Journal of Mathematical Psychology 85 (2018) 1–16

Contents lists available at ScienceDirect

Journal of Mathematical Psychology

journal homepage: www.elsevier.com/locate/jmp

A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions

Eric Schulz [a,*], Maarten Speekenbrink [b], Andreas Krause [c]

[a] Department of Psychology, Harvard University, United States
[b] Department of Experimental Psychology, University College London, United Kingdom
[c] Department of Computer Science, Swiss Federal Institute of Technology, Zürich, Switzerland

ARTICLE INFO

Article history:
Received 11 October 2017

Keywords:
Gaussian process regression
Active learning
Exploration–exploitation
Bandit problems

ABSTRACT

This tutorial introduces the reader to Gaussian process regression as an expressive tool to model, actively explore and exploit unknown functions. Gaussian process regression is a powerful, non-parametric Bayesian approach towards regression problems that can be utilized in exploration and exploitation scenarios. This tutorial aims to provide an accessible introduction to these techniques. We will introduce Gaussian processes which generate distributions over functions used for Bayesian non-parametric regression, and demonstrate their use in applications and didactic examples including simple regression problems, a demonstration of kernel-encoded prior assumptions and compositions, a pure exploration scenario within an optimal design framework, and a bandit-like exploration–exploitation scenario where the goal is to recommend movies. Beyond that, we describe a situation modelling risk-averse exploration in which an additional constraint (not to sample below a certain threshold) needs to be accounted for. Lastly, we summarize recent psychological experiments utilizing Gaussian processes. Software and literature pointers are also provided.

1. Introduction

Whether we try to find a function that accurately describes participants' behaviour (Cavagnaro, Aranovich, McClure, Pitt, & Myung, 2014), estimate parameters of psychological models (Wetzels, Vandekerckhove, Tuerlinckx, & Wagenmakers, 2010), try to sequentially optimize the stimuli used in an experiment (Myung & Pitt, 2009), or model how participants learn to interact with their environment (Meder & Nelson, 2012), many problems require us to assess unknown functions that map inputs to outputs. Often, the shape of the underlying function is unknown, the function might be hard to evaluate analytically, or other requirements such as design costs might complicate the process of information acquisition. In these situations, Gaussian process regression can serve as a useful tool for performing inference both passively (for example, describing a given data set as best as possible, allowing one to also predict future data) as well as actively (for example, learning while choosing input points to produce the highest possible outputs, cf Williams & Rasmussen, 2006). Gaussian process regression is a non-parametric Bayesian approach (Gershman & Blei, 2012) towards regression problems. It can capture a wide variety of relations between inputs and outputs by utilizing a theoretically infinite number of parameters and letting the data determine the level of complexity through the means of Bayesian inference (Williams, 1998).

This tutorial will introduce Gaussian process regression as an approach towards describing, and actively learning and optimizing unknown functions. It is intended to be accessible to a general readership and focuses on practical examples and high-level explanations. It consists of six main parts: The first part will introduce the mathematical underpinnings of Gaussian process regression. The second part will show how different kernels can encode prior assumptions about the underlying function. Next, we will show how Gaussian processes can be used in problems of optimal experimental design, when the goal is pure exploration, i.e., to learn a function as well as possible. The fourth part will describe how Gaussian process-based Bayesian optimization (here defined as an *exploration–exploitation problem*) works. In the fifth part, we will talk about ways of utilizing Gaussian process exploration–exploitation methods in situations with additional requirements and show one example of "safe exploration", where the goal is to avoid outputs below a certain threshold. We will conclude by summarizing current research that treats Gaussian process regression as a psychological model to assess human function learning.

As a tutorial like this can never be fully comprehensive, we have tried to provide detailed references and software pointers whenever possible.

* Corresponding author.
E-mail addresses: ericschulz@fas.harvard.edu (E. Schulz), m.speekenbrink@ucl.ac.uk (M. Speekenbrink), krausea@ethz.ch (A. Krause).

https://doi.org/10.1016/j.jmp.2018.03.001
0022-2496/

Table 1
Overview of different Gaussian process methods (including their example applications) introduced in this tutorial.

Method	Purpose	Approach	Example
Modelling	Simple regression	passive	Mouse trajectories
Compositional modelling	Find patterns within data	passive	Response time patterns
Exploration	Learn function as quickly as possible	active	Learn simulated functions
Exploration–exploitation	Optimize function	active	Movie recommendation
Safe exploration	Optimize function while staying above a threshold	active	Cautious stimulus optimization

2. Gaussian processes — distributions over functions

2.1. Motivation

Let f denote an (unknown) function which maps inputs x to outputs y: $f : X \rightarrow Y$. Throughout the following examples, we will use Gaussian process regression to accomplish either one of three different goals:

By *modelling* a function f we mean mathematically representing the relation between inputs and outputs. An accurate model of f allows us to predict the output for many possible input values. In practice, this means collecting observations of both inputs and outputs and on the basis of this generating accurate predictions for newly observed points. As an example of this, we will use Gaussian process regression to model mouse trajectories in a categorization experiment. Additionally, we will use compositional Gaussian process regression to decompose temporal dependencies in participants' reaction times into interesting patterns.

By *exploring* a function we mean to actively choose the input points for which to observe the outputs in order to accurately model the function. In pure exploration problems, the only objective is to explore the underlying function well in order to learn about it as quickly and accurately as possible. This set-up is closely related to optimal experimental design scenarios as it equates to adaptively selecting the input points based on what is already known about the function and where knowledge can be improved. In a simple simulation experiment, we will show how exploration based on Gaussian process regression can recover underlying response functions faster than other commonly used techniques.

In *exploration–exploitation* problems, the outcomes of chosen inputs are accrued over time. The objective is to find inputs that produce the highest outputs in order to maximize the total reward accrued within a particular period of time. Exploration solely serves the purpose of doing so most effectively. This set-up is closely related to optimization problems as the goal is to find the maximum of the function as efficiently as possible. It is called *exploration–exploitation* as scenarios where the output of the underlying function has to be optimized require us to both sample uncertain areas in order to gain more knowledge about the function (exploration) as well as sampling input points that are likely to generate high outputs given the current knowledge of the function (exploitation). As an example, we will show how Gaussian process-based exploration–exploitation quickly finds highly rated items in a movie recommendation application. Moreover, we will show how this method can be adapted to additional requirements such as avoiding outputs below a given threshold.

Both *exploration* and *exploration–exploitation* tasks require choosing useful inputs. Doing so requires two ingredients:

1. A model used to learn about the function f.
2. A method to select inputs based on the current knowledge of f.

As a valid model of the underlying function f is crucial for all three goals of modelling, exploration, and exploitation, we will first focus on Gaussian processes as a powerful and expressive method to model unknown functions. We will focus on applying this tool to exploration–exploitation scenarios afterwards. Table 1 provides an overview of the different Gaussian process methods (and their example applications) introduced in this tutorial.

Table 2
Observations for the regression example. Inputs x_t and corresponding outputs y_t observed at 6 different times $t = 1, \ldots, 6$.

t	x_t	y_t
1	0.9	0.1
2	3.8	1.2
3	5.2	2.1
4	6.1	1.1
5	7.5	1.5
6	9.6	1.2

2.2. Modelling functions: the weight space view

Let us start by considering a standard approach to model functions: linear regression (here approached from a Bayesian viewpoint). Imagine we have collected the observations shown in Table 2 and that we want to predict the value of y for a new input point $x_\star = 3$. In linear regression (see Fig. 1), we assume that the outputs are a linear function of the inputs with additional noise:

$$\begin{aligned} y_t &= f(x_t) + \epsilon_i \\ &= \beta_0 + \beta_1 x_t + \epsilon_t, \end{aligned}$$

where the noise term ϵ_t follows a normal distribution

$$\epsilon_t \sim \mathcal{N}(0, \sigma_\epsilon^2)$$

with mean 0 and variance σ_ϵ^2. As this will be useful later, we can also write this in matrix algebra as

$$y_t = \mathbf{x}_t^\top \mathbf{w} + \epsilon_i$$

defining the vectors

$$\mathbf{x}_t = \begin{bmatrix} 1 \\ x_t \end{bmatrix}, \qquad \mathbf{w} = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}.$$

To predict the output for $x_\star$, we need to estimate the weights from the previous observations

$$\mathbf{X}_t = \begin{bmatrix} 1 & 0.9 \\ 1 & 3.8 \\ \vdots & \vdots \\ 1 & 9.6 \end{bmatrix}, \qquad \mathbf{y}_t = \begin{bmatrix} 0.1 \\ 1.2 \\ \vdots \\ 1.2 \end{bmatrix}.$$

Adopting a Bayesian framework, we do so through the posterior distribution over the weights. If we use a Gaussian prior over the weights $p(\mathbf{w}) = \mathcal{N}(0, \Sigma)$ and the Gaussian likelihood $p(\mathbf{y}_t|\mathbf{X}_t, \mathbf{w}) = \mathcal{N}(\mathbf{X}_t^\top \mathbf{w}, \sigma_\epsilon^2 \mathbf{I})$, then this posterior distribution is

$$\begin{aligned} p(\mathbf{w}|\mathbf{y}_t, \mathbf{X}_t) &\propto p(\mathbf{y}_t|\mathbf{X}_t, \mathbf{w}) p(\mathbf{w}) \\ &= \mathcal{N}\left(\frac{1}{\sigma_\epsilon^2} \mathbf{A}_t^{-1} \mathbf{X}_t \mathbf{y}_t, \mathbf{A}_t^{-1}\right) \end{aligned} \tag{1}$$

where $\mathbf{A}_t = \Sigma^{-1} + \sigma_\epsilon^{-2} \mathbf{X}_t \mathbf{X}_t^\top$ (see also Williams, 1998).

As inference is performed over the weights (i.e., we try to find the best estimate for the β-weights given the data), this is also sometimes referred to as "the weight space view of regression". To predict the output $y_\star$ at a new test point $\mathbf{x}_\star$, we can average out the error term and focus on the expected value which is provided by the function f, predicting $f_\star = y_\star - \epsilon_\star = f(\mathbf{x}_\star)$. In the predictive