Mining Points-of-Interest for Explaining Urban Phenomena: A Scalable Variational Inference Approach

Christof Naumzik  
ETH Zurich  
Zurich, Switzerland  
cnaumzik@ethz.ch

Patrick Zöchbauer  
ETH Zurich  
Zurich, Switzerland  
pzoechbauer@ethz.ch

Stefan Feuerriegel  
ETH Zurich  
Zurich, Switzerland  
sfeuerriegel@ethz.ch

ABSTRACT

Points-of-interest (POIs; i.e., restaurants, bars, landmarks, and other entities) are common in web-mined data: they greatly explain the spatial distributions of urban phenomena. The conventional modeling approach relies upon feature engineering, yet it ignores the spatial structure among POIs. In order to overcome this shortcoming, the present paper proposes a novel spatial model for explaining spatial distributions based on web-mined POIs. Our key contributions are: (1) We present a rigorous yet highly interpretable formalization in order to model the influence of POIs on a given outcome variable. Specifically, we accommodate the spatial distributions of both the outcome and POIs. In our case, this modeled by the sum of latent Gaussian processes. (2) In contrast to previous literature, our model infers the influence of POIs without feature engineering, instead we model the influence of POIs via distance-weighted kernel functions with fully learnable parameterizations. (3) We propose a scalable learning algorithm based on sparse variational approximation. For this purpose, we derive a tailored evidence lower bound (ELBO) and, for appropriate likelihoods, we even show that an analytical expression can be obtained. This allows fast and accurate computation of the ELBO. Finally, the value of our approach for web mining is demonstrated in two real-world case studies. Our findings provide substantial improvements over state-of-the-art baselines with regard to both predictive and, in particular, explanatory performance. Altogether, this yields a novel spatial model for leveraging web-mined POIs. Within the context of location-based social networks, it promises an extensive range of new insights and use cases.

CCS CONCEPTS

- Mathematics of computing → Variational methods;  
- Computing methodologies → Gaussian processes;  
- Information systems → Location based services; Web mining.

KEYWORDS

Spatial Analytics; Point-of-Interest; Web-Mined Location Data; Gaussian Process; Variational Inference

1 INTRODUCTION

Points-of-interest (POIs) refer to geographic entities that are of relevance to people or businesses [33]. Illustrative examples include airports, bars, schools, restaurants, and landmarks. They are known as decisive factors that explain the spatial variance of social or urban phenomena [4, 17, 30, 36, 51]. POI data has recently become available in large quantity due to two reasons [3, 33], namely location-based social networks (e.g., Yelp, Foursquare) and web mapping services (e.g., Google Maps).

POIs can and have been used to explain various urban phenomena. Here the objective is to describe the spatial distribution of an outcome variable \( y \) based on POIs \( P \). This relevant to answering a number of question where POIs in a spatial neighborhood are highly discriminatory. For instance, what is the influence of neighboring POIs such as museums, shops, train stations, etc. on real estate prices? How do POIs such as bars or access to public transport describe real estate prices? Here POIs should be highly decisive in describing the distributions of the variables of interest (i.e., check-in counts, real estate prices). By properly understanding the relationships, the insights can help in generating managerially-relevant implications for business planning and, in particular, urban planning. Yet a holistic, interpretable model for measuring the influence of POIs on an urban outcome variable is missing.

Several approaches have been proposed in the literature in order to model the influence of POIs on outcomes. Typically, the POI influence at a location \( s \) is captured by feature engineering. These features are either distance-based [e.g., 37, 44] (e.g., distance to closest train airport), or density-based [e.g., 20–22, 40, 43]. Density-based features are derived from raw counting or the relative frequency of POIs in a neighborhood around \( s \). Despite being state-of-the-art, both approaches are of approximate nature and thus subject to inherent limitations: distance-based features consider only a small subset of POIs and their influence is not learnable as a function of the distance. In density-based features, the actual locations of POIs are ignored. Therefore, a comprehensive framework which is able to make rigorous statistical inference based on POIs is required and presents the contribution of this paper.

Modeling the influence of POIs: Our objective is to measure influence of the POIs \( P \) on an outcome \( y \). For this reason, we develop a novel probabilistic POI model for measuring the relationship between POIs and an outcome variable \( y(s) \) at a location \( s \) under full consideration of their spatial distributions. In our model, the influence of a POI depends on its distance to the location \( s \) and is...
modeled by an arbitrary kernel function. The kernel parameterization is fully learnable and thus does not require hyperparameter tuning. Furthermore, we distinguish different POI types by modeling a type-specific influence yielding more granular insights. Examples for POI types include train station, bar, restaurant, etc. Mathematically, we introduce a sum of latent Gaussian processes, which allows us to learn the effect of individual POIs. Altogether, the resulting model formulation naturally aids interpretability. We later discuss extensions to our model that highlight its flexibility.

**Adapted learning algorithm:** We develop a scalable learning algorithm to estimate the POI-based spatial model. A core objective is to obtain a scalable Bayesian approach, so that, given the outcome $y$, we compute a variational approximation of the full posterior distribution of the latent processes. This has several benefits: First, we obtain estimates of the model uncertainty, which is a desirable property in decision support. Specifically, this lets us determine uncertainty regions for all latent POI influences. Second, our proposed sparse variational inference approach allows scalability in contrast to traditional Bayesian sampling approaches. Third, tuning parameters are eliminated and, therefore, all parameters are estimated directly.

We obtain our learning algorithm for our POI model by deriving a tailored evidence lower bound (ELBO). Based on this, we retrieve multiple variational estimates, namely, one for the posterior of each individual latent Gaussian process. This differs from a straightforward approach where only one variational estimate for the posterior distribution of the sum of the latent processes would have been obtained. Owing to our derivation, the learning algorithm has a complexity of $O(NM^3)$ instead of $O(N^3)$, where $M \ll N$ denotes the number of inducing points used for the sparse approximation of the posterior distribution given $N$ observations and where $N$ refers to the minibatch size during stochastic gradient descent.

**Evaluation:** Our combination of POI model and learning algorithm is evaluated based on different urban applications in order to demonstrate its widespread applicability. For this reason, we demonstrate how POI locations describe, on the one hand, real estate prices and, on the other hand, restaurant popularity. In both cases, our POI model is highly interpretable and, on top of that, state-of-the-art approaches (both linear and non-linear such neural networks) based on feature engineering are outperformed.

## 2 RELATED WORK

**Spatial analytics:** Spatial analytics offers a variety of techniques to estimate locational variations of social, urban, and business-related observations. Such estimators must adapt to spatial autocorrelation and heterogeneity that is commonly present when making inferences with locational entities.

Spatial analytics gave rise to a wide range of applications including, for instance, ecology [6], crime prediction [19, 20], traffic accident investigation [24], social sciences [7], geo-statistics [31], and business management [37]. Finally, spatial analytics has become important when mining the abundant information that originates from location-based services [30].

**POI modeling:** POIs can explain various spatial distributions in order to aid computational social sciences [20, 21], urban computing [51], and entrepreneurial decision support [22, 37, 40]. Identifying points of interest has been subject to extensive research, where challenges arise from attributing locations and from processing unstructured data sources [26, 29, 33].

A common approach to including POI locations in descriptive or predictive models generates distance-based features measuring the distance between POIs and the location of outcome variables [e.g., 37, 44]. Features are sometimes subject to transformations [e.g., 44], for which the relationship is not learnable but fixed.

An alternative approach is density-based estimates where the relative or absolute frequency of POIs in a spatial proximity are counted [e.g., 20–22, 40, 43]. This is common when outcomes are reported not as locations but at an aggregated spatial unit (e.g., neighborhoods or other discretizations). As detailed earlier, both approaches to feature engineering are subject to inherent theoretical and numerical limitations, which motivates the need for our research.

Mathematically, our POI model shares methodological similarities with co-kriging [1]. Co-kriging is a multivariate variant of the ordinary kriging operation. It makes predictions for a spatially distributed variable (i.e., the outcome) with the help of another spatially distributed variable (the so-called co-variable). Similar to our model, it can be described as a combination of multiple Gaussian processes. However, in the case of co-kriging, each Gaussian process underlies an individual observed outcome variable. The latent influence functions of POIs are, however, not observable. Hence, co-kriging is not applicable in our setting. Furthermore, its inference algorithm is limited to a complicated (heuristic) computation scheme.

**POI recommendation:** Another branch of research focuses on POI recommendation [10, 39, 48, 50, 52]. These works suggest POIs for future visits. Mathematically, the objective is model the probability of visiting a POI based on the past history of a user’s check-ins. For this reason, the output is at POI-level. This is in stark contrast to our work where POIs are used to describe a different spatial distribution as output and where POI-level data thus serve merely as input. Owing to this, POI recommendation appears related but does not yield suitable baselines. Furthermore, POI recommendation draws upon sequential input, i.e., check-in histories [39, 52]. This differs from our work where we face purely spatial input for which the influence is modeled while ensuring interpretability.

POI recommendation makes use of additional covariates, such as geographical influence [10, 12, 18, 48, 49, 52–55], social influence [11, 45, 55], temporal patterns [10–12, 18, 47, 52, 53], or user-provided content [41]. However, this stream of research differs from our objective, since POI visits represents the outcome variable, whereas we leverage POIs for describing a different spatial distribution of interest (e.g., not how crime risk influences POI visits as in POI recommendation but, vice versa, how POI locations influence crime risk).

**Urban planning** leverages POI data in order to improve decision-making, e.g.: POIs signal the popularity of urban environments [4, 51], measure urban social diversity [17], help location planning [27], and predict the spread of diseases [42], or crime [20, 38]. Yet a holistic approach for modeling the influence of POIs is missing.
3 MODELING THE INFLUENCE OF POIS

In this section, we introduce our proposed probabilistic model in order to estimate the influence of POIs. In doing so, we adhere to common practice in spatial modeling by incorporating feature vectors with additional covariates and a term controlling for spatial heterogeneity. A detailed introduction of spatial modeling can be found in [1].

3.1 Motivation

Our POI model draws upon the following input. Let \( y = \{y_n\}_{n=1}^N \) denote the outcome variable of interest with \( N \) observations.\(^2\) In our motivational examples, this variable would correspond to, for instance, the sales of a restaurant or visitor counts at tourist hotspots. Each observation is subject to a spatial distribution and, hence, the outcome \( y_n \) is associated with a location \( s_n \).

The outcomes can be optionally associated with an additional feature vector \( x_n \), e.g., type of restaurant or price range, with \( n = 1, \ldots, N \). The set of POIs is given by \( \mathcal{P} \), where each POI \( p_j \in \mathcal{P} \) is given by a 3-tuple, i.e., \( p_j = (\omega_j, y_j, \eta_j) \) with location \( \omega_j \), type \( y_j \in \Gamma \), and POI-specific covariates \( \eta_j \), e.g., size of a train station. \( \Gamma \) denotes the set of POI types, e.g., airport, bar, restaurant etc. For reasons of notation, we introduce \( P_y = \{p_j \mid y_j = y\} \), which gives the subset of POIs of type \( y \in \Gamma \). Based on the above input, our objective is to describe the dependence of the outcome variable on the POI influence \( \gamma \).

Formally, let \( f \) refer to a corresponding function that describes the relationship among the input variables in order to obtain the outcome under consideration of its spatial distribution, i.e.,

\[
    f : (x_n, s_n, \mathcal{P}) \mapsto y_n, \quad \text{for } n = 1, \ldots, N. \quad (1)
\]

Learning the function \( f \) eventually allows us to answer the question: what is the influence of the POIs \( \mathcal{P} \) on \( y \)?

3.2 Model Specification

In order to formalize the relationship \( f \), we propose a model formulation that describes an outcome \( y_n \) on the basis of three components: (1) a feature influence from \( x_n \), (2) a POI influence from \( \mathcal{P} \), and (3) the remaining spatial variation as follows:

1. The feature influence \( g(x_n; \theta_	ext{f}) \) with parameters \( \theta_	ext{f} \) depends on the features \( x_n \) that are directly associated with the location \( s_n \) of the outcome \( y_n \). Thus, this component captures the variance introduced by the outcome-specific features. For instance, when predicting check-ins, it would relate to the heterogeneity introduced by the restaurant itself, such as its type, its mean rating, and the price range. At this point, we make no specific assumptions concerning the function \( g \) and, depending on the objective, it could be set, for instance, to a linear function if interpretability is desired or to a neural network if predictive performance is of importance.

2. The POI influence describes the effect of all POIs on the outcome. Each POI \( p \in \mathcal{P} \) with \( p = (\omega, y, \eta) \) has an influence \( \alpha_p k_p(s_n, \omega, \eta; \theta_p) \) on the outcome \( g(s_n) \) at location \( s_n \), where \( \alpha_p \) is a POI-specific scaling factor and where the kernel \( k_p \) measures the (normalized) effect of \( p \) on \( y_n \). This effect is POI-type-specific, i.e., we retrieve separate kernel functions \( k_p \) for all \( y \in \Gamma \), i.e., for airports, bars, restaurants, and so forth. From a practical perspective, this is highly desired, since, for example, an airport may have a much larger area of influence than a bar.

It is important to highlight that \( k_p \) depends on both \( s_n \) and \( \omega \): this allows the POI influence to vary according to the distance between the outcome location \( s_n \) and the POI location \( \omega \) and, for this reason, we obtain a variable influence of POIs with respect to their locations. The influence \( k_p \) is further parameterized by parameters \( \theta_p \) that are learnable. As we shall see later, we obtain efficient inferences regardless of the chosen kernel function \( k_p \). Later, in our model, we obtain the overall POI influence by summing over all POIs \( \rho \in \mathcal{P} \).

3. Spatial variation beyond POIs is additionally included as commonly done in spatial modeling [1]. This is given by \( h_0(s_n) \). We remind the reader that this is only the remaining heterogeneity that was not yet captured by the POIs and, as we shall see later in our experiments, its magnitude/shape depends on the use case. Again, we make no specific assumptions concerning \( h_0 \) and thus follow the literature on spatial modeling [1] where it is standard to estimate the spatial heterogeneity based on a latent zero-mean Gaussian process \( \mathcal{G}(0, h_0(\cdot, \cdot; \theta_0)) \) with a kernel function \( k_0 \) and parameters \( \theta_0 \).\(^3\)

Combining the different components yields

\[
    f(x_n, s_n, \mathcal{P}; \theta) = g(x_n; \theta_	ext{f}) + \sum_{p \in \mathcal{P}} \alpha_p k_p(s_n, \omega, \eta; \theta_p) + h_0(s_n), \quad (2)
\]

where we use \( \theta = (\theta_	ext{f}, \theta_p, \theta_0) \).

3.3 Strengths of our POI Model

Our formulation for modeling the influence of \( \mathcal{P} \) on \( y \) immediately reveals several benefits. Most importantly, we overcome the limitations of state-of-the-art approaches based on feature engineering as follows.

First, we consider the complete spatial distribution of POIs. This differs from distance-based metrics used for feature engineering where often only the single nearest POI of a specific type is considered. In contrast, our POI model considers the locations of all POIs. Our POI model also circumvents weaknesses of density-based approaches, which, by definition, neglect the true locations of POIs. In this sense, our POI model also takes spatial interactions between POIs into account.

Second, the intensity of our POI influence is variable and can thus have a larger impact in close proximity to \( s_n \) of \( y_n \). This relationship is flexible due to our learnable parameterization. Such flexibility aids interpretations, as is best illustrated with an example: in crime forecasting, a bar might act as an attractor of crime. However, this only holds true up to a maximum but unknown distance, \( d_{\text{max}} \), after which the impact of the bar vanishes. In our model, this maximum distance \( d_{\text{max}} \) can be estimated from the data. By definition, the distance-dependent utility of POIs cannot be considered in density-based feature engineering. Distance-based feature engineering is

\(^{3}\)Additional noise that cannot be explained by \( f \) alone may be captured by a noise term included in \( h_0 \), often referred to as “nugget” in the spatial literature.
sometimes subject to manually-defined transformations [e.g., 44]. However, such an approach is not directly learnable with the same flexibility as our kernel parameterization. Furthermore, our model formulation facilitates interpretation: we numerically recover the latent POI influence as a function of the distance $\|s_n - \omega\|$.

Third, we explicitly control for the remaining spatial heterogeneity. While this is not unique to our POI model, it reveals weaknesses in some of the approaches considered in the literature where spatial heterogeneity is frequently incorporated via coarse district-based dummy variables. In contrast, we decided to include a flexible function $h_0$, so that we can adapt to continuous spatial variations.

Accordingly, we overcome inherent theoretical limitations of state-of-the-art approaches based on feature engineering, and, as we show later, this is also confirmed in our numerical experiments. On top of that, we obtain a model that benefits from rich interpretability.

### 3.4 Bayesian Model Formulation

Our aim below is to translate the previous model structure into a Bayesian formulation. For this, we first introduce additional notation. We combine the POI influence from all POIs of the same type into a function $h_{\gamma}$ given by

$$h_{\gamma}(s_n, P_\gamma; \theta_\gamma) = \sum_{\rho \in P_\gamma} \alpha_{\rho} f_\gamma(s_n, \omega_\rho, \eta; \theta_\gamma).$$

(3)

Based on $h_{\gamma}$, the above POI model from Equation 2 may be written as

$$f(x_n, s_n, P; \theta) = g(x_n; \theta_x) + \sum_{\gamma \in G} h_{\gamma}(s_n, P_\gamma; \theta_\gamma) + h_0(s_n).$$

(4)

To ease notation, we further introduce a vector notation for the spatial function $f$, i.e., $f = \{f_n = f(x_n, s_n, P; \theta)\}_{n=1}^N$ as well as the POI influence of type $\gamma$, i.e., $h_{\gamma} = \{h_{\gamma}(s_n, P_\gamma; \theta_\gamma)\}_{n=1}^N$.

We now state the generative equation of our proposed model. It consists of (1) the observed responses, (2) the spatial regression function $f$, and (3) the likelihood $p(y \mid f)$. Based on this, the objective is to estimate the model parameters $\theta$ as well as the latent functions $h_0$ and $h_{\gamma}$. For ease of notation, we define $h^0_0 = h_0(s_n)$. This gives

$$\theta - \rho;$$

(5a)

$$\alpha_{\gamma} = \{\alpha_{\rho}\}_{\rho \in P_\gamma} \sim p(\alpha_\gamma), \quad \text{for all } \gamma \in G,$$

(5b)

$$h_0 = \{h^0_0\}_{n=1}^N \sim \mathcal{N}(0, \delta h_0(\cdot; \theta_0)),$$

(5c)

$$f(x_n, s_n, P; \theta) = g(x_n; \theta_x) + \sum_{\gamma \in G} h_{\gamma}(s_n, P_\gamma; \theta_\gamma) + h_0^\gamma,$$

(5d)

$$y_n \sim p(f(x_n, s_n, P; \theta)).$$

(5e)

Equations (5a) to (5c) provide the priors; Equation (5d) the spatial regression which is identical to our initial structure from Equation (2); and Equation (5e) the likelihood.

We emphasize the following properties. In Equation (5e), we make two assumptions: On the one hand, the outcomes $y_n$ are conditionally independent given $f$ and, based on this, one can factorize the likelihood; i.e., $p(y \mid f) = \prod_{n=1}^N p(y_n \mid f_n)$. The likelihood is further problem-specific and must be chosen in accordance with the distribution of the outcome variable. For instance, it could follow a Gaussian or any other distribution and thus contributes to the widespread applicability of our approach. On the other hand, we incorporate an implicit hierarchical structure: the scaling parameter $\alpha_{\gamma}$ is allowed to be sampled from a different distribution with its own mean and variance for each POI type, where the effect scales $\alpha_{\gamma}$ are assumed independent between POI-types $\gamma$.

### 3.5 Interpretability

Our POI model has extensive capacity for interpretation, as it recovers the latent POI influence $h_{\gamma}$, which consists of a kernel $k_\gamma$ and scaling factor $\alpha_{\gamma}$. The kernel is best interpreted by visualization, as it shows the POI influence as a function of distance. Examples can be found in Figure 3 for the kernel functions chosen in the case studies. For the scaling factor, we suggest computing the estimated average effect size, which is given by

$$\bar{\alpha}_{\gamma} = \frac{1}{|P_{\gamma}|} \sum_{\rho \in P_{\gamma}} \alpha_{\rho}.$$  

(6)

The estimated average effect size quantifies the magnitude of influence for each POI type $\gamma$. In addition, the scaling factors $\alpha_{\gamma}$ allow us to uncover the contribution of each individual POI to the outcome variable $y(s)$ at location $s$. An example of this is given in Figure 5.

### 4 LEARNING ALGORITHM

We remind the reader of our objective. Our primary interest is to infer the posterior distribution of the function $f$ and, more specifically, the posterior distributions of $h_{\gamma}$. The latter is important as it uncovers the latent POI influence. The parameter vector $\theta$ merely describes the influence of control variables for which we adjust. Hence, only point estimates for $\theta$ (rather than the complete posterior) are required to enable sufficient interpretation.

We now derive a scalable learning algorithm for our POI model. Alternative estimation procedures can be ruled out for various reasons. On the one hand, the full posterior distribution is not analytically tractable and, hence, requires Bayesian sampling (e.g., MCMC), which is naturally expensive. On the other hand, sparse and variational approximations for the posterior distribution are available [cf. 32, 35, for an overview], but limited to the Gaussian process $h_0$ and, hence, cannot be directly used for estimating our complete POI model.

As our contribution, we show that one can still develop efficient inference for Equation (5). In a first step, we derive that the POI model has an equivalent formulation as a sum of latent Gaussian processes. Second, we derive a novel ELBO for such latent models, which eventually gives rise to variational approximations of the posterior distribution. Moreover, it fulfills our objective of inferring posterior distributions for each latent POI influence $h_{\gamma}$. Third, we provide an analytical derivation of the ELBO for specific likelihood distributions, which can additionally benefit efficient computations.

Our Theorem 4.4 advances the literature on variational inference for latent structure models. The inducing point framework from

---

4As our objective is to provide a general framework for POI modeling, we refrain from specifying any of the distributions at this point but use a generic $p$ for all of the distributions.

5In a slight abuse of notation, we write $\theta_{\gamma}$ for both the parameters of the kernel function $k_{\gamma}$ as well as for the parameters of the prior distribution for $\alpha_{\gamma}$.

6The model was implemented using the GPyFlow package [25] for python. The models were fit using stochastic gradient descent using the Adam optimization algorithm with an initial learning rate of 0.01. We applied a 70/30 split for training and test data. The inducing points are randomly sampled from the training data with $M = 400$. 

2345
We recall that one may think of Gaussian processes as convolutions of random fields of Gaussian white noise via a kernel function \[ \kappa \]. Based on this, we show that, under mild assumptions for the prior distribution of the effect sizes \( \alpha_y \) in Equation (5b), the function \( h_y \) follows a Gaussian process and, hence, Equation (5d) can be written as a sum of latent Gaussian processes. Accordingly, the following lemma states that each POI influence \( h_y \) induces a latent zero-mean Gaussian process.

**Lemma 4.1.** For \( \alpha_y \sim \mathcal{N}(0, \Sigma_y) \) it holds that the model specified in Equation (5) describes a Gaussian process where each POI influence function \( h_y \) itself induces a zero-mean Gaussian process.

**Proof.** To see this, let

\[
 h_y(s_n, p_y; \theta_y) = \sum_{p \in P_y} \alpha_{p} \kappa_p(s_n, wp; \theta_y) = \kappa_y(s_n, p_y; \theta_y)^T \alpha_y .
\]  

(7)

Since the multivariate Gaussian distribution is closed under linear combinations, the POI influences \( h_y(s_1), \ldots, h_y(s_n) \) are each given by a multivariate Gaussian process with zero-mean and covariance \( \Sigma_y \), i.e.,

\[
 \mathbb{E} \left[ h_y(s) h_y(s') \right] = \mathbb{E} \left[ \kappa_y(s, p_y; \theta_y)^T \alpha_y \right] \mathbb{E} \left[ \kappa_y(s', p_y; \theta_y) \right] = \kappa_y(s, s'; \theta_y) .
\]  

(8)

Each POI type thus induces a latent zero-mean Gaussian process with kernel \( \kappa_y(\cdot, \cdot; \theta_y) \) and, hence, the model developed in Equation (5) can be reformulated as

\[
 \theta \sim \mathcal{P}, \quad h_y \sim \mathcal{GP}(0, \kappa_y(\cdot, \cdot; \theta_y)), \quad h_0 \sim \mathcal{GP}(0, \kappa_0(\cdot, \cdot; 0_0)), \quad f(x_n, s_n; \theta) = g(x_n; \theta) + \sum_{p \in P} h^*_p + h^*_0
\]  

(9a)

\[
 y_n \sim \mathcal{P}(f(x_n, s_n; \theta))
\]  

(9b)

for \( n = 1, \ldots, N \).

We note that the result holds independently of the specific kernel function \( \kappa \) and, hence, provides a modeling approach that is highly flexible. We refer to the covariance \( \Sigma_y \) as the **effect size structure** as it determines the dependency between POIs of the same type. We later discuss how a suitable choice of \( \Sigma_y \) can incorporate further domain knowledge for a specific use case. Furthermore, although the assumption of a multivariate normal distribution as a prior might seem restrictive, we note that it typically presents only a weakly informative prior \([9]\).

We note that under this model formulation we are still able to recover the individual POI effects \( \{\alpha_p\}_{p \in P_y} \) using Equation (3) which yields

\[
 \alpha_y = \{\alpha_p\}_{p \in P_y} = (\kappa_y(S, P_y))^{\dagger} h_y ,
\]  

(10)

where the dagger operator \( \dagger \) denotes the Moore-Penrose pseudoinverse and \( S = \{s_n\}_{n=1}^N \).

### 4.2 ELBO Derivation with Inducing Points for Additive Latent Gaussian Processes

Our proposed learning algorithm for Equation (9) makes use of two distinctive concepts in order to reduce computational complexity. To ease notation, we define \( \mathcal{I}_y = \mathcal{I} \cup \{0\} \) as the index set which refers to both the spatial Gaussian process \( h_0 \) and the POI-specific Gaussian processes \( h_y \).

First, we come up with a variational approximation of each Gaussian process; i.e., we find suitable variational approximations \( q(h_y) \) of the posterior distribution \( p(h_y | y) \).\(^7\) For this reason, we derive a lower bound for the marginal likelihood \( p(y) \) – the so-called evidence lower bound (ELBO) – for the specific case of additive latent Gaussian processes. The key idea is then to maximize our problem-specific ELBO with respect to the variational approximations \( q(h_y) \), thereby maximizing the likelihood \( p(y) \).

Second, we further improve scalability by applying a sparse approximation. For this, we draw upon an inducing point method \([32]\).

Formally, we augment the latent variables with additional outputs \( u_y \) from the corresponding Gaussian process at inducing point locations \( \mathcal{Z} = \{z_m\}_{m=1}^M \subseteq S \). Hence \( p(u_y) \) is a multivariate Gaussian distribution. Now, instead of maximizing the ELBO directly with respect to \( q(h_y) \), we define \( q \) with output \( u_y \) at the inducing points. The following assumptions are required for deriving our ELBO for a sparse variational approximation:

**Assumption 4.2 (Independence at inducing points).** There exist inducing points \( \mathcal{Z} = \{z_m\}_{m=1}^M \subseteq S \) with mutually independent outputs \( u_y \); i.e.,

\[
 p(u_y, u_{y'}) = p(u_y) p(u_{y'}) \quad \text{and} \quad q(u_y, u_{y'}) = q(u_y) q(u_{y'})
\]  

(11)

for \( y \neq y' \) with \( y, y' \in \mathcal{I}_0 \).

**Assumption 4.3 (Conditional independence).** For \( y, y' \in \mathcal{I}_0 \) with \( y \neq y' \), the joint conditional distribution factorizes as

\[
 p(h_y, h_{y'} | u_y, u_{y'}) = p(h_y | u_y) p(h_{y'} | u_{y'})
\]  

(12)

rem.[Independence assumptions for POI model] Since in Equation (9), \( \alpha_y \) are independent and multivariate normal, it follows that \( h_y \) and \( h_{y'} \) are independent for \( y \neq y' \). Hence both Assumption 4.2 and Assumption 4.3 are fulfilled for our model. In Equation (23), we also show that \( h_y \) are also independent with respect to the variational approximation \( q \).

The following theorem derives the ELBO for additive latent Gaussian processes under sparse approximation.

**Theorem 4.4 (ELBO with inducing points for additive latent Gaussian processes).** Under Assumption 4.2 and Assumption 4.3, a sparse variational approximation \( q(h_y) \) (see \([15]\) for Definition) to the posterior distribution of \( p(h_y | y) \) from Equation (9) can be found by maximizing the following ELBO with respect to \( q(u_y) \) for \( y \in \mathcal{I}_0 \):

\[
 \log p(y) \geq \mathbb{E}_{q(h_0, h_y, \ldots, h_{y|\mathcal{I}})} \left[ \log p(y | h_0, h_y, \ldots, h_{y|\mathcal{I}}) \right] - \sum_{y \in \mathcal{I}_0} \text{KL} \left[ q(u_y) \parallel p(u_y) \right] ,
\]  

(13)

\(^7\)For readability, we ease notation by omitting the dependency of the posterior distribution from \( h_y \) on \( S, \{s_n\}_{n=1}^N, \mathcal{P}, \) and parameters \( \theta \).
where \( q(h_f) = \int q(u_f)p(h_f|u_f)du_f \) and where \( \text{KL}[q(u_f)||p(u_f)] \) denotes the Kullback-Leibler divergence between \( q(u_f) \) and \( p(u_f) \).

**Proof.** The proof is given in Appendix A. \( \square \)

We note that neither the additive structure of Equation (9d) nor any assumption on the variational distribution \( q \) are necessary for the results to hold. However, both will be used in the next sections. First, in Section 4.3, we assume multivariate normality of \( q \) to calculate the KL divergence and use the additive structure to analytically derive the variational bounds for particular likelihoods. Second, in Section 4.4 both assumptions are required to derive the variational approximation \( q(f) \) as well as predictions of \( y(s^*) \) at new locations \( s^* \).

### 4.3 Analytical Expressions of ELBO

The ELBO for our problem as stated in Equation (13) can be further simplified under additional assumptions:

1. We assume that, for \( f \in \mathbb{I}_n \), the variational approximation is a multivariate normal distribution, i.e., \( q(u_f) = N(u_f|m_f, S_f) \). This is the canonical choice, since we model \( h_f \) as a Gaussian process (see Lemma 4.1). The quality of the variational approximation, i.e., how well \( q(h_f) \) approximates \( p(h_f | y) \) can be assessed using generalized Pareto distribution diagnostics [46].

2. We exploit the specific likelihood structure in case of a Gaussian or Poisson distribution. Mathematically, we need to make two changes to Equation (13): (i) We replace the KL divergence with an explicit computational statement. We leverage the fact that the variational approximation is multivariate Gaussian and, hence, the problem reduces to the calculation of the KL divergence between two Gaussians, since \( p(u_f) \) has a Gaussian process prior. The detailed expression is given in Appendix B. (ii) We need to find an analytical expression for the variational expectation of the conditional log-likelihood. Here our initial assumption that \( y_n \) are independent given \( f \) comes into play, based on which we can factorize the joint conditional log-likelihood over the complete \( y \) and obtain a sum over

\[
\mathbb{E}[q(h_{f1}^n \cdots h_{fL}^n) | \log p(y_n | h_{f1}^n, \ldots, h_{fL}^n)].
\]

In case of a Gaussian or Poisson likelihood, which are common in many applications, there exist an analytical solution to the above expression. Explicit expressions are provided in Appendix B. These analytical expressions additionally aid the overall computational performance. For all other cases without an analytical solution, this quantity must be computed numerically.

### 4.4 Predictions

Under the same set of assumptions as in Equation (4.3), we can derive \( q(h_f^*) \), as well as \( q(f^*) \) at a new location \( s^* \). To make predictions at \( s^* \) for the latent function \( h_f^* \), we apply [15, cf. Eq. (21) therein] to obtain

\[
q(h_f^*) \approx \int p(h_f^* | u_f)q(u_f)du_f.
\]

This integral can be solved due to the fact that \( h_f^* \) and \( u_f \) follow the same Gaussian Process and that \( q(u_f) \) is Gaussian by assumption. It follows that \( q(h_f^*) \) is again Gaussian, see [34, Sec. A.2 therein]. Since \( f^* = \sum_{f \in \mathbb{I}_L} h_f^* \), the posterior distribution \( q(f^*) \) can be derived analogously.\(^9\)

### 4.5 Computational Complexity

The computational complexity of our learning algorithm can be derived analogously to [13–15]. It requires maximizing the statement in Equation (13), which comprises \( O(N^2) \) operations due to the expected variational log-likelihood and \( O(|\mathbb{I}|^2|M^3) \) operations due to computing the KL divergence. Notably, the inequality \( M \ll N \) should hold in practice, i.e., the computational effort of calculating the expected variational log-likelihood dominates. Following [14], the computation of Equation (13) allows parallelization across mini-batches of size \( N \) in stochastic gradient optimization. This results in a reduced overall complexity of \( O(NM^2) \).

For comparison, the exact calculation of the posterior distribution with MCMC would be computationally costly: each iteration of it entails a complexity of \( O(|\mathbb{I}|N^3) \). This complexity originates from the Gaussian processes and is a result of the inversion of \((N \times N)\)-dimensional covariance matrices.

### 5 EXPERIMENTS

#### 5.1 Datasets

We demonstrate the effectiveness of our POI model based on two real-world tasks, namely, (1) modeling prices of real estate property [5, 23, 44] and (2) predicting check-in counts as a form of customer visit forecasting.\(^9\) Both cases draw upon POIs in order to describe the spatial variation in the outcome variable. We used web mining based on the Google Maps API to obtain up-to-date POI data.\(^11\)

The choice of our case studies was driven by the aim to validate the POI model under different distributions: following common conventions, the log-prices are modeled as a Gaussian distribution and thus provide a case where we can experiment with our analytical solutions to the ELBO. Our second case study uses a Poisson distribution to model check-ins. Non-Gaussian likelihoods typically require expensive sampling methods.

##### 5.1.1 Case Study 1: Real Estate Prices

We analyze \( N = 6,196 \) house prices based on 43,559 POIs in Melbourne with an average property valuation of AU$ 1,068,828.\(^12\) Figure 1 visualizes the spatial distribution of the housing prices. The log-price is modeled as a Gaussian likelihood based on various property-specific features \( x_n \), namely, *number of bedrooms, bathrooms, parking spots, lot size, and size* (i.e., finished area). We include multiple POI types (Figure 1): *department store, movie theater, police station, train station*, and *university*. We omit POI-specific covariates \( \eta \) in this case study for simplicity.

##### 5.1.2 Case Study 2: Restaurant Check-Ins

We analyze the average number of daily check-ins from \( N = 2,925 \) restaurants based on 740

---

8Without loss of generality, we assume that \( q \equiv 0 \).

9This business problem was defined in https://www.kaggle.com/c/facebook-v-predicting-check-ins.

10https://developers.google.com/maps/documentation/.

11https://www.kaggle.com/anthonypino/melbourne-housing-market.
5.2 Baseline Models

Our choice of baselines is informed by common practice in spatial modeling [1] and earlier works with POI data. Accordingly, we performed feature engineering [21, 22, 37, 40, 43, 44] in order to translate POIs into statistical metrics from previous literature as follows.

- **Density-based features**: Given a maximal distance $D_{\text{max}}^y$, we count the relative or absolute number of POIs of type $y$ with distance $\|s_n - o\| < D_{\text{max}}^y$ for each observation $y_n$.

- **Distance-based features**: For each observation $y_n$ and POI type $y$, we calculate (1) the distance to the closest POI of type $y$ and (2) the sum over the distances to all POIs closer than a fixed distance $D_{\text{max}}^y$.

For all baseline models, the distance threshold $D_{\text{max}}^y$ is considered a hyperparameter that needs to be tuned using cross-validation. This is in contrast to our Bayesian framework, where this parameter is simultaneously learned with other model parameters.

The engineered features are subsequently fed into a variety of models that are common in spatial analytics, namely, a geographically weighted regression, a conditional auto-regressive, and kriging model. For ease of notation, we introduce the baseline models assuming a Gaussian likelihood, i.e., $y = f(x) + \epsilon$ with $\epsilon \sim N(0, \Sigma)$. However, all proposed models can be generalized to other likelihoods, e.g., a Poisson likelihood, using the maximum likelihood approach. For the baselines, the feature vector $x_n$ includes both the outcome-specific features as well as the features generated from the POIs.

5.2.1 Geographically Weighted Regression (GWR). GWR fits a locally varying coefficient $\beta(s_n)$ to a linear regression model of the form

$$y_n = x_n^T \beta(s_n) + \epsilon_n.$$  

(16)

The application of GWR is restricted to linear relationships. A detailed introduction can be found in [2].

5.2.2 Conditional Auto-Regressive Model (CAR). CAR models present a flexible class of spatial models that are typically used with areal data. Areal data differs from point data, since, for point data, the relationship between observations is continuous with respect to the distance, whereas, for areal data, it is defined in terms of adjacency. We apply the CAR models to point data by defining two points to be adjacent if their distance is below a threshold $\Delta$, which presents an additional hyperparameter which we tune via cross-validation. The CAR model is then given as

$$y_n = x_n^T \beta + \phi_n + \epsilon_n, \quad \phi \sim N(0, [D_\tau (I - \alpha D^\tau W)]^{-1}).$$  

(17)

where the parameter $\phi_n$ encodes the spatial dependence via the joint prior distribution on the right. Here the parameter $\alpha \in [0, 1)$ controls the level of spatial dependence. $W$ denotes the adjacency matrix $(w_{ij} = 0$ and $w_{ij} = w_{ji} = 1$ if locations $s_i$ and $s_j$ are adjacent), and $D_\tau = \tau D$ is a diagonal matrix with $d_{ii}$ the number of locations adjacent to $s_i$ scaled by $\tau$. Additional details on CAR models can be found in [8].

5.2.3 Kriging. In spatial analytics, (linear) kriging is a commonly used approach for modeling spatial dependence [1]. The model equation is given by

$$y_n = x_n^T \beta + \epsilon(s_n),$$  

(18)

where the error term $\epsilon$ depends on the location $s_n$ of $y_n$. The linear function $x_n^T \beta$ may be replaced by a more complex function $m(x)$ depending on the complexity of the features. In our case studies, we utilized random forest kriging in addition to linear kriging.

The model in Equation (18) is fitted using a two-step procedure. First, the linear component is fitted globally (assuming i.i.d. errors). In a second step, the residuals are fitted based on a spatially-dependent covariance structure. Hence, no theoretical foundation for the obtained estimates is available. We also note that the interpretation of coefficients obtained from a two-step procedure is debatable. We note that co-kriging is not applicable in our situation, as we only observe a single outcome variable $y$ and the assumed POI processes $h_x$ are not observed.

5.3 Case Study 1: Modeling Prices of Real Estate Property

5.3.1 Model Specification. In this case study, our POI model is parameterized as follows. The likelihood is set to a Gaussian likelihood, which allows us to benefit from our analytical solution to the ELBO. The feature influence $f$ is set to a linear function $x_n^T \theta_x$. On the one hand, this aids the possibility for interpretation and, on the other hand, it is analogous to the baselines in spatial modeling, thus giving a fair comparison. The kernel inside $h_y$ for weighting the POI influence by distance is set to a rectified linear unit (ReLU), again for easier interpretation. According to it, the POI has a linear effect up to a maximum distance $\theta_y$, after which it vanishes. Formally, we obtain

$$k_y(s, \omega; \theta_y) = \text{ReLU} \left( 1 - \frac{\|s - \omega\|}{\theta_y} \right).$$  

(19)

For $\alpha = 1$, the prior distribution is improper. However, the corresponding intrinsic conditional auto-regressive (ICAR) model is nevertheless widely used in practice.
Furthermore, we require the model to learn an individual but independent effect \( \alpha_p \) for each POI and, hence, choose \( \Sigma_\gamma = \sigma_\gamma^2 I \). The remaining spatial variation \( h_0 \) is set to a Matérn (3/2) kernel, which is common in spatial modeling [1, 28]. The corresponding GP is only twice differentiable and is able to capture wide spatial structures.

### 5.3.2 Performance

To compare the performance of the models we report the out-of-sample log-likelihood and the root mean squared error (RMSE). Table 1 (left) shows the out-of-sample performance. The proposed POI model clearly outperforms all baselines.

| Model [feature engineering] | (1) Real estate prices | (2) Restaurant check-ins |
|-----------------------------|------------------------|-------------------------|
| GWR [density]               | RMSE 0.488             | Log-lik. 530.188        |
| GWR [distance]              | 0.511                  | 1.010                   |
| CAR [density]               | 0.766                  | 1.117                   |
| CAR [distance]              | 0.872                  | 1.145                   |
| Linear kriging [density]    | 0.614                  | 1.095                   |
| Linear kriging [distance]   | 0.593                  | 1.098                   |
| Proposed POI model [n/a]    | 0.443                  | 564.014                 |

Table 1: Out-of-sample performance highlights the superiority of our proposed POI model.

### 5.3.3 Interpretation

In many applications, interpretability and knowledge discovery are as important as predictive power. In the following, we demonstrate the vast range of insights that can be derived from our POI model. Figure 2 (left) shows the predictive posterior distribution of the influence function \( h_\gamma \) for POI type \textit{train station}. Evidently, higher prices correspond to closer proximity to the train network. Figure 2 (right) depicts the remaining spatial heterogeneity \( h_0 \). It controls for and confirms the fact that a city center generally entails higher housing prices than suburban regions.

Figure 2: \textit{Left:} Recovered latent POI influence \( h_\gamma \) for POI type “train station” on housing prices. \textit{Right:} Remaining spatial variation \( h_0 \).

Table 2 describes the nature of the POI influence \( h_\gamma \): the estimated average effect size and the maximum distance \( \theta_\gamma \) for which an effect is measurable. We observe that movie theaters have the largest effect size. Although the effect size of universities is relatively small, it exhibits the largest distance. The estimated shape of the kernel \( k_\gamma \) is additionally visualized in Figure 3. In contrast to feature engineering, our model provides a detailed understanding of the POI influence as a function of the distance.

![Figure 3: Estimated kernels \( k_\gamma \) for real estate pricing (left) and restaurant check-ins (right). It shows the maximum radius out to which the outcome is affected by a POI location.](image)

### 5.4 Case Study 2: Predicting Restaurant Check-Ins

#### 5.4.1 Model Specification

Check-ins are widely modeled as Poisson distributions and, hence, we also use a Poisson likelihood. The feature influence \( g \) is again a linear function \( x^T \theta_x \), and thus matches the baselines from spatial analytics. The kernel is analogous to the previous case study; however, we additionally leverage the visitor counts \( \eta \) at the POIs and, for this reason, incorporate \( \eta \) as an additional scaling factor, i.e.,

\[
k_\gamma(s, \omega; \eta; \theta_\gamma) = \eta \text{ReLU}\left(1 - \frac{||s - \omega||}{\theta_\gamma}\right) \quad (20)
\]

We again fit an individual but independent effect \( \alpha_p \) for each POI. Therefore, and as a result of the inclusion of \( \eta = \{\eta_p\}_{p \in P} \), we obtain \( \Sigma_\gamma = \sigma_\gamma^2 \text{diag}(\eta_1, \ldots, \eta_{|P|}) \). The spatial kernel was chosen as in the previous case study.

#### 5.4.2 Performance

We again evaluate the out-of-sample performance against a series of baselines. The results are stated in Table 1 (right) and reveal the superiority of our approach. Our POI model is ranked first in terms of likelihood and second with respect to the RMSE.

#### 5.4.3 Interpretation

Table 3 lists the characteristics of the estimated POI influence. It spans a maximum radius of between 270 m and 510 m, with monuments exhibiting the largest effects. In addition, the posterior POI influence is exemplified in Figure 4. As expected, we observe a larger number of restaurant check-ins in crowded areas such as Grand Central Station and Penn Station, the two main public transport hubs in Manhattan. In order to better identify their relevance, we show map of the posterior influence in Figure 5. Here, we observe that the influence area of Grand Central is slightly larger than that of Penn Station. This matches our initial expectations that Grand Central is the more popular station for commuters. Figure 3 (right) visualizes the shape of the estimated kernel \( k_\gamma \). We observe shorter distances compared to the real estate

### Table 2: Estimated parameters characterizing the structure of POI influence for real estate prices (case study (1)), i.e., average effect size \( \alpha_\gamma \) and affected radius \( \theta_\gamma \) per POI type.

| POI type (\( \gamma \)) | Average size \( \bar{\gamma} \) | SD \( \sigma_\gamma \) | Cut-off \( \theta_\gamma \) [in km] |
|------------------------|-------------------------------|-----------------|-----------------|
| Police station         | 0.000                         | 0.013           | 0.399           |
| Department store       | 0.001                         | 0.014           | 0.305           |
| University             | 0.037                         | 0.223           | 1.499           |
| Train station          | 0.075                         | 0.202           | 1.451           |
| Movie theater          | 0.085                         | 0.268           | 1.228           |

---

C. Naumzik et al.
prices case study, as restaurant check-ins might only be influenced by POIs in walking distances [37].

Table 3: Estimated parameters characterizing the structure of POI influence for restaurant check-ins (case study (2)), i.e., average effect size $\bar{\alpha}_y$ and affected radius $\bar{\theta}_y$ per POI type.

| POI type ($y$) | Effect size | Effect distance |
|----------------|-------------|-----------------|
|               | Average size $\bar{\alpha}_y$ | SD $\sigma_y$ | Cut-off $\bar{\theta}_y$ [in km] |
| Museum        | 0.004       | 0.218           | 0.451 |
| Stadium       | 0.000       | 0.022           | 0.420 |
| Police station | 0.001       | 0.153           | 0.271 |
| University    | 0.004       | 0.208           | 0.396 |
| Shopping      | 0.009       | 0.252           | 0.360 |
| Movie theater | 0.011       | 0.180           | 0.514 |
| Monument      | 0.018       | 0.251           | 0.499 |
| Train station | 0.020       | 0.230           | 0.450 |

Table 4: Performance comparison across model variations for case study (1) with real estate prices.

| Model variation | RMSE  | Log-lik |
|-----------------|-------|---------|
| Variation: kernel |      |         |
| Linear $g$, ReLU kernel | 0.443 | -564.014 |
| Linear $g$, Gaussian kernel | 0.440 | -520.312 |
| Variation: non-linear feature influence |   |         |
| Neural network $g$, ReLU kernel | 0.431 | -479.954 |
| Non-linear baseline: | | |
| Random forest kriging | 0.470 | -548.878 |
| Neural baseline | 0.455 | -484.86 |

Table 4: Performance comparison across model variations for case study (1) with real estate prices.

6 DISCUSSION

In short, our main objective is to model the urban influence of POIs $P$ on an outcome variable $y$. For this purpose, a model is needed that ensures interpretability. As demonstrated, our approach scales easily to 10,000s of POIs that one encounters when modeling urban phenomena within a city of interest. Owing to our tailored model, the prediction performance of other linear models is outperformed; and, when extending to non-linearities, the neural baseline is also surpassed.

Contributions: Our proposed spatial inferences with POI locations are of considerable value to both researchers and practitioners. This includes all use cases from Section 1 modeling the influence of POIs on variables from urban planning. Here, our POI model promises to retrieve valuable explanatory insights paired with high predictive capacity. This is achieved by overcoming existing limitations of feature engineering used in the state-of-the-art approaches. Instead, we propose a comprehensive POI model: we consider the actual spatial distribution of POIs, whereby their influence is allowed to vary by distance and does not rely on hyperparameter tuning. Despite the fact that probabilistic modeling is naturally challenging in terms of computational complexity, we establish that our learning algorithm is highly scalable and can be effectively applied to large-scale data.

Guidelines: Our model provides flexibility as the underlying likelihood (i.e., the distribution of $y$) can be flexibly chosen. However, we remind that this is not needed: users of our model can simply follow common conventions in spatial modeling $[2, 6, 8]$ and, for all continuous variable, choose a normal distribution. If count data is modeled, one analogously chooses a Poisson likelihood. In practice, this ensures a straightforward use.

Extensibility: As repeatedly demonstrated in our experiments, the proposed variational inference seamlessly adapts to various model and kernel specifications. The likelihood can be updated depending on the outcome distribution.

Non-linearities. Our POI model is not restricted to a linear feature influence $g$ as are GWR and CAR from spatial analytics. On the contrary, the linear structure can be easily relaxed by a different $g$. Here we opted for a simple fully-connected neural network. As a result, the out-of-sample performance improves and, in comparison to random forest kriging and a deep neural network as non-linear baselines, our POI model remains superior.

5.5 Extensibility

A benefit of our Bayesian model formulation is its flexibility. Table 4 reports additional results where we vary the model configuration:

- Alternative kernels. We demonstrate the flexibility of our POI model under alternative kernels. We previously chose a ReLU kernel for its expressive power. However, our POI model can be utilized with arbitrary kernels. To demonstrate this, we repeat our previous experiment with a traditional Gaussian kernel, i.e.,

$$k_y(s, \omega; \theta_y) = \exp\left(-\frac{||s - \omega||^2}{2\theta_y^2}\right), \quad (21)$$

resulting in a fairly similar model fit.

The structure was kept overly simple (i.e., ReLU activation with 1 hidden layer and 4 neurons) but, despite that fact, the results are already favorable.
Nevertheless, our experiments have revealed that the overall performance remains stable under different kernels. Hence, we recommend to use an interpretable kernel.

Limitations: In comparison to GWR, CAR, and (co-)kriging, our work makes fairly little assumptions, namely, (1) independence of the observation $y_n$ given the function $f$ and (2) multivariate normality and independence of the POI effects $\alpha_i$. These assumptions are owed to the learning algorithm in order to provide efficient inference. Nevertheless, an inherent theoretical limitation remains when $|\mathcal{P}|$ approaches $N$ (e.g., a large number of POIs but few observations), which results into overfitting and requires subsampling of $\mathcal{P}$ or alternative means such as spatial smoothing.

A ELBO FOR LATENT ADDITIVE GAUSSIAN PROCESSES

Proof of Theorem 4.4 Without loss of generality, we assume that $g \equiv 0$ and $|\Gamma| = 1$. Let $h_0$ and $h_1$ denote the values of the two latent Gaussian processes at the observed locations $S = \{s_n\}_{n=1}^N$. The corresponding augmented outputs are given as $u_0$ and $u_1$ at inducing points $Z = \{z_m\}_{m=1}^M \subseteq S$.

Using Jensen’s inequality, we derive a first lower bound for the likelihood of $y$ given the augmented outputs $u_0$ and $u_1$ via

$$\log p(y|u_0, u_1) = \log \int p(y|h_0, h_1, u_0, u_1) \frac{p(u_0, u_1)}{p(u_0)} \, dh \geq E_{p(h_0, h_1|u_0, u_1)} [\log p(y|h_0, h_1)].$$

(22)

Drawing upon Jensen’s inequality together with Assumption 4.2, we yield the following inequality

$$\log p(y) = \log \int p(y|u_0, u_1) q(u_0, u_1) \frac{p(u_0, u_1)}{q(u_0, u_1)} \, du \geq \mathbb{E}_{q(u_0, u_1)} [\log p(y|u_0, u_1)] - \text{KL}[q(u_0) \| p(u_0)] - \text{KL}[q(u_1) \| p(u_1)].$$

With $Z \subseteq S$ and Assumption 4.3, we finally obtain

$$q(h_0, h_1) = \int p(h_0, h_1|u_0, u_1) q(u_0, u_1) \, du = q(h_0) q(h_1).$$

(23)

and the ELBO for latent additive Gaussian processes, i.e.,

$$\log p(y) \geq \mathbb{E}_{q(h_0, h_1)} [\log p(y|h_0, h_1)] - \text{KL}[q(u_0) \| p(u_0)] - \text{KL}[q(u_1) \| p(u_1)].$$

(24)

B ANALYTICAL EXPRESSIONS OF ELBO

Analytical solutions to the ELBO from Equation (13) require expressions for the KL divergence and tailored derivations of the conditional log-likelihood. For this, we first report the KL divergence under the assumption of a Gaussian variational distribution; and, second, we derive the variational expectation of the conditional log-likelihood.

B.1 ELBO Simplification Under a Gaussian Variational Distribution

The ELBO as stated in Equation (13) can be further simplified if we assume that, for $\gamma \in \Gamma_0$, the variational approximation is a multivariate normal distribution, i.e., $q(u_\gamma) = N(u_\gamma | m_\gamma, S_\gamma)$. Note that $u_\gamma$ and $h_\gamma$ are drawn from the same Gaussian process prior; hence, the joint distribution can be written as

$$p(h_\gamma, u_\gamma) = N \left[ \begin{array}{c} 0 \\ K_{y \gamma n} \\ K_{y \gamma m} \\ K_{y \gamma m} \end{array} \right].$$

(25)

where $K_{y \gamma n}$ and $K_{y \gamma m}$ denote the kernel function $k_\gamma$ evaluated at $S \times S$, $S \times Z$, and $Z \times Z$, respectively. Hence $p(u_\gamma) = N(u_\gamma | 0, K_{y \gamma m}).$

B.1.1 Kullback-Leibler Divergence. Using these results, the KL divergences between $q(u_\gamma)$ and $p(u_\gamma)$ can be calculated as

$$\frac{1}{2} \text{tr} (K_{y \gamma m} S_\gamma) + m_\gamma^T K_{y \gamma m} m_\gamma - M + \log \left( \frac{\det K_{y \gamma m}}{\det S_\gamma} \right).$$

(26)

See [34, p. 203].

B.1.2 Variational Expectation of Conditional Log-Likelihood. In order to calculate the expectation of the logarithm of the conditional likelihood in Equation (13), we need to calculate the variational approximation to the posterior for $h_\gamma$. Since $q(u_\gamma)$ is Gaussian, it follows that $q(h_\gamma) = \int p(h_\gamma|u_\gamma) q(u_\gamma) \, du_\gamma$ is also Gaussian. In particular, for $q(u_\gamma) = N(u_\gamma | m_\gamma, S_\gamma)$, we yield

$$q(h_\gamma) = N(h_\gamma | A_\gamma m, K_{y \gamma n} + A_\gamma (S_\gamma - K_{y \gamma m}) A_\gamma^T),$$

(27)

where $A_\gamma = K_{y \gamma m} K_{y \gamma n}^{-1}$.

We recall that the likelihood factorizes given the latent function $f$. Denoting the marginal distribution $q(h_\gamma)$ as $q(h_\gamma)$ yields

$$\mathbb{E}_{q(h_\gamma, h_{\gamma_1}, \ldots, h_{\gamma_N})} [\log p(y|h_0, h_1, \ldots, h_N)] = \sum_{n=1}^N \mathbb{E}_{q(h_n^m)} [\log p(y_n|h_0, \ldots, h_{\gamma_1}^m, \ldots, h_{\gamma_N}^m)].$$

(28)

Now the evaluation of the last equation depends on the chosen likelihood. For some likelihoods e.g., the Gaussian or Poisson, the expectation can be solved analytically (see Appendix B.2).

B.2 Simplifications under Gaussian and Poisson Likelihood

To simplify notation, we write $q(h_\gamma) \sim N(\tilde{m}_\gamma, \tilde{S}_\gamma)$ with $\tilde{m}_\gamma = (\tilde{m}_\gamma^1, \ldots, \tilde{m}_\gamma^N)$ and $\tilde{S}_\gamma = \{\tilde{s}_\gamma^{ij} \}_{i,j=1}^N$ given by Equation (27). For a Gaussian likelihood $p(y_n | f_n) = N(y_n | f_n, \sigma^2)$, the $n$-th summand of the conditional log-likelihood Equation (28) is given by

$$-\frac{1}{2} \left[ \log (2\pi) + \log (\sigma^2) + \frac{1}{\sigma^2} \left( y_n - \sum_{y \in i} \tilde{m}_\gamma^y + \sum_{y \in i} \tilde{s}_\gamma^{yn} \right)^2 \right].$$

(29)

In the Poisson case $p(y_n | f_n) = \text{Poisson}(y_n | \lambda_n)$ with $\lambda_n = \exp(f_n)$, the $n$-th summand of the conditional log-likelihood Equation (28) is given by

$$-\log(y_n!) - y_n \sum_{y \in i} \tilde{m}_\gamma^y - \exp \left( \sum_{y \in i} \tilde{m}_\gamma^y + \frac{1}{\sigma^2} \sum_{y \in i} \tilde{s}_\gamma^{yn} \right).$$

(30)

The derivations of Equation (29) and Equation (30) follow by direct calculation.
[51] Jing Yuan, Yu Zheng, and Xing Xie. 2012. Discovering regions of different functions in a city using human mobility and POIs. In ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD). 186–194. https://doi.org/10.1145/2339530.2339561

[52] Jia-Dong Zhang and Chi-Yin Chow. 2015. Spatiotemporal sequential influence modeling for location recommendations. ACM Transactions on Intelligent Systems and Technology 7, 1 (2015), 1–25. https://doi.org/10.1145/2786761

[53] Shenglin Zhao, Tong Zhao, Irwin King, and Michael R. Lyu. 2017. Geo-Teaser: Geo-temporal sequential embedding rank for point-of-interest recommendation.

[54] Ning Zheng, Xiaoming Jin, and Lianghao Li. 2013. Cross-region collaborative filtering for new point-of-interest recommendation. In International Conference on World Wide Web (WWW).

[55] Fan Zhou, Ruiyang Yin, Kunpeng Zhang, Goce Trajcevski, Ting Zhong, and Jin Wu. 2019. Adversarial point-of-interest recommendation. In The Web Conference (WWW).