

On Gaussian Processes for Regression

Jeffrey Alido

Department of Electrical and Computer Engineering
Boston University
jalido@bu.edu

Shashank Manjunath

Department of Electrical and Computer Engineering
Boston University
manjuns@bu.edu
April 27, 2021

Abstract

- Gaussian processes emerged in machine learning as a powerful tool for regression and classification that provides interpretability through kernel choice and uncertainty quantification.
- By leveraging properties of multivariate normal distributions and Bayes's rule, we may infer a probability distribution over possible functions when fitting a dataset
- This Bayesian framework allows flexibility through choosing a covariance function as a prior belief about the dataset, which can provide further insight into the trends of the training data.

Our specific contributions are:

- We implement a multi-dimensional Gaussian process regressor in MATLAB, as well as several kernels for use with this regressor
- We evaluate the GP regression performance on the Boston Housing dataset, achieving root mean square errors comparable to those in the top 10 of released results on the Kaggle website
- We implement hyperparameter optimization through maximum likelihood estimation, to remove the need for manual tuning of the hyperparameters

Gaussian Random Variables

A random variable is a function that maps from an event space to a measurable space. The event space represents a set of all possible outcomes that the random variable may take, and the measurable space is a probability measure between 0 and 1 (inclusive). We say that a random variable X is normally distributed if the event space has a Gaussian probability distribution, fully characterized by two parameters: a mean μ and variance σ^2 :

$$X \sim \mathcal{N}(\mu, \sigma^2)$$

Gaussian Random Variables

For a one-dimensional Gaussian random variable, we refer to its distribution as a univariate Gaussian distribution. A set of Gaussian random variables may be characterized jointly as a multivariate Gaussian distribution, with joint probability distribution fully characterized by a mean vector and a covariance matrix:

$$\mathbf{X} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{bmatrix} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$$

where $\boldsymbol{\mu}$ is the mean vector, and Σ is the covariance matrix whose entries describe the covariance between each pair of random variables.

Gaussian Processes

A random process is essentially a collection of random variables jointly characterized as a set or vector of random variables with a multivariate joint probability distribution. A Gaussian process $f(\mathbf{x})$ is defined as a random process where each set of random variable in the random process has a multivariate Gaussian distribution. $f(\mathbf{x})$ is fully characterized by a mean function $m(\mathbf{x})$ and covariance function, $K(\mathbf{x}, \mathbf{x}')$:

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), K(\mathbf{x}, \mathbf{x}'))$$

Typically, the mean function is zero. The kernel for the covariance is chosen based on some prior belief about the dataset; more on kernels is discussed in Slide 8.

Gaussian Processes for Regression

Suppose we observe training data \mathbf{t} and choose kernel κ . Then the mean and covariance functions are given by

$$\begin{aligned}m(\mathbf{x}) &= \mathbf{C}_{\mathbf{x}\mathbf{t}}^\top \mathbf{C}_{\mathbf{t}}^{-1} \mathbf{t} \\K(\mathbf{x}, \mathbf{x}') &= C_{\mathbf{x}\mathbf{x}'} - \mathbf{C}_{\mathbf{x}\mathbf{t}}^\top \mathbf{C}_{\mathbf{t}}^{-1} \mathbf{C}_{\mathbf{x}\mathbf{t}}\end{aligned}$$

where $C_{\mathbf{x}\mathbf{t}} = \kappa(\mathbf{x}, \mathbf{t})$, $C_{\mathbf{t}} = \kappa(\mathbf{t}, \mathbf{t})$, and $C_{\mathbf{x}\mathbf{x}'} = \kappa(\mathbf{x}, \mathbf{x}')$. Those interested in the derivation of the results are encouraged to consult section 6.4.2 of [Bis06].

Covariance Functions or kernels, denoted $\kappa(\mathbf{x}, \mathbf{x}')$, form the core of a Gaussian process. Kernels allow projection of input data into an alternate feature space, allowing easier separability of data in this new feature space. Gaussian processes leverage kernels to featurize input data. In particular, if we have a function $\Phi(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}^n$, we can write the kernel defined by this function as:

$$\kappa(\mathbf{x}, \mathbf{x}') = \langle \Phi(\mathbf{x}), \Phi(\mathbf{x}') \rangle = \Phi(\mathbf{x})^\top \Phi(\mathbf{x}')$$

In order to illustrate how kernels fit into Gaussian Processes, we will
finish derivation of kernel trick in GP

Radial Basis Function Kernel

The Radial Basis Function (RBF) Kernel, also known as the Squared Exponential Kernel, is given by:

$$\kappa_{RBF}(\mathbf{x}, \mathbf{x}') = \sigma^2 \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|_2^2}{2\ell^2}\right)$$

This kernel is parametrized by two parameters, the lengthscale ℓ and the variance σ^2 . The lengthscale determines the width of the kernel, and the variance scales the kernel[Duv14]. We provide an images of the RBF Kernel with various lengthscales and variances in Figure ??

Rational Quadratic Kernel

The Rational Quadratic Kernel is another standard kernel is similar to the RBF kernel. It can be constructed from summing RBF kernels with varying lengthscales. The kernel is given by:

$$\kappa_{RQ}(\mathbf{x}, \mathbf{x}') = \sigma^2 \left(1 + \frac{\|\mathbf{x} - \mathbf{x}'\|_2^2}{2\alpha\ell^2} \right)^{-\alpha}$$

This kernel is parametrized by three parameters, the lengthscale ℓ , the variance σ^2 , and the lengthscale weighting parameter α [Duv14]. We provide images of the Rational Quadratic Kernel with various α values in Figure ??

Periodic Kernel

The periodic kernel allows us to model periodic functions. The kernel is given by:

$$\kappa_P(\mathbf{x}, \mathbf{x}') = \sigma^2 \exp\left(-\frac{2 \sin^2(\pi \|\mathbf{x} - \mathbf{x}'\|)}{p\ell^2}\right)$$

This kernel is parametrized by two parameters, p which describes the period of the function, and ℓ which is the lengthscale[Duv14]. We provide images of the Periodic Kernel with various p values in Figure ??

Bayesian Hyperparameter Optimization

@manjuns need to finish this

A Simple Demonstration

fit some GPs on sine waves or something

The Boston Housing Dataset

The Boston Housing Dataset, originally published in 1978 contains 506 data points, each containing 13 features and 1 label for regression[HR78]. The dataset provides the median value of houses in Boston suburbs. This dataset is particularly suitable for Gaussian processes, as the dataset is quite small. The label is the median value of owner-occupied homes in \$1000s, and all other features are used for model fitting. The features included in the dataset are given in Table 1.

Table 1: Table of Boston Housing Dataset feature names and features

Feature Name	Feature Description
CRIM	Per capita crime rate by town
ZN	Proportion of residential land zoned for lots over 25,000 sq.ft.
INDUS	Proportion of non-retail business acres per town.
CHAS	Charles River dummy variable (1 if tract bounds river; 0 otherwise)
NOX	Nitric oxides concentration (parts per 10 million)
RM	Average number of rooms per dwelling
AGE	Proportion of owner-occupied units built prior to 1940
DIS	Weighted distances to five Boston employment centres
RAD	Index of accessibility to radial highways
TAX	Full-value property-tax rate per \$10,000
PTRATIO	Pupil-teacher ratio by town
B	$1000(B_k - 0.63)^2$ where B_k is the proportion of Black people by town
LSTAT	% lower status of the population
MEDV	Median value of owner-occupied homes in \$1000's




Normalization

For the application of Gaussian Processes, we use the regression task, i.e. fitting to the MEDV feature. Prior to fitting on the data, we normalize the data per feature. Specifically, for each feature in the dataset, we perform the following operation:

$$X_{\text{feat}} = \frac{X_{\text{feat}} - \mu(X_{\text{feat}})}{\sigma(X_{\text{feat}})}$$

where $\mu(X)$ is the mean value of that feature in the training set, and $\sigma(X)$ is the standard deviation of that feature in the training set. We additionally normalize the MEDV feature, and convert it back to non-normalized units before calculating our RMSE.

References

-  Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.
-  David Duvenaud, *Automatic Model Construction with Gaussian Processes*, PhD Thesis, Computational and Biological Learning Laboratory, University of Cambridge, 2014.
-  David Harrison and Daniel L. Rubinfeld, *Hedonic housing prices and the demand for clean air*, *Journal of Environmental Economics and Management* **5** (1978), no. 1, 81–102.