kernel function $\boldsymbol{\kappa}(\mathbf{t},\mathbf{t}')$ return a diagonal matrix with diagonal elements $k_j(\mathbf{t},\mathbf{t}')$. Let $\mathbf{K} \in \mathbb{R}^{n\times n}$ be a matrix with components $k_{ij} = \mathbf{x}_i^\mathsf{T}\boldsymbol{\kappa}(\mathbf{t}_i,\mathbf{t}_j)\mathbf{x}_j$ and $\mathbf{k} \in \mathbb{R}^n$ be a vector with components $k_i = \mathbf{x}_i^\mathsf{T}\boldsymbol{\kappa}(\mathbf{t}_i,\mathbf{t}_\star)\mathbf{x}_\star$. Then, the predictive distribution for the varying-coefficient model defined in Section 2 is given by*

$$p(y_\star|\mathbf{X},\mathbf{y},\mathbf{T},\mathbf{x}_\star,\mathbf{t}_\star) = p(y_\star|\mathbf{K},\mathbf{k},\mathbf{y},\mathbf{x}_\star,\mathbf{t}_\star) = \mathcal{N}(y_\star|\mu,\sigma^2+\tau^2) \tag{7}$$

$$\begin{aligned} \textit{with} \qquad \mu &= \mathbf{k}^\mathsf{T}(\mathbf{K}+\tau^2\mathbf{I}_{n\times n})^{-1}\mathbf{y}, \\ \sigma^2 &= \mathbf{x}_\star^\mathsf{T}\boldsymbol{\kappa}(\mathbf{t}_\star,\mathbf{t}_\star)\mathbf{x}_\star - \mathbf{k}^\mathsf{T}(\mathbf{K}+\tau^2\mathbf{I}_{n\times n})^{-1}\mathbf{k}. \end{aligned}$$

Before we prove Theorem 1, we highlight three observations about this result. First, the distribution $p(y_\star|\mathbf{X}, \mathbf{y}, \mathbf{T}, \mathbf{x}_\star, \mathbf{t}_\star)$ has a surprisingly simple form. It is identical to the predictive distribution of a standard Gaussian process that uses concatenated vectors $(\mathbf{x}_1, \mathbf{t}_1), \ldots, (\mathbf{x}_n, \mathbf{t}_n) \in \mathcal{X} \times \mathcal{T}$ as training instances, labels $y_1, \ldots, y_n$, and the kernel function $k((\mathbf{x}_i, \mathbf{t}_i), (\mathbf{x}_j, \mathbf{t}_j)) = \mathbf{x}_i^\mathsf{T} \boldsymbol{\kappa}(\mathbf{t}_i, \mathbf{t}_j)\mathbf{x}_j$. Covariance matrix $\boldsymbol{\kappa}(\mathbf{t}_i, \mathbf{t}_j)$ is diagonal; when the GP prior is isotropic, then all diagonal elements are identical and we refer to their value as $k(\mathbf{t}_i, \mathbf{t}_j)$. We can see that in this case, the predictive distribution of Equation 7 is identical to the predictive distribution of a standard Gaussian process with concatenated vectors $(\mathbf{x}_i, \mathbf{t}_i)$ and product kernel function $k((\mathbf{x}_i, \mathbf{t}_i), (\mathbf{x}_j, \mathbf{t}_j)) = \mathbf{x}_i^\mathsf{T}\mathbf{x}_j k(\mathbf{t}_i, \mathbf{t}_j)$.

Secondly, when the GP is isotropic, then instances $\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_\star \in \mathcal{X}$ only enter Equation 7 in the form of inner products. The model can therefore directly be kernelized by defining the kernel matrix as $\mathbf{K}_{ij} = k_\mathcal{X}(\mathbf{x}_i, \mathbf{x}_j)k(\mathbf{t}_i, \mathbf{t}_j)$ with kernel function $k_\mathcal{X}(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i)^\mathsf{T}\Phi(\mathbf{x}_j)$ where Φ maps to a reproducing kernel Hilbert space. When the feature space is finite, then $\boldsymbol{\omega}$ maps the $\mathbf{t}_i$ to a finite-dimensional $\mathbf{w}_i$ and Theorem 1 implies a Bayesian predictive distribution derived from the generative process that Section 2 specifies. When the reproducing kernel Hilbert space does not have a finite dimension, Section 2 does no longer specify a corresponding proper stochastic process because $p(\mathbf{w}_1, \ldots, \mathbf{w}_n|\mathbf{T})$ would become infinite-dimensionally normally distributed. However, given the finite sample $\mathbf{X}$ and $\mathbf{T}$, a Mercer map (see, *e.g.,* Schölkopf and Smola, 2002, Section 2.2.4) constitutes a finite-dimensional space $\mathbb{R}^n$ for which Section 2 again characterizes a corresponding stochastic process.

Thirdly and finally, Theorem 1 shows how Bayesian inference in varying-coefficient models with isotropic priors can be implemented efficiently. For general varying-coefficient models, the most expensive step of inference is to perform, for each sample generated by Gibbs sampling, a Cholesky decomposition of a matrix of size $mn \times mn$ (discussed above Equation 17 of Gelfand et al., 2003). A sampled parameter vector $\hat{\omega}(\mathbf{t}_1) \ldots \hat{\omega}(\mathbf{t}_n)$ is of size nm. In each sampling step, a Cholesky decomposition of the covariance matrix (which then has size $nm \times nm$) of parameter vectors has to be performed. This makes inference impractical for large-scale problems. Theorem 1 shows that under the assumption of an isotropic prior—or at least an independent prior distribution for each dimension—the latent parameter vectors $\mathbf{w}_1, \ldots, \mathbf{w}_n$ can be integrated out, which results in a GP formulation in which the covariance structure over parameter vectors resolves to an $n \times n$ product-kernel matrix.

Proof (Proof of Theorem 1) Let w_{ir} and $w_{\star r}$ denote the r-th elements of vectors $\mathbf{w}_i$ and $\mathbf{w}_\star$, and let x_{ir} and $x_{\star r}$ denote the r-th elements of vectors $\mathbf{x}_i$ and $\mathbf{x}_\star$. Let $\mathbf{z}_\star = (z_1, \ldots, z_n, z_\star)^\mathsf{T} \in \mathbb{R}^{n+1}$ with $z_i = \mathbf{x}_i^\mathsf{T}\mathbf{w}_i$ and $z_\star = \mathbf{x}_\star^\mathsf{T}\mathbf{w}_\star$. Because $\mathbf{w}_1, \ldots, \mathbf{w}_n, \mathbf{w}_\star$ are evaluations of the function $\boldsymbol{\omega}$ drawn from a Gaussian process (Equation 1), they are jointly Gaussian distributed and thus $z_1, \ldots, z_n, z_\star$ are also jointly Gaussian (*e.g.,* Murphy, 2012, Chapter 10.2.5). Because $\boldsymbol{\omega}$ is drawn from a zero-mean process, it holds that $\mathbb{E}[z_i] = \mathbb{E}[\sum_{r=1}^m x_{ir}w_{ir}] = \sum_{r=1}^m x_{ir}\mathbb{E}[w_{ir}] = 0$ as well as $\mathbb{E}[z_\star] = 0$ and therefore

$$p(\mathbf{z}_\star|\mathbf{X}, \mathbf{T}, \mathbf{x}_\star, \mathbf{t}_\star) = \mathcal{N}(\mathbf{z}_\star|\mathbf{0}, \mathbf{C})$$

where $\mathbf{C} \in \mathbb{R}^{(n+1)\times(n+1)}$ denotes the covariance matrix.

For the covariances $\mathbb{E}[z_i z_j]$ it holds that

$$\begin{aligned}
\mathbb{E}\left[z_i z_j\right] &= \mathbb{E}\left[\mathbf{x}_i^\mathsf{T}\mathbf{w}_i\mathbf{x}_j^\mathsf{T}\mathbf{w}_j\right] \\
&= \mathbb{E}\left[\left(\sum_{s=1}^{m} x_{is}w_{is}\right)\left(\sum_{r=1}^{m} x_{jr}w_{jr}\right)\right] \\
&= \sum_{s=1}^{m}\sum_{r=1}^{m} x_{is}x_{jr}\mathbb{E}\left[w_{is}w_{jr}\right] \\
&= \sum_{s=1}^{m} x_{is}x_{js}\mathbb{E}\left[w_{is}w_{js}\right] \qquad (8) \\
&= \mathbf{x}_i^\mathsf{T}\boldsymbol{\kappa}(\mathbf{t}_i,\mathbf{t}_j)\mathbf{x}_j. \qquad (9)
\end{aligned}$$

In Equations 8 and 9 we exploit the independence of the Gaussian process priors for all dimensions: the covariance $\mathbb{E}[w_{is}w_{jr}]$ is the element in row s and column r of the matrix $\boldsymbol{\kappa}(\mathbf{t}_i,\mathbf{t}_j) \in \mathbb{R}^{m\times m}$ obtained by evaluating the kernel function $\boldsymbol{\kappa} : \mathcal{T}\times\mathcal{T} \to \mathbb{R}^{m\times m}$ at $(\mathbf{t}_i,\mathbf{t}_j)$; by the independence assumption, $\boldsymbol{\kappa}(\mathbf{t}_i,\mathbf{t}_j)$ is a diagonal matrix and $\mathbb{E}[w_{is}w_{jr}] = 0$ for $s \neq r$ (see Section 2). We analogously derive

$$\mathbb{E}[z_i z_\star] = \mathbf{x}_i^\mathsf{T}\boldsymbol{\kappa}(\mathbf{t}_i,\mathbf{t}_\star)\mathbf{x}_\star, \qquad (10)$$

$$\mathbb{E}[z_\star z_\star] = \mathbf{x}_\star^\mathsf{T}\boldsymbol{\kappa}(\mathbf{t}_\star,\mathbf{t}_\star)\mathbf{x}_\star. \qquad (11)$$

Equations 9, 10 and 11 define the covariance matrix $\mathbf{C}$, yielding

$$p(\mathbf{z}_\star|\mathbf{X},\mathbf{T},\mathbf{x}_\star,\mathbf{t}_\star) = \mathcal{N}\left(\mathbf{z}_\star|\mathbf{0},\begin{pmatrix}\mathbf{K} & \mathbf{k}\\ \mathbf{k}^\mathsf{T} & k_\star\end{pmatrix}\right)$$

where $k_\star = \mathbf{x}_\star^\mathsf{T}\boldsymbol{\kappa}(\mathbf{t}_\star,\mathbf{t}_\star)\mathbf{x}_\star$. For $\mathbf{y}_\star = (y_1,\ldots,y_n,y_\star)$ it now follows that

$$p(\mathbf{y}_\star|\mathbf{X},\mathbf{T},\mathbf{x}_\star,\mathbf{t}_\star) = \mathcal{N}\left(\mathbf{y}_\star|\mathbf{0},\begin{pmatrix}\mathbf{K}+\tau^2\mathbf{I}_{n\times n} & \mathbf{k}\\ \mathbf{k}^\mathsf{T} & k_\star+\tau^2\end{pmatrix}\right). \qquad (12)$$

The claim now follows by applying standard Gaussian identities to compute the conditional distribution $p(y_\star|\mathbf{X},\mathbf{y},\mathbf{T},\mathbf{x}_\star,\mathbf{t}_\star)$ from Equation 12.

3.2 Classification

The result given by Theorem 1 can be extended to classification settings with $\mathcal{Y} = \{0,1\}$ by using non-Gaussian likelihoods $p(y|z)$ that generate labels $y \in \mathcal{Y}$ given outputs $z \in \mathbb{R}$ of the linear model.

Corollary 1 (Bayesian predictive distribution for non-Gaussian likelihoods) *Let $\mathcal{Y} = \{0,1\}$. Let $p(y_i|\mathbf{x}_i,\mathbf{w}_i)$ be given by a generalized linear model, defined by $z_i \sim \mathcal{N}(z|\mathbf{w}_i^\mathsf{T}\mathbf{x}_i,\tau^2)$ and $y_i \sim p(y|z_i)$. Let $p(y_\star|\mathbf{x}_\star,\mathbf{w}_\star)$ be given by $z_\star \sim \mathcal{N}(z|\mathbf{w}_\star^\mathsf{T}\mathbf{x}_\star,\tau^2)$ and $y_\star \sim p(y|z_\star)$. Let furthermore $\mathbf{z} = (z_1,\ldots,z_n)^\mathsf{T} \in \mathbb{R}^n$. For all attributes $j \in \{1,\ldots,m\}$, let $k_j(\mathbf{t},\mathbf{t}')$ a positive definite kernel function and let the task-kernel function $\boldsymbol{\kappa}(\mathbf{t},\mathbf{t}')$ return a diagonal matrix with diagonal elements $k_j(\mathbf{t},\mathbf{t}')$.*

Let $\mathbf{K} \in \mathbb{R}^{n \times n}$ *be a matrix with components* $k_{ij} = \mathbf{x}_i^\mathsf{T} \boldsymbol{\kappa}(\mathbf{t}_i, \mathbf{t}_j)\mathbf{x}_j$ *and* $\mathbf{k} \in \mathbb{R}^n$ *a vector with components* $k_i = \mathbf{x}_i^\mathsf{T} \boldsymbol{\kappa}(\mathbf{t}_i, \mathbf{t}_\star)\mathbf{x}_\star$. *Then, the predictive distribution for the GP model defined in Section 2 is given by*

$$p(y_\star|\mathbf{X}, \mathbf{y}, \mathbf{T}, \mathbf{x}_\star, \mathbf{t}_\star) = p(y_\star|\mathbf{K}, \mathbf{k}, \mathbf{y}, \mathbf{x}_\star, \mathbf{t}_\star)$$
$$\propto \iint p(y_\star|z_\star)\mathcal{N}(z_\star|\mu_\mathbf{z}, \sigma_\mathbf{z}^2)p(\mathbf{y}|\mathbf{z})\mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{K} + \tau^2\mathbf{I}_{n\times n})\mathrm{d}\mathbf{z}\mathrm{d}z_\star \quad (13)$$

with

$$\mu_\mathbf{z} = \mathbf{k}^\mathsf{T}(\mathbf{K} + \tau^2\mathbf{I}_{n\times n})^{-1}\mathbf{z},$$
$$\sigma_\mathbf{z}^2 = \mathbf{x}_\star^\mathsf{T}\boldsymbol{\kappa}(\mathbf{t}_\star, \mathbf{t}_\star)\mathbf{x}_\star - \mathbf{k}^\mathsf{T}(\mathbf{K} + \tau^2\mathbf{I}_{n\times n})^{-1}\mathbf{k} + \tau^2.$$

A straightforward calculation shows that Equation 13 is identical to the predictive distribution of a standard Gaussian process that uses concatenated vectors $(\mathbf{x}_1, \mathbf{t}_1), \ldots, (\mathbf{x}_n, \mathbf{t}_n) \in \mathcal{X} \times \mathcal{T}$ as training instances, labels $y_1, \ldots, y_n$, the kernel $k((\mathbf{x}_i, \mathbf{t}_i), (\mathbf{x}_j, \mathbf{t}_j)) = \mathbf{x}_i^\mathsf{T}\boldsymbol{\kappa}(\mathbf{t}_i, \mathbf{t}_j)\mathbf{x}_j$, and likelihood function $p(y|z)$. For isotropic GP priors, the kernel function is the product kernel $k((\mathbf{x}_i, \mathbf{t}_i), (\mathbf{x}_j, \mathbf{t}_j)) = \mathbf{x}_i^\mathsf{T}\mathbf{x}_j k(\mathbf{t}_i, \mathbf{t}_j)$. For non-Gaussian likelihoods, exact inference in Gaussian processes is generally intractable, but approximate inference methods based on, *e.g.*, Laplace approximation, variational inference or expectation propagation are available. The proof is included in the appendix.

3.3 Algorithm

We are now ready to summarize the inference procedure in Algorithm 1. The *isoVCM* algorithm summarizes two cases. In the *primal* case, instances $\mathbf{x}$ are represented by explicit features. In this case, task-kernel function $\boldsymbol{\kappa}(\mathbf{t}, \mathbf{t}')$ maps a pair of tasks to a diagonal covariance matrix whose diagonal elements $k_j(\mathbf{t}, \mathbf{t}')$ may differ over the features. This allows us to express background knowledge about differing variances of features. In the *dual* case, a kernel function $k_\mathcal{X}(\mathbf{x}, \mathbf{x}')$ is provided. In this case, an explicit feature representation could be constructed for any finite data set using a Mercer map. However, one then has no background knowledge about the variance of these artificially constructed features and the covariance matrices $\boldsymbol{\kappa}(\mathbf{t}, \mathbf{t}')$ are isotropic with diagonal elements $k(\mathbf{t}, \mathbf{t}')$. Algorithm 1 combines input matrix $\mathbf{X}$ and task variables $\mathbf{T}$ into an overall kernel matrix $\mathbf{K}$, and infers the distribution of target variable $y_\star$ for test input $\mathbf{x}_\star$ by standard inference in a GP with kernel matrix $\mathbf{K}$.

To implement Algorithm 1, we use the *Gaussian Processes for Machine Learning (GPML)* toolbox (Rasmussen and Nickisch, 2010). We use a normal likelihood function for regression. For classification experiments, we use the logistic likelihood function, and the Laplace approximation. To further speed up inference calculations for large data sets, we use the FITC approximation (Snelson and Ghahramani, 2005) as implemented in *GPML* with 1,000 randomly sampled inducing points. FITC approximates the overall kernel matrix $\mathbf{K}$ by the covariances between instances and a set of inducing points, and the covariances between the inducing points. This reduces the costs of handling the kernel matrix from quadratic

in the number of instances, to linear in the number of instances and quadratic in the (constant) number of inducing points.

Hyper-parameters of *isoVCM* are the observation noise parameter and the kernel parameters; these are always tuned on the respective training set using gradient ascent in the marginal likelihood (again implemented through *GPML*).

Algorithm 1 Geospatial Transfer with *isoVCM*

Input: training instances $\mathbf{X} \in \mathbb{R}^{n \times m}$, task variables $\mathbf{T} \in \mathbb{R}^{n \times d}$, target variables $\mathbf{y} \in \mathbb{R}^n$; task kernel function $\boldsymbol{\kappa}(\cdot, \cdot)$; optionally ("dual case") input kernel function $k_{\mathcal{X}}(\cdot, \cdot)$; test instance $\mathbf{x}_\star$, $\mathbf{t}_\star$.

1: **primal case:** let $\mathbf{K}_{ij} = \mathbf{x}_i^\top \boldsymbol{\kappa}(\mathbf{t}_i, \mathbf{t}_j)\mathbf{x}_j$ and $\mathbf{k}_i = \mathbf{x}_i^\top \boldsymbol{\kappa}(\mathbf{t}_i, \mathbf{t}_\star)\mathbf{x}_\star$.
2: **dual case:** let $\mathbf{K}_{ij} = k_{\mathcal{X}}(\mathbf{x}_i, \mathbf{x}_j)k(\mathbf{t}_i, \mathbf{t}_j)$ and $\mathbf{k}_i = k_{\mathcal{X}}(\mathbf{x}_i, \mathbf{x}_\star)k(\mathbf{t}_i, \mathbf{t}_\star)$.
3: tune all kernel parameters and the observation noise hyper-parameter using marginal likelihood on the training data.
4: **regression:** infer $p(y_\star|\mathbf{K}, \mathbf{y}, \mathbf{k})$ with a standard GP toolbox using the FITC approximation.
5: **classification:** infer $p(y_\star|\mathbf{K}, \mathbf{y}, \mathbf{k})$ with a standard GP toolbox using the FITC and Laplace approximations.

Return $p(y_\star|\mathbf{K}, \mathbf{y}, \mathbf{k})$

3.4 Product Kernels in Transfer Learning

Sections 3.1 and 3.2 have shown that inference in the varying-coefficient model with isotropic GP priors is equivalent to inference in standard Gaussian processes with products of task kernels and instance kernels. Similar product kernels are used in several existing transfer learning models. Our results identify the generative assumptions that underlie these models by showing that the product kernels which they employ can be derived from the assumption of a varying-coefficient model with isotropic GP prior and an appropriate kernel function.

Bonilla et al. (2007) study a setting in which there is a discrete set of k tasks, which are described by task-specific attribute vectors $\mathbf{t}_1, \ldots, \mathbf{t}_k$. They study a Gaussian process model based on concatenated feature vectors $(\mathbf{x}, \mathbf{t})$ and a product kernel $k((\mathbf{x}, \mathbf{t}), (\mathbf{x}', \mathbf{t}')) = k_{\mathcal{X}}(\mathbf{x}, \mathbf{x}')k_{\mathcal{T}}(\mathbf{t}, \mathbf{t}')$, where $k_{\mathcal{X}}(\mathbf{x}, \mathbf{x}')$ reflects instance similarity and $k_{\mathcal{T}}(\mathbf{t}, \mathbf{t}')$ reflects task similarity. Theorem 1 and Corollary 1 identify the generative assumptions underlying this model: a varying-coefficient model with isotropic Gaussian process prior and kernel $k_{\mathcal{T}}$ generates task-specific parameter vectors in a reproducing Hilbert space of the instance kernel $k_{\mathcal{X}}$; a linear model in that Hilbert space generates the observed labels.

Evgeniou et al. (2005) and Álvarez et al. (2011) study multitask-learning problems in which task similarities are given in terms of a task graph. Their method uses the product of an instance kernel and the graph-Laplacian kernel of the task graph. We will now show that, when the task graph is a tree, that kernel emerges from Proposition 1. This signifies that, when the task graph is a tree, the graph regularization method of Evgeniou et al. (2005) is the dual formulation of hierarchical Bayesian multitask learning, and therefore Bayesian inference for hierarchical Bayesian models can be carried out efficiently using a standard Gaussian process with a graph-Laplacian kernel.

Definition 2 (Graph-Laplacian Multitask Kernel) Let $\mathcal{G} = (\mathcal{T}, \mathbf{M})$ denote a weighted undirected graph structure over tasks $\mathcal{T} = \{1, \ldots, k\}$ given by a symmetric adjacency matrix $\mathbf{M} \in \mathbb{R}^{k \times k}$, where $\mathbf{M}_{i,j}$ is the positive weight of the edge between tasks i and j or $\mathbf{M}_{i,j} = 0$ if no such edge exists. Let $\mathbf{D}$ denote the weighted degree matrix of the graph, and $\mathbf{L} = \mathbf{D} + \mathbf{R} - \mathbf{M}$ the graph Laplacian, where a diagonal matrix $\mathbf{R}$ that acts as a regularizer has been added to the degree matrix (Álvarez et al., 2011). The kernel function $k_{\mathbf{M},\mathbf{R}} : (\mathcal{X} \times \mathcal{T}) \times (\mathcal{X} \times \mathcal{T}) \to \mathbb{R}$ given by

$$k_{\mathbf{M},\mathbf{R}}((\mathbf{x}, \mathbf{t}), (\mathbf{x}', \mathbf{t}')) = \mathbf{L}^{\dagger}_{\mathbf{t},\mathbf{t}'} \mathbf{x}^{\mathsf{T}} \mathbf{x}',$$

where $\mathbf{L}^{\dagger}$ is the pseudoinverse of $\mathbf{L}$, will be referred to as the *graph-Laplacian multitask kernel*.

Proposition 2 states that the graph-Laplacian multitask kernel emerges as kernel function of the dual formulation of hierarchical Bayesian multitask learning (Definition 1).

Proposition 2 *Let $\mathcal{G} = (\mathcal{T}, \mathbf{A})$ denote a directed tree structure given by an adjacency matrix $\mathbf{A}$. Let $\boldsymbol{\sigma} \in \mathbb{R}^k$ be a vector with entries $\sigma_1, \ldots, \sigma_k$. Let $\mathbf{B} \in \mathbb{R}^{k \times k}$ denote the diagonal matrix with entries $0, \sigma_2^{-2}, \ldots, \sigma_k^{-2}$, let $\mathbf{R} \in \mathbb{R}^{k \times k}$ denote the diagonal matrix with entries $\sigma_1^{-2}, 0, \ldots, 0$, let $\mathbf{M} = \mathbf{BA} + (\mathbf{BA})^{\mathsf{T}}$, and let $k_{\mathbf{A},\boldsymbol{\sigma}}(\mathbf{t}, \mathbf{t}')$ be defined as in Proposition 1. Then*

$$k_{\mathbf{M},\mathbf{R}}((\mathbf{x}, \mathbf{t}), (\mathbf{x}', \mathbf{t}')) = k_{\mathbf{A},\boldsymbol{\sigma}}(\mathbf{t}, \mathbf{t}') \mathbf{x}^{\mathsf{T}} \mathbf{x}'.$$

Note that in Proposition 2, $\mathbf{BA}$ is an adjacency matrix in which an edge from node i to node j is weighted by the respective precision σ_j^{-2} of the conditional distribution (Equation 4); adding the transpose yields a symmetric matrix $\mathbf{M}$ of task relationship weights. The precision σ_1^{-2} of the root node prior is subsumed in the regularizer $\mathbf{R}$. The proof is included in the appendix.

4 Empirical Study

The main application areas of varying-coefficient models are prediction problems with underlying spatial or temporal dynamics (Gelfand et al., 2003; Fan and Zhang, 2008; Zhu et al., 2014; Estes et al., 2014). In this section, we will explore housing-price prediction and seismic-hazard analysis empirically.

4.1 Housing Prices

A typical instance of this class of problems is housing-price prediction. Theorem 1 and Corollary 1 show how Bayesian inference for varying-coefficient models can be carried out efficiently for regression and classification problems. We will therefore study varying-coefficient models and reference methods for larger-scale real-estate-price and monthly-rent prediction problems. These applications do not only represent the class of geospatial prediction problems, but are economically relevant problems in their own right.

Propositions 1 and 2 explain how hierarchical Bayesian multitask-learning models can be derived as varying-coefficient models with a hierarchy of tasks.

They reveal previously unknown relationships between known models which have been studied extensively, but do not lead to new learning techniques. These theoretical findings do not raise any questions that require empirical investigation.

4.1.1 Models under Investigation

We study *isoVCM*, as shown in Algorithm 1 (dual case). As kernel function $k(\mathbf{t}_i, \mathbf{t}_j)$ we always employ a Matérn kernel of degree $\nu = 1/2$, that is, $k(\mathbf{t}_i, \mathbf{t}_j) = \theta_{\mathbf{t}} \exp(-\|\mathbf{t}_i - \mathbf{t}_j\|/\rho_{\mathbf{t}})$. As kernel function $k_{\mathcal{X}}(\mathbf{x}_i, \mathbf{x}_j)$, we study both a linear kernel $k_{\mathcal{X}}(\mathbf{x}_i, \mathbf{x}_j) = \theta_{\mathbf{x}} \mathbf{x}_i^{\mathsf{T}} \mathbf{x}_j$ and a Matérn kernel $k_{\mathcal{X}}(\mathbf{x}_i, \mathbf{x}_j) = \theta_{\mathbf{x}} \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|/\rho_{\mathbf{x}})$ of degree $\nu = 1/2$. Here, $\theta_{\mathbf{t}}$, $\theta_{\mathbf{x}}$, $\rho_{\mathbf{t}}$, and $\rho_{\mathbf{x}}$ are hyperparameters of the kernel functions that are tuned according to marginal likelihood. The two resulting versions of our model are denoted by *isoVCM*^{lin} and *isoVCM*^{mat}, respectively.

We compare Algorithm 1 to the varying-coefficient model with nonisotropic GP prior by Gelfand et al. (2003), in which the covariances are inferred from data (denoted *Gelfand*). We also compare against the kernel-local smoothing varying-coefficient model of Fan and Zhang (2008) that infers point estimates of model parameters. We study this model using a linear feature map for instances $\mathbf{x} \in \mathcal{X}$ (*Fan & Zhang*^{lin}) and a nonlinear feature map constructed from the same Matérn kernel as used for *isoVCM*^{mat} (*Fan & Zhang*^{mat}). We add an ℓ_2 regularizer to the models of Fan and Zhang (2008), because this improves their prediction accuracy. Hyper-parameters of the model of Fan and Zhang (2008) are the regularization parameter and a bandwidth parameter in the smoothing kernel employed by their model. As the model only infers point estimates, hyper-parameters cannot be tuned by marginal likelihood; instead, optimal values of these parameters are inferred by cross-validation grid search on the training data.

We finally compare against *iid* models that assume that $p(y|\mathbf{x})$ is constant in $\mathbf{t}$, and models that treat the variables in $\mathbf{t}$ as additional features by appending them to the feature vector $\mathbf{x}$. Specifically, we study standard Gaussian processes on the original features $\mathbf{x}$, with a linear kernel $k(\mathbf{x}_i, \mathbf{x}_j) = \theta \mathbf{x}_i^{\mathsf{T}} \mathbf{x}_j$ and a Matérn kernel $k(\mathbf{x}_i, \mathbf{x}_j) = \theta \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|/\rho)$ of degree $\nu = 1/2$, denoted as $GP_{\mathbf{x}}^{lin}$ and $GP_{\mathbf{x}}^{mat}$. We also study standard Gaussian processes on a concatenated feature representation $(\mathbf{x}, \mathbf{t})$, again using a linear kernel $k((\mathbf{x}_i, \mathbf{t}_i), (\mathbf{x}_j, \mathbf{t}_j)) = \theta (\mathbf{x}_i, \mathbf{t}_i)^{\mathsf{T}} (\mathbf{x}_j, \mathbf{t}_j)$ and a Matérn kernel $k((\mathbf{x}_i, \mathbf{t}_i), (\mathbf{x}_j, \mathbf{t}_j)) = \theta \exp(-\|(\mathbf{x}_i, \mathbf{t}_i) - (\mathbf{x}_j, \mathbf{t}_j)\|/\rho)$ of degree $\nu = 1/2$. These baselines are denoted as $GP_{\mathbf{x},\mathbf{t}}^{lin}$ and $GP_{\mathbf{x},\mathbf{t}}^{mat}$. The standard Gaussian process baselines are also implemented using *GPML* and use the same likelihood functions and inference methods (including FITC) as *isoVCM*. Hyper-parameters of the baselines (observation noise parameter and kernel parameters θ, ρ) are also tuned using gradient ascent in the marginal likelihood on the training data.

4.1.2 Experimental Setting

We acquire records of real-estate sales in New York City (City of New York, 2013). The data set and our preprocessing are detailed in the appendix. For regression, the sales price serves as target variable y; for classification, y is a binary indicator that distinguishes between transactions with a price above the median of 450,000 dollars from transactions below it. After preprocessing, the data set contains 231,708 sales records with 94 attributes such as the the floor space, plot area,

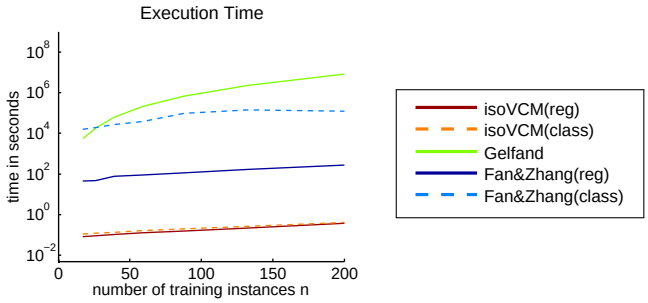

Fig. 1 Execution time over training set size n.

property class (*e.g.,* family home, condominium, office, or store), and the date of construction. We transform addresses into geographical latitude and longitude. We encode the sales date and geographical latitude and longitude of the property as task variable $\mathbf{t} \in \mathbb{R}^3$. This reflects the assumption that the relationship between property features and sales price vary smoothly in geographical location and time. We divide the records, which span dates from January 2003 to December 2009, into 25 consecutive blocks. Models are trained on a set of n instances sampled randomly from a window of five blocks of historical data and evaluated on the subsequent block; results are averaged over all blocks.

For rent prediction, we acquire records on the monthly rent for apartments and houses in the states of California and New York (US Census Bureau, 2013). Again, data set and preprocessing are detailed in the appendix. For regression, the target variable y is the monthly rent; for classification, y is a binary indicator that distinguishes contracts with a monthly rent above the median from those with a rent below. The preprocessed data sets contain 36,785 records (state of California) and 17,944 records (state of New York) with 24 input variables. Geographical latitude and longitude constitute the task variable $\mathbf{t} \in \mathbb{R}^2$. Models are evaluated using 20-fold cross validation; in each iteration, a random subset of n training instances is sampled randomly from the training part of the data.

4.1.3 Execution Time

We compare the execution times of *isoVCM*^{lin}, *Gelfand*, and *Fan & Zhang*^{lin}. Figure 1 shows the execution time for training and prediction on one block of test instances in the sales-price prediction task over the training sample size n.

For *Gelfand*, the most expensive step during inference is computation of the inverse of a Cholesky decomposition of an $nm \times nm$ matrix, which needs to be performed within each Gibbs sampling iteration. Figure 1 shows the execution time of 5000 iterations of this step (3,000 burn-in and 2,000 sampling iterations, according to Gelfand et al., 2003) which is a lower bound on the overall execution time. Bayesian inference in *isoVCM*^{lin} is between 6 and 7 orders of magnitude faster than in the *Gelfand* model. *isoVCM*^{lin} uses the FITC approximation; but since we use 1,000 inducing points and the sample size n stays below that in this experiment, FITC does not accelerate the inference. We conclude that *Gelfand*

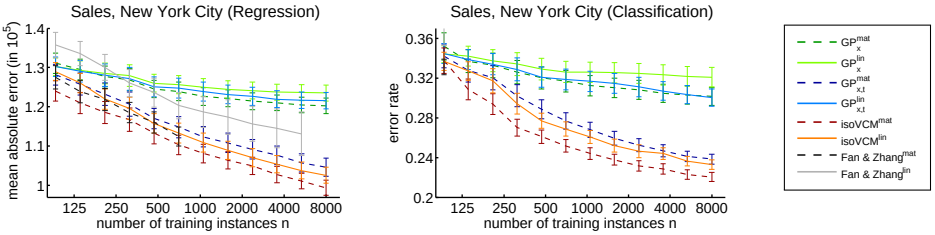

Fig. 2 Mean absolute error for real-estate sales prices in New York City. Error bars indicate the standard error.

is impractical for this application and exclude this method from the remaining experiments.

For *Fan & Zhang*^{lin}, separate point estimates of model parameters have to be inferred for each test instance, which involves solving a separate optimization problem. For regression, efficient closed-form solutions for parameter estimates are available. For classification, more expensive numerical optimization is required (Fan and Zhang, 2008); this results in a higher execution time

4.1.4 Prediction Accuracy

In all subsequent experiments, each method is given 30 CPU core days of execution time; experiments are run sequentially for increasing number n of training instances until the cumulative execution time reaches this limit.

Figure 2 shows the mean absolute error for real-estate sales-price predictions (left) and the mean zero-one loss for classifying sales transactions (right) as a function of training set size n. For regression, *Fan & Zhang*^{lin} and *Fan & Zhang*^{mat} partially completed the experiments; for classification, both methods did not complete the experiment for the smallest value of n. All other methods completed the experiments within the time limit. For regression, we observe that *isoVCM*^{lin} is substantially more accurate than $GP^{lin}_{\mathbf{x}}$, $GP^{lin}_{\mathbf{x},\mathbf{t}}$, and *Fan & Zhang*^{lin}; *isoVCM*^{mat} is more accurate than $GP^{mat}_{\mathbf{x}}$ and $GP^{mat}_{\mathbf{x},\mathbf{t}}$ with $p < 0.01$ for all training set sizes according to a paired t-test. Significance values of paired t-test comparing *isoVCM*^{mat} and *Fan & Zhang*^{mat} fluctuate between $p < 0.01$ and $p < 0.2$ for different n, indicating that *isoVCM*^{mat} is likely more accurate than *Fan & Zhang*^{mat}. For classification, *isoVCM*^{lin} substantially outperforms $GP^{lin}_{\mathbf{x}}$ and $GP^{lin}_{\mathbf{x},\mathbf{t}}$; *isoVCM*^{mat} outperforms $GP^{mat}_{\mathbf{x}}$ and $GP^{mat}_{\mathbf{x},\mathbf{t}}$ ($p < 0.01$ for $n > 125$).

Figure 3 shows the mean absolute error for predicting the monthly rent (left) and the mean zero-one loss for classifying rental contracts (right) for the states of California (upper row) and New York (lower row) as a function of training set size n. *Fan & Zhang*^{lin} completed the regression experiments within the time limit and partially completed the classification experiment; *Fan & Zhang*^{mat} partially completed the regression experiment but did not complete the classification experiment for the smallest value of n. We again observe that *isoVCM*^{mat} yields the most accurate predictions for both classification and regression problems; *isoVCM*^{lin} always

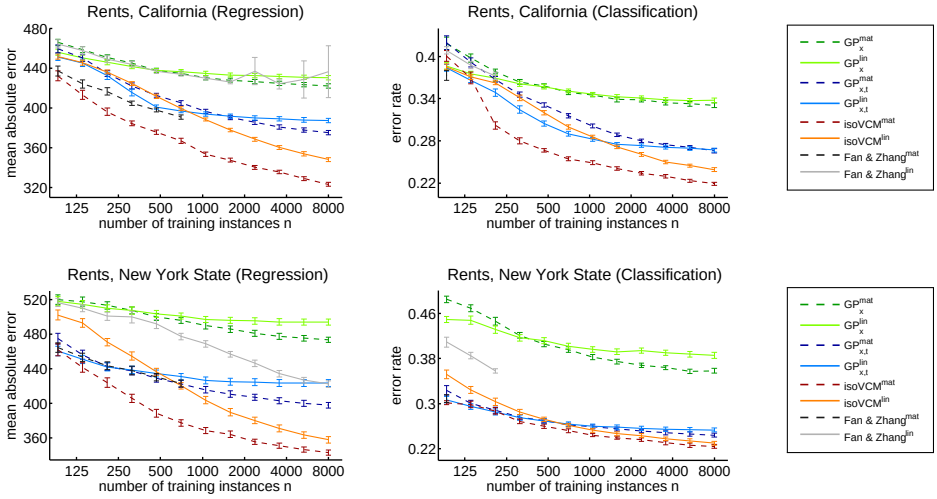

Fig. 3 Mean absolute error for monthly housing rents in the states of California (upper row) and New York (lower row).

yields more accurate predictions than *Fan & Zhang*^{lin} and more accurate predictions than $GP^{lin}_{\mathbf{x},\mathbf{t}}$ for training set sizes larger than $n = 1000$.

4.2 Seismic-Hazard Analysis

In this section, we study the model's ability to predict ground motion during seismic events in the State of California. Here, we focus on evaluating the model and exploring its versatility; an extended description of this study that focuses on the seismological findings has been published by Landwehr et al. (2016).

In this application, instances $\mathbf{x}_i = [M, R_{JB}, V_{S30}, F_{NM}, F_R]$ are seismic readings that consist of the magnitude of an earthquake, the Joyner-Boore distance (the distance to the vertical projection of the fault to the earth's surface), the time-averaged shear-wave velocity in the upper 30ms, and the style (normal or reverse) of faulting. This representation is commonly used in current ground-motion models. Target values y_i are the logarithmic peak ground acceleration—the highest acceleration which the ground at the given location will experience—and logarithmic spectral accelerations at time periods of 0.02, 0.05, 0.1,, 0.2, 0.5, 1, and 4s, in units of the earth gravity g. Spectral accelerations indicate resonance that may occur in buildings.

For each ground-motion record, latitude and longitude for both the seismic event, $\mathbf{t}_e$, and the station, $\mathbf{t}_s$, are available. The event coordinates are given by the horizontal projection of the geographical center of the rupture, estimated from the NGA West 2 source flatfile (Ancheta et al., 2014).

4.2.1 Models under Investigation

The structure of a ground-motion model is strongly guided by the underlying basic physical processes. An ergodic ground-motion model has the form

$$y = w_1 + w_2 M + w_3 M^2 + (w_4 + w_5 M)\sqrt{R_{JB}^2 + h^2} + w_6 R_{JB} + w_7 \log V_{s30} + w_8 F_R + w_9 F_{NM} \quad (14)$$

Since h is a constant, Equation 14 can be phrased as a linear model $y = \mathbf{w}^\top \mathbf{x}$ by including the compound terms M^2, $\sqrt{R_{JB}^2 + h^2}$, and $M\sqrt{R_{JB}^2 + h^2}$ in the input vector $\mathbf{x}$. Based on our understanding of the seismic process, we now allow some of these coefficients to vary with $\mathbf{t}_e$ and $\mathbf{t}_s$ by imposing a GP prior on them. For the remaining parameters, which by the nature of the underlying physical processes cannot depend on $\mathbf{t}_e$ or $\mathbf{t}_s$, the GP prior resolves to a standard Gaussian prior. Hence, the ground-motion model with varying coefficients has the form

$$y = \boldsymbol{\omega}_0(\mathbf{t}_e) + \boldsymbol{\omega}_1(\mathbf{t}_s) + \boldsymbol{\omega}_2 M + \boldsymbol{\omega}_3 M^2 + (\boldsymbol{\omega}_4(\mathbf{t}_e) + \boldsymbol{\omega}_5 M)\sqrt{R_{JB}^2 + h^2} + \boldsymbol{\omega}_6(\mathbf{t}_e) R_{JB} + \boldsymbol{\omega}_7(\mathbf{t}_s) \log V_{s30} + \boldsymbol{\omega}_8 F_R + \boldsymbol{\omega}_9 F_{NM} \quad (15)$$

This model is implemented using Algorithm 1 (primal case). Values of $\boldsymbol{\kappa}(\mathbf{t}, \mathbf{t}')$ are diagonal matrices with entries $k_1(\mathbf{t}, \mathbf{t}'), ..., k_d(\mathbf{t}, \mathbf{t}')$. This means that each dimension of $\boldsymbol{\omega}(\mathbf{t})$—corresponding to a particular coefficient—is generated by an independent scalar-valued Gaussian process whose covariance is given by $k_j(\mathbf{t}, \mathbf{t}') \in \mathbb{R}$. The kernel functions $k_j(\mathbf{t}, \mathbf{t}')$ are given by

$$k_j(\mathbf{t}, \mathbf{t}') = \begin{cases} \theta_j & \text{if } j \in \{2, 3, 5, 8, 9\} \\ \theta_j \exp\left(-\frac{\|\mathbf{t}_e - \mathbf{t}'_e\|}{\rho_j}\right) + \pi_j & \text{if } j \in \{0, 4, 6\} \\ \theta_j \exp\left(-\frac{\|\mathbf{t}_s - \mathbf{t}'_s\|}{\rho_j}\right) + \pi_j & \text{if } j \in \{1, 7\} \end{cases} \quad (16)$$

where θ_j, ρ_j and π_j are kernel parameters. For coefficients that do not depend on either event or station coordinates ($j \in \{2, 3, 5, 8, 9\}$), the kernel function $k_j(\mathbf{t}, \mathbf{t}')$ is constant, which implies that any function $\boldsymbol{\omega}$ drawn from the GP prior (Equation 1) is constant in its j-th dimension. For coefficients that depend on event or station coordinates ($j \in \{0, 4, 6\}$ or $j \in \{1, 7\}$, respectively), kernel function $k_j(\mathbf{t}, \mathbf{t}')$ is a Matérn kernel function of degree $\nu = 1/2$ based on the Euclidian distance between event or station coordinates, implying that the j-th dimension of $\boldsymbol{\omega}$ varies with $\mathbf{t}_s$ or $\mathbf{t}_e$. Parameter π_j is a constant offset to the Matérn kernel.

We compare against a global GP model of the form stated in Equation 14 that does not assume that any parameter varies with geographical latitude or longitude. Both models are implemented in *GPML*, using FITC inference and tuning all hyper-parameters according to marginal likelihood on the training data.

4.2.2 Experimental Setting

We use the NGA West 2 data set (Abrahamson et al., 2014). We use only the data from California and Nevada, since data from other regions will be spatially uncorrelated to this region. In total, there are 10,692 records from 221 earthquakes, recorded at 1425 stations. We run-10-fold cross validation on this data set.

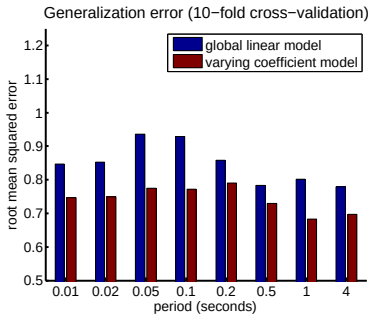

Fig. 4 Prediction of logarithmic spectral acceleration: RMSE for the VCM and the ergodic global model, estimated by 10-fold cross validation.

4.2.3 Results

Figure 4 shows the root-mean-squared prediction error (RMSE), estimated by 10-fold cross-validation. The VCM has a consistently lower RMSE than the global model. This indicates that incorporating spatial differences improves ground-motion prediction, even for a relatively small region such as California.

5 Discussion and Related Work

Gaussian processes are used widely in geospatial analysis (*e.g.,* Matheron, 1963). The covariogram is a basic tool in spatial models (*e.g.,* Cressie, 2015); it models the covariance between the values of a quantity at different points in space as a function of the distance between these points. Gaussian-process convolutions (Higdon, 2002) let an arbitrary kernel function induce a covariance function in (time and) space. Gaussian-process convolutions can be applied to discrete multi-task problems: multiple dependent output variables can share the same underlying spatial covariance (Ver Hoef and Barry, 1998). For instance, this allows to build a model for the concentration levels of multiple pollutants that share a joint spatial covariance.

In the linear model of coregionalization, multiple output dimensions (or discrete tasks) are coupled by scalar weights; the resulting kernel is a sum where each summand is the product of a covariance functions describing the dependence of output dimensions and a covariance function of the input coordinates. Gaussian-process regression networks (Wilson et al., 2011) model the dependency of multiple output variables (tasks) by an adaptive mixture of Gaussian processes, which resembles the way in which neural networks couple multiple output units via shared hidden variables. In all these models, time and space constitute the input or are part of the input; the multi-task nature of the learning problems results from multiple dependent output variables.

By contrast, varying-coefficient models reflect applications in which the conditional distribution $p(y|\mathbf{x}, \mathbf{t}, \mathbf{w})$ of a single output variable y shares much of its structure across different values of task variables $\mathbf{t}$; in our applications, these task variables are continuous and reflect space, time, or both. That is, a geospatial

correlation structure couples the parameters $\mathbf{w}$ of the relationship between $\mathbf{x}$ and y, and input variables $\mathbf{x}$ are treated differently from time and/or space variables $\mathbf{t}$. In varying-coefficient models with GP priors, $p(y|\mathbf{x}, \boldsymbol{\omega}(\mathbf{t}))$ varies smoothly in $\mathbf{t}$. While ridge and logistic regression assume that parameters $\mathbf{w}$ are generated by an isotropic Gaussian prior, we explore a model in which function $\boldsymbol{\omega}$ is governed by an independent GP prior for each dimension. Ridge regression, logistic regression, and *isoVCM* are discriminative models—they do not model the likelihood $p(\mathbf{x}|y, \mathbf{w})$ of the input variables. For discriminative models, the isotropy assumption on the model parameters does not translate into an isotropy assumption on the input attributes.

Propositions 1 and 2 shows that a GP with graph-Laplacian multi-task kernel (Evgeniou et al., 2005) emerges as dual formulation of hierarchical Bayesian multitask learning. The main motivation of varying-coefficient models, however, lies in application domains in which $p(y|\mathbf{x}, \boldsymbol{\omega}(\mathbf{t}))$ varies in time, location, or both.

In varying-coefficient models, each output y_i is generated by its own parameter vector $\mathbf{w}_i = \boldsymbol{\omega}(\mathbf{t}_i)$. Inference therefore involves nm parameters; and without an independence or isotropy assumption, $nm \times nm$ many covariances have to be inferred. This makes GP priors with full covariance (Gelfand et al., 2003) impractical for all but the smallest samples. Theorem 1 shows that, for isotropic GP priors, Bayesian inference in varying-coefficient models can be carried out efficiently with a standard GP using the product of a task kernel and an instance kernel. This clarifies the exact modeling assumptions required to derive the multitask kernel of Bonilla et al. (2007), and also highlights that hierarchical Bayesian inference can be carried out efficiently by using a standard GP with graph-Laplacian kernel (Evgeniou et al., 2005).

Product kernels play a role in other multitask learning models. In the linear coregionalization model, several related functions are modeled as linear combinations of GPs; the covariance function then resolves to a product of a kernel function on instances and a matrix of mixing coefficients (Journel and Huijbregts, 1978; Álvarez et al., 2011). A similar model is studied byWang et al. (2007); here mixing coefficients are given by latent variables. Zhang and Yeung (2010) study a model for learning task relationships, and show that under a matrix-normal regularizer the solution of a multitask-regularized risk minimization problem can be expressed using a product kernel. Theorem 1 can be seen as a generalization of their result in which the regularizer is replaced by a prior over functions, and the regularized risk minimization perspective by a fully Bayesian analysis.

Non-stationarity can also be modeled in GPs by assuming that either the residual variance (Wang and Neal, 2012), the scale of the covariance function (Tolvanen et al., 2014), or the amplitude of the output (Adams and Stegle, 2008) are input-dependent. The varying-coefficient model differs from these models in that the source of nonstationarity is observed in the task variable.

The main application areas of varying-coefficient models are prediction problems with underlying spatial or temporal dynamics, such as real-estate pricing (Gelfand et al., 2003; Fan and Zhang, 2008) neuroimaging (Zhu et al., 2014), and modeling of time-varying medical risks (Estes et al., 2014). In the domain of real estate price prediction, the dependency between property attributes and the market price changes continuously with geographical coordinates and time. Empirically, we observe that Bayesian inference in *isoVCM* is several orders of magnitude faster than inference in varying-coefficient models with nonisotropic

GP priors. We observe that the linear and kernelized *isoVCM* models predict real estate prices and housing rents more accurately over time and space than kernel-local smoothing varying-coefficient models, and are also more accurate than linear and kernelized models that append the task variables to the attribute vector or ignore the task variables.

In seismic hazard analysis, the model allows parameters of ground-motion models to vary smoothly in the location of seismic event and station. Today, seismic hazards in the State of California are predicted with an ergodic model whose parameters are fixed over all of California. We have derived a ground-motion model that imposes a GP prior on some model parameters. We observe that the varying-coefficient model consistently reduces the RMSE for predictions of the peak-ground acceleration and spectral accelerations substantially compared to the ergodic model.

Acknowledgements We would like to thank Jörn Malich and Ahmed Abdelwahab for their help in preparing the data sets of monthly housing rents. We gratefully acknowledge support from the German Research Foundation (DFG), grant LA 3270/1-1.

Appendix

A Proof of Proposition 1

The marginal $p(\mathbf{y}|\mathbf{X}, \mathbf{T}; \boldsymbol{\kappa})$ is defined by the generative process of drawing $\boldsymbol{\omega} \sim \mathcal{GP}(\mathbf{0}, \boldsymbol{\kappa})$, evaluating $\boldsymbol{\omega}$ for the k different tasks to create parameter vectors $\boldsymbol{\omega}(1), \ldots, \boldsymbol{\omega}(k)$, and then drawing $y_i \sim p(y|\mathbf{x}_i, \boldsymbol{\omega}(\mathbf{t}_i))$ for $i = 1, \ldots, n$. The marginal $p(\mathbf{y}|\mathbf{X}, \mathbf{T}; \mathcal{G}, \boldsymbol{\sigma})$ is defined by the generative process of generating parameter vectors $\bar{\mathbf{w}}_1, \ldots, \bar{\mathbf{w}}_k$ according to Equations 3 and 4 in Definition 1, and then drawing $y_i \sim p(y|\mathbf{x}_i, \bar{\mathbf{w}}_{\mathbf{t}_i})$ for $i = 1, \ldots, n$. Here, the observation models $p(y|\mathbf{x}_i, \bar{\mathbf{w}}_{\mathbf{t}_i})$ and $p(y|\mathbf{x}_i, \boldsymbol{\omega}(\mathbf{t}_i))$ are identical. It therefore suffices to show that $p(\boldsymbol{\omega}(1), \ldots, \boldsymbol{\omega}(k)|\boldsymbol{\kappa}) = p(\bar{\mathbf{w}}_1, \ldots, \bar{\mathbf{w}}_k|\mathcal{G}, \boldsymbol{\sigma})$.

The distribution $p(\bar{\mathbf{w}}_1, \ldots, \bar{\mathbf{w}}_k|\mathcal{G}, \boldsymbol{\sigma})$ can be derived from standard results for Gaussian graphical models. Let $\bar{\mathbf{W}} \in \mathbb{R}^{k \times m}$ denote the matrix with row vectors $\bar{\mathbf{w}}_1^\mathsf{T}, \ldots, \bar{\mathbf{w}}_k^\mathsf{T}$, and let $\mathrm{vec}(\bar{\mathbf{W}}^\mathsf{T}) \in \mathbb{R}^{km}$ denote the vector of random variables obtained by stacking the vectors $\bar{\mathbf{w}}_1, \ldots, \bar{\mathbf{w}}_k$ on top of another. According to Equations 3 and 4, the distribution over the random variables within $\mathrm{vec}(\bar{\mathbf{W}}^\mathsf{T})$ is given by a Gaussian graphical model (*e.g.,* Murphy (2012), Chapter 10.2.5) with weight matrix $\mathbf{A} \otimes \mathbf{I}_{m \times m} \in \mathbb{R}^{km \times km}$ and standard deviations $\boldsymbol{\sigma} \otimes \mathbf{1}_m$, where $\mathbf{1}_m \in \mathbb{R}^m$ is the all-one vector. It follows that the distribution over $\mathrm{vec}(\bar{\mathbf{W}}^\mathsf{T}) \in \mathbb{R}^{km}$ is given by

$$p(\mathrm{vec}(\bar{\mathbf{W}}^\mathsf{T})|\mathcal{G}, \boldsymbol{\sigma}) = \mathcal{N}(\mathrm{vec}(\bar{\mathbf{W}}^\mathsf{T})|\mathbf{0}, \bar{\boldsymbol{\Sigma}})$$

with

$$\bar{\boldsymbol{\Sigma}} = (\mathbf{I}_{km \times km} - \mathbf{A} \otimes \mathbf{I}_{m \times m})^{-1} \,\mathrm{diag}(\boldsymbol{\sigma} \otimes \mathbf{1}_m)^2 (\mathbf{I}_{km \times km} - \mathbf{A}^\mathsf{T} \otimes \mathbf{I}_{m \times m})^{-1},$$

where $\mathrm{diag}(\boldsymbol{\sigma} \otimes \mathbf{1}_m) \in \mathbb{R}^{km \times km}$ denotes the diagonal matrix with entries $\boldsymbol{\sigma} \otimes \mathbf{1}_m$.

The distribution $p(\boldsymbol{\omega}(1), \ldots, \boldsymbol{\omega}(k)|\boldsymbol{\kappa})$ is given directly by the Gaussian process defining the prior over vector-valued functions $\boldsymbol{\omega} : \mathcal{T} \to \mathbb{R}^m$ (see Equation 1). Let $\boldsymbol{\Omega} \in \mathbb{R}^{k \times m}$ denote the matrix with row vectors $\boldsymbol{\omega}(1)^\mathsf{T}, \ldots, \boldsymbol{\omega}(k)^\mathsf{T}$, then the Gaussian process prior implies

$$p(\mathrm{vec}(\boldsymbol{\Omega}^\mathsf{T})|\boldsymbol{\kappa}) = \mathcal{N}(\mathrm{vec}(\boldsymbol{\Omega}^\mathsf{T})|\mathbf{0}, \mathbf{G} \otimes \mathbf{I}_{m \times m})$$

(see, *e.g.,* Álvarez et al. (2011), Section 3.3). A straightforward calculation now shows $\mathbf{G} \otimes \mathbf{I}_{m \times m} = \bar{\boldsymbol{\Sigma}}$ and thereby proves the claim. □

B Proof of Corollary 1

Rewriting $p(y_\star|\mathbf{X}, \mathbf{y}, \mathbf{T}, \mathbf{x}_\star, \mathbf{t}_\star)$ in terms of a marginalization over the variables $\mathbf{z}$ and $z_\star$ leads to:

$$\begin{aligned}
&p(y_\star|\mathbf{X}, \mathbf{y}, \mathbf{T}, \mathbf{x}_\star, \mathbf{t}_\star) \\
=&\int p(y_\star|z_\star)p(z_\star|\mathbf{X}, \mathbf{y}, \mathbf{T}, \mathbf{x}_\star, \mathbf{t}_\star)\mathrm{d}z_\star \\
=&\iint p(y_\star|z_\star)p(z_\star|\mathbf{X}, \mathbf{z}, \mathbf{T}, \mathbf{x}_\star, \mathbf{t}_\star)p(\mathbf{z}|\mathbf{X}, \mathbf{y}, \mathbf{T})\mathrm{d}\mathbf{z}\mathrm{d}z_\star \\
\propto&\iint p(y_\star|z_\star)p(z_\star|\mathbf{X}, \mathbf{z}, \mathbf{T}, \mathbf{x}_\star, \mathbf{t}_\star)p(\mathbf{y}|\mathbf{z})p(\mathbf{z}|\mathbf{X}, \mathbf{T})\mathrm{d}\mathbf{z}\mathrm{d}z_\star.
\end{aligned}$$

The proof now quickly follows from Theorem 1 and derivations in the proof of Theorem 1: Equation 7 implies $p(z_\star|\mathbf{X}, \mathbf{z}, \mathbf{T}, \mathbf{x}_\star, \mathbf{t}_\star) = \mathcal{N}(z_\star|\mu_\mathbf{z}, \sigma_\mathbf{z}^2)$, Equation 12 implies $p(\mathbf{z}|\mathbf{X}, \mathbf{T}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{K} + \tau^2\mathbf{I}_{n\times n})$. □

C Proof of Proposition 2

In the following we use the notation that is introduced in Proposition 1 and Definition 2. We first observe that by the definition of the graph Laplacian multitask kernel it is sufficient to show that $\mathbf{G} = \mathbf{L}^\dagger$. Since the matrix $\mathbf{G}$ is invertible, this is equivalent to $\mathbf{G}^{-1} = \mathbf{L}$.

We prove the claim by induction over the number of nodes $|\mathcal{T}|$ in the tree $\mathcal{G}$. If $|\mathcal{T}| = 1$, then we have $\mathbf{A} = 0$, $\mathbf{D} = 0$, $\mathbf{R} = \sigma_1^{-2}$ and $\mathbf{M} = 0$. This leads to

$$\mathbf{G}^{-1} = (\mathbf{I} - \mathbf{A}^\mathsf{T})\sigma_1^{-2}(\mathbf{I} - \mathbf{A}) = \sigma_1^{-1} = \mathbf{D} + \mathbf{R} - \mathbf{M} = \mathbf{L}$$

and proves the base case. Let us now assume that we have a tree $\mathcal{G}_k$ with $|\mathcal{T}| = k > 1$ nodes. Let $\mathbf{t}$ be a leaf of this tree and $\mathbf{t}'$ shall be its unique parent. Suppose we have $\mathbf{t}' = i$ and w.l.o.g. we assume that $\mathbf{t} = k$. Let furthermore $\mathcal{G}_{k-1}$ be the tree which we get by removing the node k and its adjacent edge from the tree $\mathcal{G}_k$. Let $\mathbf{A}_k$ and $\mathbf{A}_{k-1}$ denote the adjacency matrices and $\mathbf{D}_k$ and $\mathbf{D}_{k-1}$ the degree matrices of $\mathcal{G}_k$ and $\mathcal{G}_{k-1}$. Let $\boldsymbol{\sigma}_k \in \mathbb{R}^k$ be the vector with entries $\sigma_1, \ldots, \sigma_k$, and $\boldsymbol{\sigma}_{k-1} \in \mathbb{R}^{k-1}$ be the vector with entries $\sigma_1, \ldots, \sigma_{k-1}$. Let $\mathbf{R}_k \in \mathbb{R}^{k\times k}$ denote the diagonal matrix with entries $\sigma_1^{-2}, 0, \ldots, 0$, and $\mathbf{R}_{k-1} \in \mathbb{R}^{k-1\times k-1}$ the diagonal matrix with entries $\sigma_1^{-2}, 0, \ldots, 0$. Let $\mathbf{B}_k \in \mathbb{R}^{k\times k}$ denote the diagonal matrix with entries $0, \sigma_2^{-2}, \ldots, \sigma_k^{-2}$ and $\mathbf{B}_{k-1} \in \mathbb{R}^{k-1\times k-1}$ the diagonal matrix with entries $0, \sigma_2^{-2}, \ldots, \sigma_{k-1}^{-2}$. Let $\mathbf{M}_k = \mathbf{B}_k\mathbf{A}_k + (\mathbf{B}_k\mathbf{A}_k)^\mathsf{T}$ and $\mathbf{M}_{k-1} = \mathbf{B}_{k-1}\mathbf{A}_{k-1} + (\mathbf{B}_{k-1}\mathbf{A}_{k-1})^\mathsf{T}$. Let $\mathbf{L}_k = \mathbf{D}_k + \mathbf{R}_k - \mathbf{M}_k$ and $\mathbf{L}_{k-1} = \mathbf{D}_{k-1} + \mathbf{R}_{k-1} - \mathbf{M}_{k-1}$.

In the following, we write $\mathrm{diag}(\mathbf{v})$ to denote a diagonal matrix with entries $\mathbf{v}$. We then have

$$\mathbf{A}_k = \left(\begin{array}{c|c} \mathbf{A}_{k-1} & \mathbf{e} \\ \hline \mathbf{0} & 0 \end{array}\right), \text{ where } \mathbf{e} = (\underbrace{0, \ldots, 0}_{i-1}, 1, 0, \ldots, 0)^\mathsf{T}$$

is the i^{th} $(n-1)$-dimensional unit vector. Using this notation we can write

$$\begin{aligned}
\mathbf{G}_k^{-1} &= (\mathbf{I} - \mathbf{A}_k^\mathsf{T})\,\mathrm{diag}(\boldsymbol{\sigma}_k)^{-2}(\mathbf{I} - \mathbf{A}_k) \\
&= \left(\begin{array}{c|c} \mathbf{I} - \mathbf{A}_{k-1}^\mathsf{T} & \mathbf{0} \\ \hline -\mathbf{e}^\mathsf{T} & 1 \end{array}\right)\left(\begin{array}{c|c} \mathrm{diag}(\boldsymbol{\sigma}_{k-1})^{-2} & \mathbf{0} \\ \hline \mathbf{0} & \sigma_k^{-2} \end{array}\right)\cdot\left(\begin{array}{c|c} \mathbf{I} - \mathbf{A}_{k-1} & -\mathbf{e} \\ \hline \mathbf{0} & 1 \end{array}\right) \\
&= \left(\begin{array}{c|c} \mathbf{L}_{k-1} + \sigma_k^{-2}\mathbf{e}\mathbf{e}^\mathsf{T} & -\sigma_k^{-2}\mathbf{e} \\ \hline -\sigma_k^{-2}\mathbf{e}^\mathsf{T} & \sigma_k^{-2} \end{array}\right).
\end{aligned}$$

In the last line we applied the induction hypothesis to the tree $\mathcal{G}_{k-1}$. Using the definitions of $\mathbf{L}$, $\mathbf{D}$, $\mathbf{R}$ and $\mathbf{M}$, we can easily finish the proof:

$$\begin{aligned}
\mathbf{G}_k^{-1} &= \left(\begin{array}{c|c} \mathbf{D}_{k-1} + \mathbf{R}_{k-1} - \mathbf{M}_{k-1} + \sigma_k^{-2}\mathbf{e}\mathbf{e}^\mathsf{T} & -\sigma_k^{-2}\mathbf{e} \\ \hline -\sigma_k^{-2}\mathbf{e}^\mathsf{T} & \sigma_k^{-2} \end{array}\right) \\
&= \left(\begin{array}{c|c} \mathbf{D}_{k-1} + \sigma_k^{-2}\mathbf{e}\mathbf{e}^\mathsf{T} & \mathbf{0} \\ \hline \mathbf{0} & \sigma_k^{-2} \end{array}\right) + \left(\begin{array}{c|c} \mathbf{R}_{k-1} & \mathbf{0} \\ \hline \mathbf{0} & 0 \end{array}\right) - \left(\begin{array}{c|c} \mathbf{M}_{k-1} & \sigma_k^{-2}\mathbf{e} \\ \hline \sigma_k^{-2}\mathbf{e}^\mathsf{T} & 0 \end{array}\right) \\
&= \mathbf{D}_k + \mathbf{R}_k - \mathbf{M}_k \\
&= \mathbf{L}_k .
\end{aligned}$$

This proves the claim. □

D Data-Set Preparation

We acquire records of real-estate sales in New York City for sales dating from January 2003 to December 2009 in June 2013 through the NYC Open Data initiative (City of New York, 2013).

Input variables include the floor space, plot area, property class (such as family home, residential condominium, office, or store), date of construction of the building, and the number of residential and commercial units in the building. After binarization of multi-valued attributes there are 94 numeric attributes in the data set. For regression, the sales price serves as target variable y; we also study a classification problem in which y is a binary indicator that distinguishes between transactions with a price above the median of 450,000 dollars from transactions below it. Date and address for every sale are available; we transform addresses into geographical latitude and longitude using an inverse geocoding service based on OpenStreetMap data. We encode the sales date and geographical latitude and longitude of the property as task variable $\mathbf{t} \in \mathbb{R}^3$. This reflects the assumption that the relationship between features of the property and its sales price vary smoothly in the geographical location and time.

A substantial number of records contain either errors or document transactions in which the valuations do not reflect the actual market values: for instance, several Manhattan condominiums sold for one dollar, and one-square-foot lots sold for massive prices. In order to filter most transactions with erroneous or artificial valuations by means of a simple policy, we only include records of sales within a price range of 100,000 to 1,000,000 dollars, a property area range of 500 to 5,000 square feet, and a land area range of 500 to 10,000 square feet. Approximately 80% of all records fall into these brackets. Additionally, we remove all records with missing values. After preprocessing, the data set contains 231,708 sales records. We divide the records, which span dates from January 2003 to December 2009, into 25 consecutive blocks. Models are trained on a set of n instances sampled randomly from a window of five blocks of historical data and evaluated on the subsequent block; results are averaged over all blocks.

For rent prediction, we acquire records on the monthly rent paid for privately rented apartments and houses in the states of California and New York from the 2013 American Community Survey's ASC public use microdata sample files (US Census Bureau, 2013). Input variables include the number of rooms, number of bedrooms, contract duration, the construction year of the building, the type of building (mobile home, trailer, or boat; attached or detached family house; apartment building), and variables that describe technical facilities (*e.g.,* variables related to internet access, type of plumbing, and type of heating). After binarization of multi-valued attributes there are 24 numerical attributes in the data. We study a regression problem in which the target variable y is the monthly rent, and a classification problem in which y is a binary indicator that distinguishes contracts with a monthly rent above the median of 1,200 dollars from those with a rent below the median. For each record, the geographical location is available in the form of a public use microdata area (PUMA) codecode (US Census Bureau, 2010).

We translate PUMA codes to geographical latitude and longitude by associating each record with the longitude-latitude-centroid of the corresponding public use microdata area; these geographical latitudes and longitudes constitute the task variable $\mathbf{t} \in \mathbb{R}^2$. We remove all records with missing values. The preprocessed data sets contain 36,785 records (state of California) and 17,944 records (state of New York).

References

N. A. Abrahamson, W. J. Silva, and R. Kamai. Summary of the ASK14 ground motion relation for active crustal regions. *Earthquake Spectra*, 30(3):1025–1055, 2014.

R. P. Adams and O. Stegle. Gaussian process product models for nonparametric nonstationarity. In *Proceedings of the 25th International Conference on Machine Learning*, 2008.

S. Akkar and Z. Cagnan. A local ground-motion predictive model for turkey, and its comparison with other regional and global ground-motion models. *Bulletin of the Seismological Society of America*, 100(6):2978–2995, 2010.

L. Al-Atik, N. Abrahamson, J. J. Bommer, F. Scherbaum, F. Cotton, and N. Kuehn. The variability of ground-motion prediction models and its components. *Seismological Research Letters*, 81(5):794–801, 2010.

M. A. Álvarez, L. Rosasco, and N. D. Lawrence. Kernels for vector-valued functions: a review. Technical Report MIT-CSAIL-TR-2011-033, Massachusetts Institute of Technology, 2011.

T. D. Ancheta, R. B. Darragh, J. P. Stewart, E. Seyhan, W. J. Silva, B. S.-J. Chiou, K. E. Wooddell, R. W. Graves, A. R. Kottke, D. M. Boore, T. Kishida, and J. L. Donahue. NGA-West2 database. *Earthquake Spectra*, 30(3):989–1005, 2014.

J. D. Anderson and J. N. Brune. Probabilistic seismic hazard analysis without the ergodic assumption. *Seismological Research Letters*, 70(1):19–28, 1999.

G. M. Atkinson and M. Morrison. Observations on regional variability in ground-motion amplitudes for small-to-moderate earthquakes in North America. *Bulletin of the Seismological Society of America*, 99(4):2393–2409, 2009.

D. Bindi, F. Pacor, L. Luzi, R. Puglia, M. Massa, G. Ameri, and R. Paolucci. Ground motion prediction equations derived from the italian strong motion database. *Bulletin of Earthquake Engineering*, 9(6):1899–1920, 2011.

E. V. Bonilla, F. V. Agakov, and C. K. I. Williams. Kernel multi-task learning using task-specific features. In *Proceedings of the 11th International Conference on Artificial Intelligence and Statistics*, 2007.

P. L. Bragato and D. Slejko. Empirical ground-motion attenuation relations for the eastern alps in the magnitude range 2.5-6.3. *Bulletin of the Seismological Society of America*, 95 (1):252276, 2005.

B. Chiou, R. Youngs, N. Abrahamson, and K. Addo. Ground-motion attenuation model for small-to-moderate shallow crustal earthquakes in California and its implications on regionalization of ground-motion prediction models. *Earthquake Spectra*, 26(4):907–926, 2010.

City of New York. NYC Open Data. `https://nycopendata.socrata.com/`, 2013.

Noel Cressie. *Statistics for spatial data.* John Wiley & Sons, 2015.

L. Danciu and G. A. Tselentis. Engineering ground-motion parameters attenuation relationships for greece. *Bulletin of the Seismological Society of America*, 97(1):162183, 2007.

Hal Daumé III. Frustratingly easy domain adaptation. *arXiv preprint arXiv:0907.1815*, 2009.

J. Estes, D. Nguyen, L. Dalrymple, Y. Mu, and D. Sentürk. Cardiovascular event risk dynamics over time in older patients on dialysis: A generalized multiple-index varying coefficient model approach. *Biometrics*, 70(3):751–761, 2014.

T. Evgeniou, C. A. Micchelli, and M. Pontil. Learning multiple tasks with kernel methods. *Journal of Machine Learning Research*, 6(1):615–637, 2005.

J. Fan and T. Huang. Profile likelihood inferences on semiparametric varying-coefficient partially linear models. *Bernoulli*, 11(6):1031–1057, 2005.

J. Fan and W. Zhang. Statistical methods with varying coefficient models. *Statistics and Its Interface*, 1(1):179–195, 2008.

J. R. Finkel and C. D. Manning. Hierarchical Bayesian domain adaptation. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2009.

A. E. Gelfand, H. Kim, C. F. Sirmans, and S. Banerjee. Spatial modeling with spatially varying coefficient processes. *Journal of the American Statistical Association*, 98(462):387–396, 2003.

A. Gelman, J. B. Carlin, H. S. Stern, and D. Rubin. *Bayesian Data Analysis.* Chapman & Hall, 1995.

N. Gianniotis, N. Kuehn, and F. Scherbaum. Manifold aligned ground motion prediction equations for regional datasets. *Computers and Geosciences*, 69:72–77, 2014.

T. Hastie and R. Tibshirani. Varying-coefficient models. *Journal of the Royal Statistical Society*, 55(4):757–796, 1993.

Dave Higdon. Space and space-time modeling using process convolutions. In *Quantitative methods for current environmental issues*, pages 37–56. Springer, 2002.

A. G. Journel and C. J. Huijbregts. *Mining Geostatistics*. Academic Press, London, 1978.

Niels Landwehr, Nicolas Kuehn, Tobias Scheffer, and Norman Abrahamson. A non-ergodic ground-motion model for California with spatially varying coefficients. *Bulletin of the Seismological Society of America*, 106:2574–2583, 2016.

P.-S. Lin, B. Chiou, N. Abrahamson, M. Walling, C.-T. Lee, and C.-T. Cheng. Repeatable source, site, and path effects on the standard deviation for empirical ground-motion prediction models. *Bulletin of the Seismological Society of America*, 101(5):2281–2295, 2011.

Georges Matheron. Principles of geostatistics. *Economic geology*, 58(8):1246–1266, 1963.

Kevin P. Murphy. *Machine learning: a probabilistic perspective*. MIT Press, Cambridge, MA, 2012.

C. E. Rasmussen and H. Nickisch. Gaussian processes for machine learning (GPML) toolbox. *Journal of Machine Learning Research*, 11:3011–3015, 2010.

B. Schölkopf and A. J. Smola. *Learning with kernels*. MIT Press, 2002.

E. Snelson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. In *Proceedings of the 19th Annual Conference on Neural Information Processing Systems*, 2005.

P. J. Stafford. Crossed and nested mixed-effects approaches for enhanced model development and removal of the ergodic assumption in empirical ground-motion models. *Bulletin of the Seismological Society of America*, 104(2):702–719, 2014.

V. Tolvanen, P. Jylanki, and A. Vehtari. Expectation propagation for nonstationary heteroscedastic Gaussian process regression. In *Proceedings of the 2014 IEEE International Workshop on Machine Learning for Signal Processing*, 2014.

US Census Bureau. Public use microdata areas. `https://www.census.gov/geo/reference/puma.html`, 2010.

US Census Bureau. 2013 American Community Surveys ASC public use microdata sample files. `http://factfinder.census.gov/faces/affhelp/jsf/pages/metadata.xhtml?lang=en&type=document&id=document.en.ACS_pums_csv_2013#main_content`, 2013.

Jay M Ver Hoef and Ronald Paul Barry. Constructing and fitting models for cokriging and multivariable spatial prediction. *Journal of Statistical Planning and Inference*, 69(2):275–294, 1998.

C. Wang and R. M. Neal. Gaussian process regression with heteroscedastic or non-Gaussian residuals. Technical Report CoRR abs/1212.6246, University of Toronto, 2012.

J. M. Wang, D. J. Fleet, and A. Hertzmann. Multifactor Gaussian process models for style-content separation. In *Proceedings of the 24th International Conference on Machine Learning*, 2007.

D. C. Wheeler and C. A. Calder. Bayesian spatially varying coefficient models in the presence of collinearity. *American Statistical Association, Spatial Modeling Section*, 2006.

Andrew Gordon Wilson, David A Knowles, and Zoubin Ghahramani. Gaussian process regression networks. *arXiv preprint arXiv:1110.4411*, 2011.

C. O. Wu and C. T. Chiang. Kernel smoothing on varying coefficient models with longitudinal dependent variable. *Statistica Sinica*, 10(1):433–456, 2000.

R. Yan and J. Zhang. Transfer learning using task-level features with application to information retrieval. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, 2009.

Y. Zhang and D. Yeung. A convex formulation for learning task relationships in multi-task learning. In *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence*, 2010.

H. Zhu, J. Fan, and L. Kong. Spatially varying coefficient model for neuroimaging data with jump discontinuities. *Journal of the American Statistical Association*, 109(507):1084–1098, 2014.