

Identifying Sources and Sinks in the Presence of Multiple Agents with Gaussian Process Vector Calculus

Adam D. Cobb
University of Oxford
Oxford, United Kingdom
acobb@robots.ox.ac.uk

Andrew Markham
University of Oxford
Oxford, United Kingdom
andrew.markham@cs.ox.ac.uk

Richard Everett
University of Oxford
Oxford, United Kingdom
richard@robots.ox.ac.uk

Stephen J. Roberts
University of Oxford
Oxford, United Kingdom
sjrob@robots.ox.ac.uk

ABSTRACT

In systems of multiple agents, identifying the cause of observed agent dynamics is challenging. Often, these agents operate in diverse, non-stationary environments, where models rely on hand-crafted environment-specific features to infer influential regions in the system’s surroundings. To overcome the limitations of these inflexible models, we present *GP-LAPLACE*, a technique for locating sources and sinks from trajectories in time-varying fields. Using Gaussian processes, we jointly infer a spatio-temporal vector field, as well as canonical vector calculus operations on that field. Notably, we do this from only agent trajectories without requiring knowledge of the environment, and also obtain a metric for denoting the significance of inferred causal features in the environment by exploiting our probabilistic method. To evaluate our approach, we apply it to both synthetic and real-world GPS data, demonstrating the applicability of our technique in the presence of multiple agents, as well as its superiority over existing methods.

CCS CONCEPTS

• **Computing methodologies** → **Gaussian processes**; *Unsupervised learning*; *Multi-agent systems*; • **Mathematics of computing** → *Bayesian nonparametric models*; • **Applied computing**;

KEYWORDS

Gaussian processes; potential fields; multi-agent systems; GPS data; animal tracking

ACM Reference Format:

Adam D. Cobb, Richard Everett, Andrew Markham, and Stephen J. Roberts. 2018. Identifying Sources and Sinks in the Presence of Multiple Agents with Gaussian Process Vector Calculus. In *KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, August 19–23, 2018, London, United Kingdom*. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3219819.3220065>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

KDD '18, August 19–23, 2018, London, United Kingdom

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-5552-0/18/08...\$15.00

<https://doi.org/10.1145/3219819.3220065>

1 INTRODUCTION

Inferring the possible cause of an agent’s behaviour from observing their dynamics is an important area of research across multiple domains [2]. For example, in ecology, the increasing availability of improved sensor technology and GPS data enables us to learn the motivations behind animal movement [9, 31], assisting with animal conservation and environmental efforts. Other domains, such as robotics, use apprenticeship learning to construct reward functions to allow them to mimic their observations [1].

Typical solutions for inferring the cause of an agent’s behaviour tend to exploit Inverse Reinforcement Learning (IRL) [25] and include learning preference value functions [7] as well as utility, value, or policy surfaces from observed actions in a space. Unfortunately, these solutions often rely on the problem being easily framed as a Markov decision process [21] which is not always appropriate. Instead, one may be interested in identifying an interpretable potential function, defined in continuous space, that can explain trajectories made by agents in a multi-agent system (MAS). This approach can be seen in previous work where agents are modelled as particles with their dynamics determined by a potential field [4, 5, 20].

In this paper, we consider observed trajectories influenced by a time- and space-varying potential function, and infer the spatio-temporal potential function from observed movement alone. To accomplish this, we present *GP-LAPLACE*: Gaussian Processes for Locating Attractive PLACES¹. By constructing Gaussian processes (GPs) [23], our method jointly infers a spatio-temporal vector field as well as canonical vector calculus operations on that field, allowing us to estimate a time-varying map of sources and sinks of potential influence on agents.

There are three notable advantages to our approach. First, it is able to reason about the interaction between agents and the environment from only agent trajectories, without requiring knowledge of the environment. Second, it allows the potential field to be non-stationary, which more accurately reflects the real-world. Finally, by exploiting a probabilistic method, we obtain a metric for denoting the significance of inferred causal features in the environment.

To demonstrate the generality of our method as a tool for explaining agent and animal behaviour, we apply it to two distinct data sets. As an illustrative example, we evaluate our approach on a synthetic data set which we can compare to the true potential

¹Code and data: <https://github.com/AdamCobb/GP-LAPLACE>

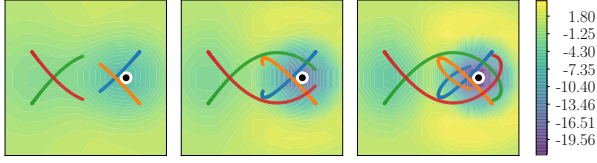


Figure 1: GP-LAPLACE tracking a time-varying attractor from observing four agent trajectories. The black and white marker indicates the true location of an attractor that is increasing in strength.

function (which isn't accessible in the real-world). Next, motivated by the existence of long-term GPS data, we apply our method to a real-world GPS data set of pelagic seabirds [19], discovering a number of attractors which influence their behaviour.

The rest of our paper is organised as follows: Section 2 introduces the GP and its derivative manipulations for vector calculus, followed by Section 3 where we present our model. In Section 4 we evaluate our method on a synthetic data set where we know the true potential function. We then demonstrate how our approach can interpret real-world data by applying it to the Scolopi's shearwater GPS data set in Section 5. Finally, we highlight the novelty of our work compared to the existing literature in Section 6 before concluding with future work in Section 7.

2 PRELIMINARIES

As a requirement for our model, we introduce the Gaussian process, defined by its mean function $\mu(\mathbf{x})$ and covariance function $\mathbf{K}(\mathbf{x}, \mathbf{x}')$ [23]. The mean and covariance functions are parameterised by their hyperparameters and encode prior information into the model, such as smoothness, periodicity and any known underlying behaviour. In our work, $\mu(\mathbf{x})$ is set to zero as our data is preprocessed to have a zero mean. We define a function, distributed via a GP, as follows:

$$f(\mathbf{x}) \sim \mathcal{GP}(\mu(\mathbf{x}), \mathbf{K}(\mathbf{x}, \mathbf{x}')). \quad (1)$$

Manipulating Gaussian identities using Bayes' rule gives formulae for the GP's posterior mean,

$$\mathbb{E}[f(\mathbf{x}_*)] = \mathbf{k}_{\mathbf{x}\mathbf{x}_*}^\top (\mathbf{K}_{\mathbf{x}\mathbf{x}} + \sigma^2 \mathbf{I})^{-1} \mathbf{y}, \quad (2)$$

and posterior covariance,

$$\mathbb{V}[f(\mathbf{x}_*)] = k_{\mathbf{x}\mathbf{x}_*} - \mathbf{k}_{\mathbf{x}\mathbf{x}_*}^\top (\mathbf{K}_{\mathbf{x}\mathbf{x}} + \sigma^2 \mathbf{I})^{-1} \mathbf{k}_{\mathbf{x}\mathbf{x}_*}, \quad (3)$$

where \mathbf{x}_* is a test point under question and σ^2 is the noise variance hyperparameter.

Any affine transformation of Gaussian distributed variables remain jointly Gaussian distributed. As differentiation is an affine operation, applying this property to any collection of random variables distributed by a GP, gives jointly Gaussian distributed derivatives, $\mathbf{f}'(\mathbf{x})$. For a test point \mathbf{x}_* and corresponding output $f(\mathbf{x}_*)$, the derivatives associated with the GP in Equation (1) are distributed with posterior mean,

$$\mathbb{E}\left[\frac{\partial^n f(\mathbf{x}_*)}{\partial \mathbf{x}_*^n}\right] = \frac{\partial^n \mathbf{k}_{\mathbf{x}\mathbf{x}_*}^\top}{\partial \mathbf{x}_*^n} (\mathbf{K}_{\mathbf{x}\mathbf{x}} + \sigma^2 \mathbf{I})^{-1} \mathbf{y}, \quad (4)$$

and posterior covariance,

$$\mathbb{V}\left[\frac{\partial^n f(\mathbf{x}_*)}{\partial \mathbf{x}_*^n}\right] = \frac{\partial^{2n} k_{\mathbf{x}\mathbf{x}_*}}{\partial \mathbf{x}_*^n \partial \mathbf{x}_*^n} - \frac{\partial^n \mathbf{k}_{\mathbf{x}\mathbf{x}_*}^\top}{\partial \mathbf{x}_*^n} (\mathbf{K}_{\mathbf{x}\mathbf{x}} + \sigma^2 \mathbf{I})^{-1} \frac{\partial^n \mathbf{k}_{\mathbf{x}\mathbf{x}_*}}{\partial \mathbf{x}_*^n}. \quad (5)$$

We define Equation (4) as the predictive mean of the n^{th} derivative with respect to any test points \mathbf{x}_* and Equation (5) as its corresponding variance.

The choice of covariance selected throughout this paper is the squared exponential kernel,

$$k_{\text{SE}}(\mathbf{x}_*, \mathbf{x}_i) = l^2 \exp\left(-\frac{1}{2}(\mathbf{x}_* - \mathbf{x}_i)^\top \Lambda^{-1}(\mathbf{x}_* - \mathbf{x}_i)\right), \quad (6)$$

with \mathbf{x}_* and \mathbf{x}_i corresponding to a test point and training point respectively. The hyperparameter Λ , is a diagonal matrix of input scale lengths, where each element determines the relevance of its corresponding dimension. The output scale, denoted by l^2 , controls the magnitude of the output [24]. The choice of kernel is motivated by the desire to obtain smooth measures over arbitrary derivative functions and the ease by which the kernel can be differentiated and used in combination with Equations (4) and (5). The following formulae define the squared exponential kernel for the first and second order derivatives [17]:

$$\frac{\partial k_{\text{SE}}(\mathbf{x}_*, \mathbf{x}_i)}{\partial \mathbf{x}_*} = -\Lambda^{-1}(\mathbf{x}_* - \mathbf{x}_i) k_{\text{SE}}(\mathbf{x}_*, \mathbf{x}_i) \quad (7)$$

$$\frac{\partial^2 k_{\text{SE}}(\mathbf{x}_*, \mathbf{x}_i)}{\partial \mathbf{x}_*^2} = \Lambda^{-1} \left((\mathbf{x}_* - \mathbf{x}_i)(\mathbf{x}_* - \mathbf{x}_i)^\top \Lambda^{-1} - \mathbf{I} \right) k_{\text{SE}}(\mathbf{x}_*, \mathbf{x}_i). \quad (8)$$

We note at this point that estimating derivatives using a joint GP model over the function and derivatives [6, 10] offers a benign noise escalation in comparison to numerical differentiation.

2.1 Vector calculus with GPs

We define a scalar potential function $\phi(\mathbf{x}, t)$ of space \mathbf{x} and time t . Furthermore, we define the time-dependent gradient of $\phi(\mathbf{x}, t)$ according to

$$\nabla_{\mathbf{x}} \phi(\mathbf{x}, t) = \mathbf{V}_t \quad (9)$$

with respect to \mathbf{x} , where $\mathbf{V}_t = [V_x \ V_y]^\top$ for $\mathbf{x} \in \mathbb{R}^2$. We can model this time-varying vector value function by a multi-input, multi-output GP with a three-dimensional input tuple consisting of $\mathbf{X} = (\mathbf{x}, t)$. This GP is constructed by introducing a separable kernel [3], such that

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} \sim \mathcal{GP}\left(\begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} k_x(\mathbf{X}, \mathbf{X}') & 0 \\ 0 & k_y(\mathbf{X}, \mathbf{X}') \end{bmatrix}\right)$$

contains an independently learned kernel for each output dimension. Applying Equation (4) to this GP model, by jointly inferring the derivatives in the x and y directions, gives the time dependent posterior for each random variable in the following tuple:

$$\left(V_x, V_y, \frac{\partial V_x}{\partial x}, \frac{\partial V_y}{\partial y}, \frac{\partial V_x}{\partial y}, \frac{\partial V_y}{\partial x}\right)_t.$$

We combine these predictive derivatives using Equations (10) and (11) to infer probability distributions over the divergence and

curl of the vector field² \mathbf{V} :

$$\nabla \cdot \mathbf{V} = \frac{\partial V_x}{\partial x} + \frac{\partial V_y}{\partial y}, \quad (10)$$

$$\nabla \times \mathbf{V} = \begin{vmatrix} \hat{\mathbf{i}} & \hat{\mathbf{j}} & \hat{\mathbf{k}} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ V_x & V_y & 0 \end{vmatrix} = \hat{\mathbf{k}} \left(\frac{\partial V_y}{\partial x} - \frac{\partial V_x}{\partial y} \right). \quad (11)$$

Additionally, by simple application of the appropriate operators, we may readily define the time-varying spatial Laplacian:

$$\nabla \cdot (\nabla \phi) \equiv \nabla^2 \phi. \quad (12)$$

This (time-varying) Laplacian is of key importance as it defines *sources* and *sinks* in the spatial domain. It can be thought of as indicative of the flow in a vector field. Positive values denote attractive regions, where the vector field indicates flow towards these regions, whereas negative values denote repulsive regions due to the vector field pointing in the opposite direction.

3 MODEL

Our model builds upon the theory introduced in Section 2. The objective is to design a model that can indicate influential features in an agent's environment from observing their trajectories.

A trajectory ζ_a for agent a is defined as a collection of time-stamped locations $\mathbf{x} \in \mathcal{R}^2$ or tuples (\mathbf{x}, t) . The elements of the vector \mathbf{x} are referred to as x and y for the purposes of this model and we continue to use the tuple $\mathbf{X} = (\mathbf{x}, t)$ to refer to the domain of space and time. We also make the assumption that each agent acts according to a utility or potential function $\phi(\mathbf{x}, t)$, which is dependent on space and time. Whilst interacting with the environment, each agent aims to maximise this utility function at all times.

3.1 Fitting to the agent trajectories

The first component of GP-LAPLACE uses Equations (2) and (3) to fit a GP to each trajectory ζ_a . Using a single GP with a separable kernel, Equation (13) defines our GP prior over the x and y components of the path $\vec{\mathbf{f}}$:

$$p(\vec{\mathbf{f}}) = \mathcal{GP} \left(\mathbf{0}, \begin{bmatrix} k_x(t, t') & 0 \\ 0 & k_y(t, t') \end{bmatrix} \right). \quad (13)$$

For each of the x and y components of $\vec{\mathbf{f}}$, a set of hyperparameters are learnt for the separable kernel. The input space of this GP model is time t and the output space consists of the x and y components of $\vec{\mathbf{f}}$. If an agent trajectory, ζ_a , consists of N data points, we can then further apply Equation (4) to infer higher order derivatives for each of the n data points, where we denote $\{\dot{f}_x, \dot{f}_y\}$ and $\{\ddot{f}_x, \ddot{f}_y\}$ as the first and second-order time derivatives in the x and y directions.

Second-order derivatives are inferred at this stage, as we make the assumption that an agent acts in accordance with a second-order dynamical system, i.e. the agent obeys Newtonian dynamics. This assumption means that an agent's underlying utility induces a "force" of influence on the agent, thus generating an acceleration (we here take the 'influence-mass' of the agent as unity). More

formally, this induces an acceleration equal to the derivative of the agent utility or potential:

$$\ddot{f}_x = \frac{\partial \phi}{\partial x}, \quad \ddot{f}_y = \frac{\partial \phi}{\partial y}. \quad (14)$$

Although we choose to infer second-order derivatives, the model is not limited to the assumption in Equation (14). The flexibility of our model means that we can also infer first-order terms, \dot{f}_x and \dot{f}_y , along with other higher-order terms. Therefore, throughout the rest of this section, references to \ddot{f}_x and \ddot{f}_y can be considered easily interchangeable with these other derivative terms.

When dealing with a multi-agent system of M homogeneous agents, a trajectory model can be calculated for each agent to form the set of joint distributions,

$$\left\{ p(\vec{\mathbf{f}}, \dot{f}_x, \dot{f}_y, \ddot{f}_x, \ddot{f}_y \mid \zeta_a) \right\}_{a=0}^M.$$

From the posterior GP model we are able to jointly predict the velocity and acceleration at any point on an agent's trajectory for M agents. At this stage, we now have a collection of posterior derivatives and their corresponding location in \mathbf{X} . If each of the agent trajectories has length N , then the size of \mathbf{X} is $M \times N$. The next layer of our model combines the outputs from the set of posterior distributions to construct a probability distribution over the extended agent environment.

3.2 Inferring the vector field and Laplacian

In order to infer the gradient of the potential function, $\nabla \phi(\mathbf{x}, t)$, the set of inferred second derivatives for all M agents is propagated through a second GP, which also has a separable kernel model, as below:

$$p(\vec{\mathbf{V}}(\mathbf{x}, t) \mid \{\ddot{f}_x, \ddot{f}_y, \zeta_a\}_{a=0}^M) = \mathcal{GP} \left(\begin{bmatrix} \mu_x^{\text{post}} \\ \mu_y^{\text{post}} \end{bmatrix}, \begin{bmatrix} k_x^{\text{post}}(\mathbf{X}, \mathbf{X}') & 0 \\ 0 & k_y^{\text{post}}(\mathbf{X}, \mathbf{X}') \end{bmatrix} \right). \quad (15)$$

The vector $\vec{\mathbf{V}}(\mathbf{x}, t) = [V_x \ V_y]^\top$ consists of two random variables that model the acceleration in the two axes and the superscript label 'post' refers to the calculated posterior mean and covariance. Equation (15) combines the M multiple agent paths into one model and enables predictions to be made at different points in space that are not constrained to a single agent trajectory as in Equation (13). The input-output pairs for this GP model are the x, y and t values in each ζ_a that correspond to the \ddot{f}_x and \ddot{f}_y values.

The Newtonian assumption made in Section 3.1 is formally included as

$$\nabla \phi(\mathbf{x}, t) \propto \vec{\mathbf{V}}(\mathbf{x}, t).$$

The distribution over the partial derivatives,

$$\left[\frac{\partial V_x}{\partial x}, \frac{\partial V_y}{\partial y}, \frac{\partial V_x}{\partial y}, \frac{\partial V_y}{\partial x} \right],$$

can then be calculated from Equation (15) by applying Equations (4) and (5). We thus calculate a distribution over the divergence of $\vec{\mathbf{V}}$. It follows that this divergence is proportional to the Laplacian

²Vectors $\hat{\mathbf{i}}, \hat{\mathbf{j}}, \hat{\mathbf{k}}$ denote unit vectors of a 3-D Cartesian coordinate system.

under the same assumption,

$$\nabla^2 \phi(\mathbf{x}, t) \propto \nabla \cdot \vec{V}(\mathbf{x}, t). \quad (16)$$

In particular, our interest lies in the estimation of the Laplacian of the utility function, as it indicates *sources* and *sinks* of the potential function in the environment. In this context, we regard *sinks* as agent *attractors* and *sources* as agent *repellers*. We have therefore introduced a novel framework, which enables us to infer sources and sinks, in an unsupervised manner, to offer an explanation behind multiple observed agent trajectories.

3.3 Metric for locating significant attractors and repellers: Kullback–Leibler divergence

We now require a metric that is able to take advantage of having access to both the posterior mean and variance over sources and sinks in the environment. Therefore, our metric of change from prior field to posterior field is measured via the Kullback–Leibler (KL) divergence [13]. The motivation for selecting the KL divergence comes from its ability measure a distance between two distributions. This provides a natural indication of the informativeness of spatial locations, at given times, and in the context of our application offers a measure of *trajectory-influencing locations*.

Given the model at time t , each point in space has an associated potential field distribution, defined via the GP posterior as a univariate normal distribution. The KL divergence can be readily calculated as the difference between two univariate normal distributions, namely the prior and posterior [8], as below:

$$D_{\text{KL}}(p_{\text{prior}} \parallel p_{\text{posterior}}) = \frac{1}{2} \left(\frac{\sigma_{\text{pr}}^2}{\sigma_{\text{po}}^2} + \frac{(\mu_{\text{po}} - \mu_{\text{pr}})^2}{\sigma_{\text{po}}^2} - 1 + \ln \left(\frac{\sigma_{\text{po}}}{\sigma_{\text{pr}}} \right) \right), \quad (17)$$

where

$$p_{\text{prior}} = \mathcal{N}(\mu_{\text{pr}}, \sigma_{\text{pr}}^2), \quad p_{\text{posterior}} = \mathcal{N}(\mu_{\text{po}}, \sigma_{\text{po}}^2).$$

We refer back to Equation (10) and (12) in order to calculate the following prior Laplacian at location \mathbf{X}_i in space-time:

$$\left(\frac{\partial V_x}{\partial x} + \frac{\partial V_y}{\partial y} \right) \sim \mathcal{N} \left(0, \frac{\partial^2 k_x(\mathbf{X}_i, \mathbf{X}'_i)}{\partial x^2} + \frac{\partial^2 k_y(\mathbf{X}_i, \mathbf{X}'_i)}{\partial y^2} \right),$$

where, $\frac{\partial^2 k_x(\mathbf{X}_i, \mathbf{X}'_i)}{\partial x^2} + \frac{\partial^2 k_y(\mathbf{X}_i, \mathbf{X}'_i)}{\partial y^2} = \frac{h_x^2}{\lambda_x^2} + \frac{h_y^2}{\lambda_y^2}$. The hyperparameters h_x and λ_x are the output and input scale lengths of the x -part of the separable kernel in the GP model, with h_y and λ_y corresponding to the y -part.

As our interest lies in determining attractors and repellers in the field, a further addition to the KL divergence in Equation (17) is to multiply it by the sign of the posterior mean of the Laplacian. This multiplication carries over the prediction of negative sinks and positive sources, whilst measuring the divergence from the zero prior mean. We refer to this extension as the *signed KL divergence*:

$$SD_{\text{KL}}(p_{\text{prior}} \parallel p_{\text{posterior}}) = \text{sign}(\mu_{\text{po}}) D_{\text{KL}}(p_{\text{prior}} \parallel p_{\text{posterior}}). \quad (18)$$

Large values in the SD_{KL} indicate significant influential features in the agent environment, whereas small values around zero indicate that the model is likely to have reverted to its prior, the sensible prior being that there is no significant feature present.

3.4 Computational complexity

In order to use GP-LAPLACE on large data sets, we must overcome the computational complexity associated with inverting the $N \times N$ covariance matrix, which is $O(N^3)$. There is a vast amount of literature that aims to overcome this issue, such as [22], and we use the sparse GP approximation built into the python package `gpfLOW` [16]. This approximation is based on work by Titsias [28], which we choose to implement when the total number of data points, N , exceeds 1000. Using this sparse approximation allows our model to scale to large data sets, as will be shown in Section 5.

4 APPLICATION TO SYNTHETIC DATA

As an illustrative example, we apply GP-LAPLACE to synthetic data, where the true potential function and its derivatives are known. Importantly, this example allows us to evaluate the performance of our approach. We are then able to build on these results for the real-world data set in Section 5, where we cannot possibly have access to the true potential function.

This experiment consists of a multi-agent system of homogeneous agents, whose dynamics we observe. Our goal is to infer, from trajectories alone, the underlying potential value function. We demonstrate that our approach is able to recover the potential field from a small number of agent trajectories by identifying the true sources and sinks. Therefore, we are able to recover trajectory-influencing locations in an unsupervised manner, with no prior knowledge of the environment.

4.1 Agent model

Agents are modelled according to a second order dynamical system, whereby, at each time step t , the acceleration $\ddot{\mathbf{x}}$, velocity $\dot{\mathbf{x}}$ and position \mathbf{x} , with η as the update increment, are given by:

$$\begin{aligned} \ddot{\mathbf{x}}_{t+1} &= \nabla \phi(\mathbf{x}_t, t), \\ \dot{\mathbf{x}}_{t+1} &= \dot{\mathbf{x}}_t + \eta \ddot{\mathbf{x}}_{t+1}, \\ \mathbf{x}_{t+1} &= \mathbf{x}_t + \eta \dot{\mathbf{x}}_{t+1}. \end{aligned}$$

4.2 Agent potential function

We define the causal potential function for our illustrative example as a Gaussian mixture model, where each Gaussian has both a time-varying mean $\mu_i(t)$ and covariance $\Sigma_i(t)$ that change harmonically according to:

$$\phi(\mathbf{x}, t) = \sum_{i=1}^K \alpha_i \mathcal{N}(\mu_i(t), \Sigma_i(t)), \quad (19)$$

where K defines the number of Gaussians and α is the weight of each corresponding Gaussian. For the purposes of our synthetic data, we can think of each Gaussian as a time-varying source or sink.

In order to validate our results, the first and second order derivatives of $\phi(\mathbf{x}, t)$ are calculated using the known derivatives of a multivariate normal distribution $p(\mathbf{x})$ [18]:

$$\begin{aligned} \frac{\partial p(\mathbf{x})}{\partial \mathbf{x}} &= -p(\mathbf{x}) \Sigma^{-1}(\mathbf{x} - \mathbf{m}), \\ \frac{\partial^2 p(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} &= p(\mathbf{x}) \left(\Sigma^{-1}(\mathbf{x} - \mathbf{m})(\mathbf{x} - \mathbf{m})^T \Sigma^{-1} - \Sigma^{-1} \right). \end{aligned}$$

These derivatives are then used in conjunction with the vector calculus operations in Equations (10) and (12) to calculate the true Laplacian of the utility function, which is used as a ground truth in our experiment.

For the purposes of our experiment, we define three different agent potential functions:

(1) *stationary attractors*

$$\phi_1(\mathbf{x}, t) = \sum_{i=1}^K \alpha_i \mathcal{N}(\mathbf{m}_i, c_i \mathbf{I})$$

(2) *varying-strength attractors*

$$\phi_2(\mathbf{x}, t) = \sum_{i=1}^K \alpha_i \mathcal{N}(\mathbf{m}_i, (\sin(t + \beta_i) + c_i) \mathbf{I})$$

(3) *rotating attractors*

$$\phi_3(\mathbf{x}, t) = \sum_{i=1}^K \alpha_i \mathcal{N}(\mathbf{m}_i \odot \begin{bmatrix} \cos t \\ \sin t \end{bmatrix}, c_i \mathbf{I})$$

where α_i , β_i and c_i are constants, \mathbf{m}_i is a two-dimensional constant vector, and \odot denotes an element-wise product. Here, we set $K = 2$ for all three potential functions, along with enforcing $\alpha_i > 0$ to define attractors. Furthermore, we display the derivatives of these agent potential functions in Figure 2 to give an understanding of what they look like in practice.

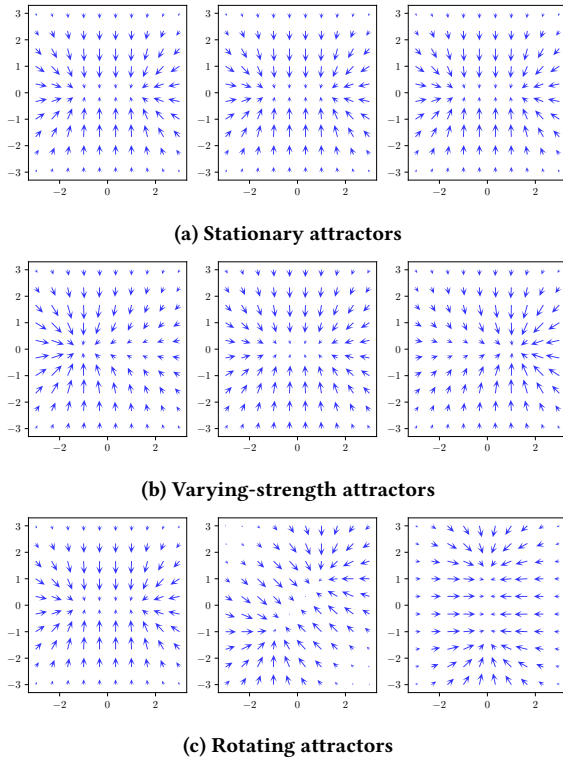


Figure 2: We display the three example vector fields introduced in Section 4.1. Their time-varying properties are displayed by including three frames at different time-steps.

Along with testing the performance of our model, we can also compare the vector fields in Figure 2 to those occurring in nature. As an example, a study by Sommer et al. [27] tracked fruit bats as they moved between their camps and foraging sites. As nocturnal animals, their foraging sites are time-varying attractors, whereby these sites become the most attractive at night. Therefore, it is important to test our model on the time-varying attractors displayed in Figure 2, where we have access to a ground truth, before applying our model to a real-world data set.

4.3 Experimental results

For each of the three potential functions, we initialised M agents with a velocity of zero at random starting locations. The experiment stepped through 200 time-steps and our model is used to infer the vector field and Laplacian from the agent trajectories $\{\zeta_a\}_{a=1}^M$ (see Equation (16)).

As a baseline, we took a simple parametric function in x - and y -space

$$\nabla U = w_0 + w_1x + w_2y + w_3xy + w_4x^2 + w_5y^2 + w_6x^2y + w_7xy^2 + w_8x^3 + w_9y^3 \quad (20)$$

of order three to model the gradient of the potential function, comparable to the approaches of [4, 5, 20].

Table 1 displays the results for the three experimental set-ups, where we have varied the number of M observed agent trajectories from 4 to 16. We vary the number of agents to demonstrate how the models behave and scale as they observe more agents. Each value in the table is the mean squared error between the inferred Laplacian $\nabla^2 \tilde{\phi}$ and the true Laplacian $\nabla^2 \phi$, along with their standard deviation.

The results show that the parametric model was able to recover the general shape of $\nabla^2 \tilde{\phi}$ and often summarised the two attractors as a bowl shape in the three experiments. In comparison, GP-LAPLACE was able to recover the exact shape of the attractors for all three experiments and therefore demonstrated the ability to model the non-stationary characteristics of the potential functions. A further result is that our model improved its accuracy as it observed more trajectories, which is a desirable property of the model, while the baseline did not consistently.

In order to interpret the behaviour of our model, the subset of frames displayed in Figure 3 show how the attractors of the true potential function are tracked across time by both the posterior inferred Laplacian and the signed KL divergence. This figure highlights how GP-LAPLACE is able to completely characterise *rotating attractors* from only observing four trajectories over 200 time-steps. In more detail, the signed KL-divergence gives a measure of significance of the inferred sinks, through taking into account the uncertainty. The black and white markers indicate the location of the rotating attractors given by the ground truth in the first row. For all three rows, regions of attraction are indicated by blue and it can be seen that the model is able to accurately track the true attractors across time.

5 APPLICATION TO REAL-WORLD DATA

In this experiment, we investigate how GP-LAPLACE can be used to determine driving forces in the environment that impact an animal's

Table 1: Results displaying the mean squared error between the true Laplacian and the inferred Laplacian. Each experiment consisted of 200 time-steps with the listed means and standard deviations calculated over 10 different random initialisations of the agents.

	Number of agents			
	4	8	12	16
<i>Stationary attractors</i> - GP-LAPLACE	6.38 ± 2.84	2.20 ± 2.81	0.55 ± 0.27	0.35 ± 0.39
<i>Stationary attractors</i> - parametric	8.27 ± 0.84	8.22 ± 0.48	8.02 ± 0.13	7.99 ± 0.49
<i>Varying-strength attractors</i> - GP-LAPLACE	62.37 ± 99.9	0.88 ± 0.37	0.64 ± 0.11	0.58 ± 0.14
<i>Varying-strength attractors</i> - parametric	23.59 ± 7.23	19.15 ± 3.37	17.67 ± 1.95	19.19 ± 2.59
<i>Rotating attractors</i> - GP-LAPLACE	13.72 ± 20.81	2.05 ± 0.96	0.78 ± 0.16	0.51 ± 0.15
<i>Rotating attractors</i> - parametric	10.75 ± 2.50	8.00 ± 1.19	6.73 ± 0.91	6.69 ± 0.61

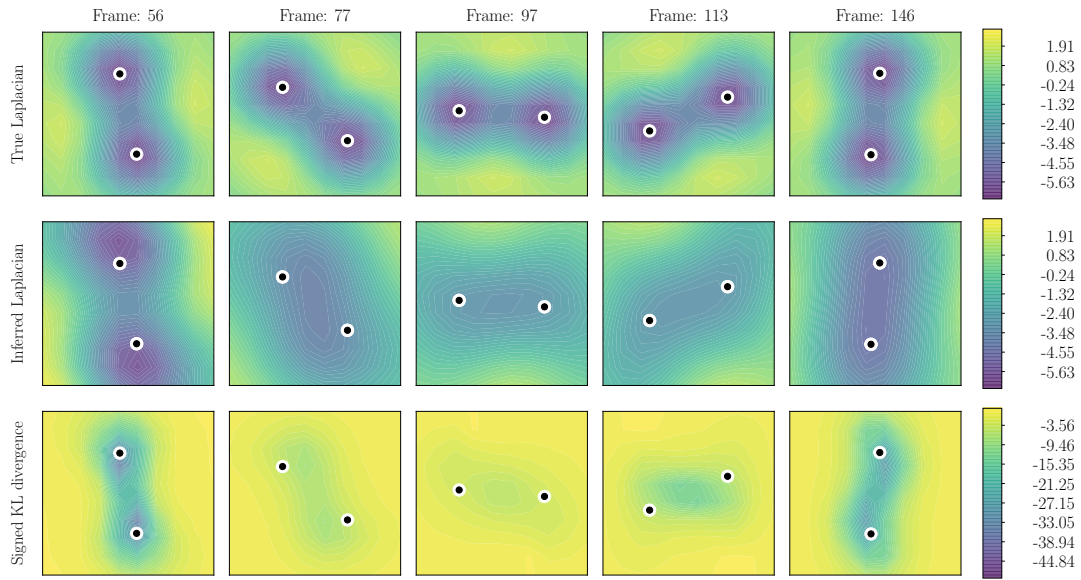


Figure 3: Top row: Laplacian of true utility function. Middle row: inferred Laplacian of utility function. Bottom row: signed KL divergence. The location of the global minimum of the true Laplacian is indicated via the black and white markers. The predicted locations of the sinks, given by both the signed KL divergence and the posterior Laplacian, align well with their true locations, therefore demonstrating the success of our model on identifying non-stationary attractors.

behaviour. We begin with an overview of the data used, present our experimental results, and then conclude with a discussion.

Unlike previous techniques for studying how animals interact with the environment, we don't rely on GPS data to build density maps to construct probability distributions of their likely locations [11, 14]. Furthermore, while more recent approaches have incorporated time into these models [15], current methods do not focus on inferring the driving force behind animal actions and instead simply show where an animal is likely to be found.

5.1 Data

We apply our model to a subset of Scolopi's shearwaters (*Calonectris diomedea*) [26] GPS data to infer the location of influential features in the environment of a MAS of pelagic seabirds. We use

the same data set from Pollonara et al. (2015) [19], made available in a *Movebank data package* [30].

In their experiment, shearwaters were released 400 km from the colony in the northern Mediterranean Sea. Using GPS trackers, the birds' trajectories were mapped and inferences were made about the way in which they navigated. Importantly, the birds were split into separate groups depending on which senses were inhibited. From these, we focus on two sets of birds consisting of the control set and the anosmic set, with the latter unable to use their sense of smell.

Of note is that we do not constrain each trajectory to be equally spaced in time, nor do we require the GPS readings to be time aligned across the trajectories.

5.2 Experimental Results

In Figure 4, we present frames at different time-steps since the four control birds were released (in chronological order from left to right). Following previous work in this area [4, 5, 20], we focus on the divergence of the velocities of the birds. The top row is our inferred Laplacian, pointing out possible sources and sinks. The bottom row is the signed KL divergence, giving a measure of the significance of these features.

Put into context, the birds' colony is slightly to the east of the northern part of Corsica, which is the island shown in the right side of each frame. Therefore, we expect to see our model placing an attractive region in the vicinity of their nest (marker (5) in figure), which is evident from the dark blue region appearing in the top-right of each of the last three frames, which is an attractor we expect to see as the birds start to approach their nest. Furthermore, the first frame shows that the region in which the birds are released is inferred to be a source as the birds fly outwards from this location (marker 1 in figure).

Whereas the control birds in Figure 4 tended to fly directly back to their colony, the anosmic birds flew north until they reached the southern coastline of France (marker (2) in figure). As pointed out in Pollonara et al. (2015) [19], these birds are thought to have relied more on visual cues associated with the coastline, rather than flying straight back to their nest. Figure 5 shows the mean of the signed KL divergence across time for both the anosmic and control set of shearwaters, normalised to be on the same scale.

In Figure 5, a direct comparison can be seen between the two sets of birds. The left-hand side plot displays the mean across time for the control set of shearwaters, confirming that the bird colony is an attractor along with parts of the coastline of Corsica (marker (3) and (4) in figure). In contrast, the right-hand side plot for the anosmic shearwaters displays a different behaviour in both the distribution of the sources and sinks and in the trajectories themselves. As previously mentioned, the anosmic birds head North immediately after being released and use the coastline to recover their bearings (marker (1) and (2) in figure). Therefore our GP model, when taking the mean across time, clearly assigns the region along the coastline of southern France as an attractive region. Attributing the coastline as an attractive region to the anosmic birds agrees with the original suggestion by Pollonara et al. [19] that they navigate via 'following coastlines as a form of search strategy, or by recognition of land features previously encountered'.

5.3 Discussion

A strength of our model is that there is no requirement to incorporate prior information on the location of environmental features. In our application to the shearwater GPS data set, the model had no prior knowledge of the location of the birds' colony or the coastline of Corsica, yet is still able to identify them as significant features. Furthermore, a strength of our technique is the way it is able to display how attractors and repellers vary with time. It would not be desirable to build a model that labels the birds' release point as a source for the entire duration of the birds' flight, although a simple density map would clearly make this error. As shown in the frames of Figure 4, GP-LAPLACE only describes their release point as a

source for the first few hours of the birds' flights and the colony only becomes an attractor towards the end of the time period.

The results of this experiment mean that we are now able to take animal GPS data and infer which environment features are influencing their movements. We are not constrained by a requirement of modelling any details of the environment and therefore point to the generality of this technique as a flexible method that can be applied to many other data sets. The ability to model time-varying drivers in MASs points to further studies such as in Ellwood et al. [9], where we are interested in setting sampling rates for sensors and also making predictions as to which area of the environment will be attractive at different times of the day.

6 RELATED WORK

In this section, we refer to relevant work from both the GP literature and also from work relating to modelling GPS data.

As already introduced as a baseline in Section 4.3, there exists other works which model agents as acting according to a potential field [4, 5, 20]. However, we show that GP-LAPLACE outperforms these methods due to the added flexibility of being able to model a time-varying potential function. There also exists alternative techniques for studying how animals interact with the environment, which often rely on using GPS data to build density maps to construct probability distributions of their likely locations [11, 14]. Although more recent approaches have incorporated time into these models [15], they do not focus on inferring the dynamics and the driving forces behind animal actions, which is one of our motivations for introducing GP-LAPLACE.

When relating our use of GPs to previous work with vector fields, Wahlström et al. [29] uses GPs to directly model magnetic fields. This work applies divergence-free and curl-free kernels to enforce linear constraints on their models, where the idea of constraining kernels to be divergence-free and curl-free is extended in [12]. Our work extends on previous work in this area with the introduction of a model that takes advantage of the derivative properties of GPs to give distributions over the operations of vector calculus from lower order observations. We have shown that inferring distributions over these operations results in an interpretable methodology that has not been previously employed in the literature.

7 CONCLUSION AND FUTURE WORK

In this paper, we present GP-LAPLACE, a technique for locating attractors from trajectories in time-varying fields. Through applying Gaussian processes combined with vector calculus, we provide a model that is able to infer sources and sinks in the presence of multiple agents from their trajectories alone. Additionally, our probabilistic technique enables us to utilise the KL divergence to give a measure of significance of environmental features.

To demonstrate the generality of our method, we applied it to two data sets. First, on our synthetic data set, we showed that GP-LAPLACE is able to reconstruct non-stationary attractors effectively, with superior performance over the baseline parametric model. Next, in an unsupervised fashion, we applied our model to a real-world example where we were able to infer features of the environment, such as the release point and bird colony, without prior knowledge of the birds' surroundings.

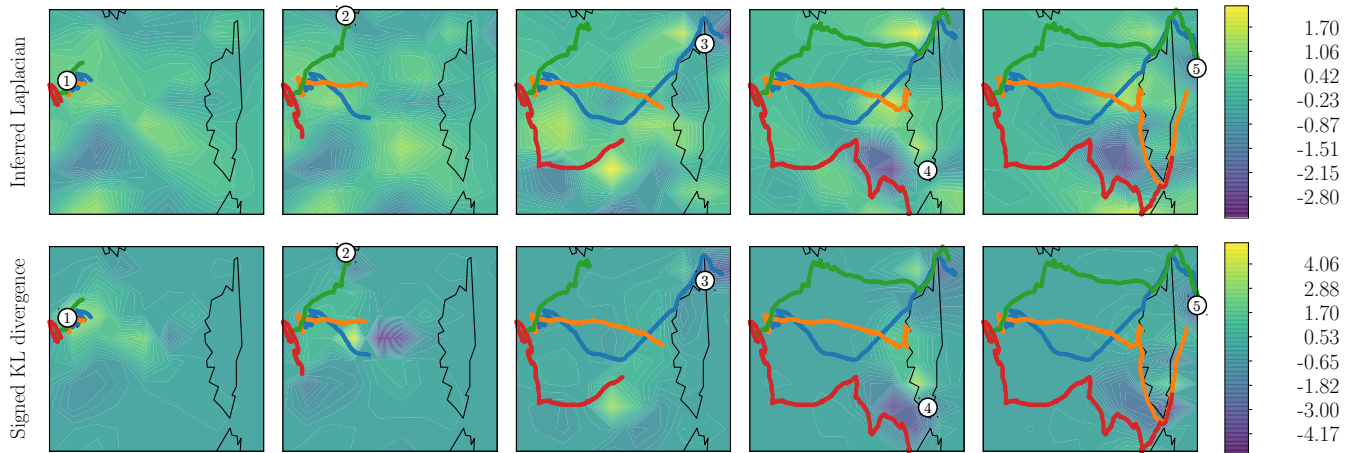


Figure 4: Top row: Inferred Laplacian of utility function. Bottom row: Inferred signed KL divergence. Both rows are superimposed on a map of the northern Mediterranean Sea. Each frame represents a snapshot of the sources and sinks relating to the velocity flow of the four shearwaters for 48 hours starting from 11:00 pm 21st June 2012. Points of note: (1) Release point, (2) South of France, (3) North of Corsica, (4) South of Corsica, (5) Nest. Best viewed in colour.

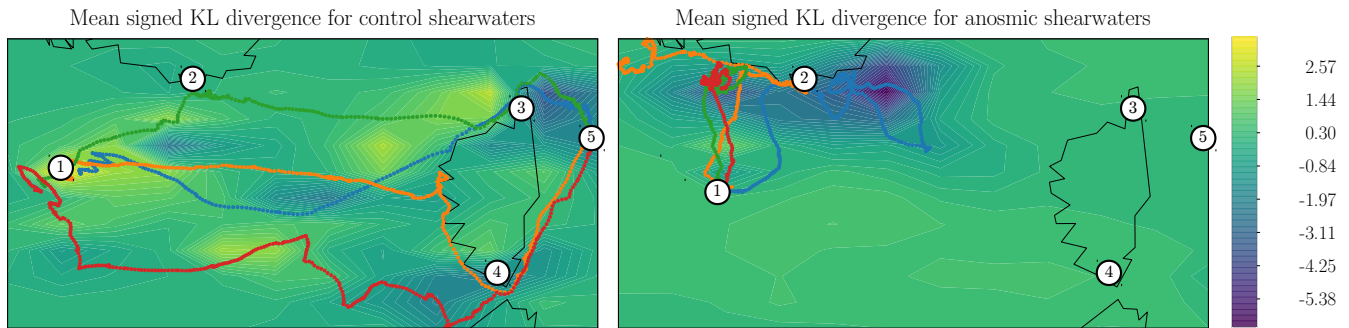


Figure 5: Left: Mean signed KL divergence for control shearwaters. Right: Mean signed KL divergence for anosmic shearwaters. Both plots contain trajectories of four separate shearwaters released at times similar times. Blue areas show attractive regions and the yellow denotes sources. The black outline corresponds to landmass, where the top is part of the southern French coastline and the island on the right is Corsica. Over a similar time period, the difference in behaviour can be seen from both the routes taken by the birds and the average placement of the sources and sinks. Points of note: (1) Release point, (2) South of France, (3) North of Corsica, (4) South of Corsica, (5) Nest. Best viewed in colour.

In future work, we will extend GP-LAPLACE to incorporate explicit agent interactions. As an example, work by Preisler et al. [20] included an interaction term in their potential function to measure the strength of disturbances as having an effect on elk movement. We will also look into applying our model to additional real-world data sets, in situations where discovering sources and sinks in the presence of multiple agents will offer further insight into the motivating factors behind agent behaviour.

ACKNOWLEDGMENTS

Adam D. Cobb is sponsored by the AIMS CDT (<http://aims.robots.ox.ac.uk>) and the EPSRC (<https://www.epsrc.ac.uk>). Richard Everett is sponsored by the UK EPSRC CDT in Cyber Security. We also thank Ivan Kiskin for extensive comments and feedback.

REFERENCES

- [1] Pieter Abbeel and Andrew Y Ng. 2004. Apprenticeship Learning via Inverse Reinforcement Learning. In *Proceedings of the twenty-first international conference on Machine learning*. ACM, 1.
- [2] Stefano V Albrecht and Peter Stone. 2017. Autonomous Agents Modelling Other Agents: A Comprehensive Survey and Open Problems. *arXiv preprint arXiv:1709.08071* (2017).
- [3] Mauricio A Alvarez, Lorenzo Rosasco, Neil D Lawrence, et al. 2012. Kernels for Vector-Valued Functions: a Review. *Foundations and Trends® in Machine Learning* 4, 3 (2012), 195–266.
- [4] DR Brillinger, HK Preisler, MJ Wisdom, et al. 2011. Modelling particles moving in a potential field with pairwise interactions and an application. *Brazilian Journal of Probability and Statistics* 25, 3 (2011), 421–436.
- [5] David R Brillinger, Brent S Stewart, Charles L Littnan, et al. 2008. Three months journeying of a Hawaiian monk seal. In *Probability and statistics: Essays in honor of David A. Freedman*. Institute of Mathematical Statistics, 246–264.
- [6] PR Brook, A Karastergiou, S Johnston, M Kerr, RM Shannon, and SJ Roberts. 2016. Emission-rotation correlation in pulsars: new discoveries with optimal

- techniques. *Monthly Notices of the Royal Astronomical Society* 456, 2 (2016), 1374–1393.
- [7] Wei Chu and Zoubin Ghahramani. 2005. Preference Learning with Gaussian Processes. In *Proceedings of the 22nd international conference on Machine learning*. ACM, 137–144.
- [8] John Duchi. 2007. Derivations for Linear Algebra and Optimization. *Berkeley, California* (2007).
- [9] Stephen A Ellwood, Chris Newman, Robert A Montgomery, Vincenzo Nicosia, Christina D Buesching, Andrew Markham, Cecilia Mascolo, Niki Trigoni, Bence Pasztor, Vladimir Dyo, et al. 2017. An active-radio-frequency-identification system capable of identifying co-locations and social-structure: Validation with a wild free-ranging animal. *Methods in Ecology and Evolution* 8, 12 (2017), 1822–1831.
- [10] Tracy Holsclaw, Bruno Sansó, Herbert KH Lee, Katrin Heitmann, Salman Habib, David Higdon, and Ujjaini Alam. 2013. Gaussian Process Modeling of Derivative Curves. *Technometrics* 55, 1 (2013), 57–67.
- [11] Jon S Horne, Edward O Garton, Stephen M Krone, and Jesse S Lewis. 2007. Analyzing animal movements using Brownian bridges. *Ecology* 88, 9 (2007), 2354–2363.
- [12] Carl Jidling, Niklas Wahlström, Adrian Wills, and Thomas B Schön. 2017. Linearly constrained Gaussian processes. *arXiv preprint arXiv:1703.00787* (2017).
- [13] Solomon Kullback and Richard A Leibler. 1951. On Information and Sufficiency. *The annals of mathematical statistics* 22, 1 (1951), 79–86.
- [14] Peter N Laver and Marcella J Kelly. 2008. A Critical Review of Home Range Studies. *Journal of Wildlife Management* 72, 1 (2008), 290–298.
- [15] Andrew J Lyons, Wendy C Turner, and Wayne M Getz. 2013. Home range plus: a space-time characterization of movement over real landscapes. *Movement Ecology* 1, 1 (2013), 2.
- [16] Alexander G. de G. Matthews, Mark van der Wilk, Tom Nickson, Keisuke Fujii, Alexis Boukouvalas, Pablo León-Villagrà, Zoubin Ghahramani, and James Hensman. 2017. GPflow: A Gaussian process library using TensorFlow. *Journal of Machine Learning Research* 18, 40 (April 2017), 1–6. <http://jmlr.org/papers/v18/16-537.html>
- [17] Andrew McHutchon. 2013. Differentiating Gaussian Processes. <http://mlg.eng.cam.ac.uk/mchutchon/DifferentiatingGPs.pdf>.
- [18] Kaare Brandt Petersen, Michael Syskind Pedersen, et al. 2008. The Matrix Cookbook. *Technical University of Denmark* 7 (2008), 15.
- [19] Enrica Pollonara, Paolo Luschi, Tim Guilford, Martin Wikelski, Francesco Bonadonna, and Anna Gagliardo. 2015. Olfaction and topography, but not magnetic cues, control navigation in a pelagic seabird: displacements with shearwaters in the Mediterranean Sea. *Scientific reports* 5 (2015).
- [20] Haiganoush K Preisler, Alan A Ager, and Michael J Wisdom. 2013. Analyzing animal movement patterns using potential functions. *Ecosphere* 4, 3 (2013), 1–13.
- [21] Martin L Puterman. 2014. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.
- [22] Joaquin Quiñero-Candela and Carl Edward Rasmussen. 2005. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research* 6, Dec (2005), 1939–1959.
- [23] Carl Edward Rasmussen. 2006. *Gaussian Processes for Machine Learning*. (2006).
- [24] Stephen Roberts, M Osborne, M Ebdon, Steven Reece, N Gibson, and S Aigrain. 2013. Gaussian processes for time-series modelling. *Phil. Trans. R. Soc. A* 371, 1984 (2013), 20110550.
- [25] Stuart Russell. 1998. Learning agents for uncertain environments. In *Proceedings of the eleventh annual conference on Computational learning theory*. ACM, 101–103.
- [26] George Sangster, J Martin Collinson, Pierre-André Crochet, Alan G Knox, David T Parkin, and Stephen C Votier. 2012. Taxonomic recommendations for British birds: eighth report. *Ibis* 154, 4 (2012), 874–883.
- [27] Philipp Sommer, Jiajun Liu, Kun Zhao, Branislav Kusy, Raja Jurdak, Adam McKeown, and David Westcott. 2016. Information Bang for the Energy Buck: Towards Energy-and Mobility-Aware Tracking.. In *EWSN*. 193–204.
- [28] Michalis K Titsias. 2009. Variational Learning of Inducing Variables in Sparse Gaussian Processes. In *AISTATS*, Vol. 5. 567–574.
- [29] Niklas Wahlström, Manon Kok, Thomas B Schön, and Fredrik Gustafsson. 2013. Modeling Magnetic Fields Using Gaussian Processes. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 3522–3526.
- [30] M Wikelski and R Kays. 2014. Movebank: archive, analysis and sharing of animal movement data. www.movebank.org. (2014).
- [31] Alan M Wilson, JC Lowe, K Roskilly, Penny E Hudson, KA Golabek, and JW McNutt. 2013. Locomotion dynamics of hunting in wild cheetahs. *Nature* 498, 7453 (2013), 185.