Gaussian Process Regression

Shreya Pathak
Prof. Suyash Awate

January 2, 2020

1 Introduction

A random process is a possibly infinite collection of indexed random variable. A Gaussian process is one such class of processes where any finite number of the random variables has a multivariate Gaussian distribution. These have useful applications in the field of Machine Learning in both the major categories of supervised learning methods- regression and classification. This report focuses on their usage for performing regression.

2 Linear Regression

In the usual simple model, the output is expressed as a weighted combination of some functions of the input terms called as features.. Suppose we have m training examples consisting of n features each, these can be expressed in a $m \times n$ dimensional matrix called the design matrix. Suppose we start with the basic case where each output is a function of just the inputs (of dimension D accompanied by a Gaussian noise. This can be expressed as:

$$y = x^T w + \varepsilon; y = f(x) + \varepsilon \tag{1}$$

where w is the vector of weights and is the gaussian noise, i.e. $\varepsilon \sim N(0, \sigma_n^2)$. In this case the likelihood can be written as (using that comes from a Gaussian distribution)

$$p(y|w,X) = \Pi \frac{1}{\sqrt{2\pi}\sigma_n} exp(-(y - x^T w)^2/2\sigma_n^2) = \frac{1}{(2\pi)^{m/2}\sigma_n^m} exp(-|y - X^T w|^2/2\sigma_n^2) \tag{2}$$

As we can see that the likelihood can be obtained from $N(X^T w, \sigma_n^2)$ To perform Bayesian analysis we take $N(0, \Sigma_p)$ as our prior for w . Therefore the posterior can be written as

$$p(w|X,y) \propto p(y|w,X) * p(w) = exp(-(y - X^T w)^T (y - X^T w)/2\sigma_n^2) * exp(-w^T \Sigma_p^{-1} w/2) \tag{3}$$

This can be shown by calculation to be corresponding to another multivariate Gaussian, $N(\bar{w}, A^{-1})$, where $\overline{w} = \sigma_n^{-2}(\sigma_n^{-2} XX^T + \Sigma_p^{-1})^{-1} Xy$ and $A = \frac{1}{\sigma_n^2} XX^T + \Sigma_p^{-1}$ Now we can find the mean and MAP estimates of w which both correspond to w To find the posterior predictive distribution i.e., $p(f(x_*)|y,X,x_*)$ where $x_*$is the test data point

$$p(f_*|X,y,x_*) = \int p(f_*|w,x_*) * p(w|X,y) dw \tag{4}$$

Which is corresponding to another multivariate Gaussian $N((1/\sigma_n^2) x_*^T A^{-1} Xy, x_*^T A^{-1} x_*)$ Instead of a linear relation between the output and input, we can also have a linear relation in the feature space, where features are obtained by a mapping from the inputs to functions of them which may increase or decrease the dimensionality of the inputs. Let $\phi(x)$ be the function which transforms the D dimensional input to the N dimensional feature space. Therefore, we can now write

$$y = \phi(x)^T w + \varepsilon \tag{5}$$

The rest of the equations are similar obtained by replacing x by $\phi(x)$ and X by $\Phi(X)$, i.e.,

$$f_*|x_*, X, y \sim N(\frac{1}{\sigma_n^2}\phi(x_*)^T\Phi(X)y, \phi(x_*)^T A^{-1}\phi(x_*)) \tag{6}$$

This can be rearranged to write as

$$f_*|x_* X, y \sim N(\phi(x_*)\Sigma_p\Phi(X)(K+\sigma_n^2 I)^{-1}y, \\ \phi(x_*)^T\Sigma_p\phi(x_*) - \phi(x_*)^T\Sigma_p\Phi(X)(K+\sigma_n^2 I)^{-1}\Phi(X)^T\Sigma_p\phi(x_*)) \tag{7}$$

Where $K = \Phi(X)^T\Sigma_p\Phi(X)$. This formulation will be useful later.

3 Gaussian Process

The feature space view is useful and more flexible than simple linear regression, however there are infinitely many mappings to choose from which makes the task difficult. GPR provides a way around this by providing a distribution over all the possible functions compatible with the data. Therefore, it is non parametric. In other respects it is similar to the Bayesian approach. A distribution over functions can be parameterised by visualising it as a mapping from a finite (say of size m) domain to R. So each element in the distribution will be an m-dimensional vector, which can then belong to say a multivariate Gaussian distribution (as in the case of Gaussian process). For infinite domain we define the mean $m(x)$ and covariance functions $k(x,x')$ for a real process $f(x)$, i.e,

$$\begin{aligned} m(x) &= E(f(x)) \\ k(x,x') &= E((f(x)-m(x))(f(x')-m(x')) \\ f(x) &\sim GP(m(x), k(x,x')) \end{aligned}$$

Such that every finite collection of random variables has a gaussian distribution. For a process to be a GP we can have any real valued function as $m(x)$ but only such functions for $k(x,x')$ which are such that for any finite collection of inputs $x_1, x_2, x_3, ...x_m$ is such that if A_{ij} is an element of the matrix K, $A_{ij} = k(x_i, x_j)$. This acts as a distribution over functions as follows. Suppose X_* corresponds to vector of inputs.

$$f_* \sim N(0, K(X_*, X_*)$$

We can see the Bayesian linear regression model as an example of Gaussian process, where $m(x) = 0$ and $k(x,x') = \phi(x)^T\Sigma_p\phi(x')$. Suppose we choose some function (say the squared exponential) as the covariance function. It can be shown by the Mercer's theorem that every function can be expressed as a weighted sum of infinite basis functions and thus corresponds to a Bayesian linear regression model.

4 Function Space View

Starting with the same expression as in weight space view, we have

$$y = f(x) + \varepsilon$$

where $\varepsilon \sim N(0, \sigma_n^2)$. Now we choose a GP prior for the function describing the distribution, i.e,

$$f(.) \sim GP(0, k(.,.)), or$$
$$f_* \sim N(0, K(X_*, X_*))$$

A popular choice for the kernel function is the squared exponential function, i.e., $\sigma_f^2 exp(-|x-x'|^2/2\lambda^2)$. The 2 parameters are varied to minimise the risk function. For the noisy test data, we have the covariance function as

$$cov(y_p, y_q) = k(x_p, x_q) + \sigma_n^2 \delta_{pq}$$

From the structure of the covariance matrix mentioned above, we get the following form,

$$\begin{bmatrix} y \\ f_* \end{bmatrix} \sim N\Big(0, \begin{bmatrix} K(X,X) + \sigma_n^2 I & K(X,X_*) \\ K(X_*,X) & K(X_*,X_*) \end{bmatrix}\Big)$$

Using standard rules for conditioning Gaussians, we get

$$y_*|y, X, X_* \sim N(\mu_*, \Sigma_*)$$

where

$$\mu_* = K(X_*,X)(K(X,X) + \sigma_n^2 I)^{-1} y$$
$$\Sigma_* = K(X_*,X_*) + K(X_*,X)(K(X,X) + \sigma_n^2 I)^{-1} K(X,X_*)$$

As we can see this is another GP, with

$$m_t(x) = K(x,X)(K(X,X) + \sigma_n^2 I)^{-1} K(X,x)$$

$$k_t(x,x') = k(x,x') + K(x,X)(K(X,X) + \sigma_n^2 I)^{-1} K(X,x')$$

Comparing these with (7) we see a correspondence between the 2, i.e.,

$$K(C,D) = \Phi(C)^T \Sigma_p \Phi(D), ork(x_p, x_q) = \phi(x_p)^T \Sigma_p \phi(x_q)$$

Therefore, we see that every regression in the weight space view can be expressed in function space view Alternatively, we can see,

$$m_t(x) = \sum_{i=1}^{t} \alpha_i k(x_i, x)$$

, where α_is are the elements of the vector $(K(X,X) + \sigma_n^2 I)^{-1} y$. So, we see that the function space view corresponds to a weight space view with $k(x_i, x)$ as the basis functions. The sum is not infinite but restricted to the number of observations as we will only have

the information about the $(n+1)$ dimensional space. The output y for a particular value of input x_* is chosen such that it minimises the risk calculated on our choice of loss function, i.e.,

$$R(y_{guess}|x_*) = \int L(y_*, y_{guess}) p(y_*|x_*, X) dy_*$$

and $y_{optimal}|x_* = argminR(y_{guess}|x_*)$
where $L(y_*, y_{guess})$ is the loss function. For example, for squared loss function $y_{optimal}$ is the mean of the Gaussian predictive distribution.

4.1 Optimisation

The marginal likelihood is defined as

$$p(y|X) = \int p(y|f,X) p(f|X) df$$

which can then be written as

$$logp(y|X) = -\frac{1}{2} y^T K_y^{-1} y - \frac{1}{2} log|K_y| - \frac{n}{2} log2\pi$$

where $K_y = (K(X,X) + \sigma_n^2)$ So, we have 3 parameters which can be varied, i.e., $\sigma_f, \lambda, \sigma_n$ to maximise the log likelihood using partial derivatives with respect to all of them.

4.2 Kernels for GPR

We can use different kernels (or covariance function) for our GP according to our requirements.They determine the shape of the prior and posterior of the GP. They serve as a similarity function under the assumption that similar datapoints give similar outputs. We can incorporate some structure in data using these kernels too.

4.3 GPR as linear smoother and equivalent kernels

From the expression of the mean of the posterior, i.e., $\mu_* = K(X_*, X)(K(X,X) + \sigma_n^2 I)^{-1} y$, we see that it can be expressed as a linear combination of the training set targets. Thus it acts as a linear smoother over them. If we write $K = \sum \lambda_i u_i u_i^T$ where λ_is are the eigenvalues and u_is are the eigenvectors. As K is real and positive semi definite, its eigenvectors form a basis so we can write $y = \sum \gamma_i u_i$. Putting this together we get

$$\mu_* = \sum \frac{\gamma_i \lambda_i}{\lambda_i + \sigma_n^2} u_i$$

Therefore, we can see that the weight of the component of y along u_i depends on the factor $\gamma_i \lambda_i / \lambda_i + \sigma_n^2$.
Another way to visualise the sum is by using the concept of equivalent kernel, i.e., we define $\kappa_i(x) = \kappa(|x - x_i|/l)$ where κ is the equivalent kernel. Then we can write $\mu_* = \sum w_i y_i$ where w_i is $\kappa_i / \sum \kappa_j$. The shape of the equivalent kernel is different from the underlying kernel and are in general oscillatory.

4.4 Non zero mean prior

Above we only discussed the case where we take the mean of the GP prior to be zero but that may not always be the case. If we have a fixed prior mean, we simply apply the above technique to the difference of the observation and mean to get

$$\mu_* = m(X_*) + K(X_*, X)(K + \sigma_n^2)^{-1}(y - m(X))$$

Another researched technique is to specify a function $g(x) = f(x) + h(x)^T\beta$ which we can see as the usual regression with features $h(x)$ and parameters β with the residuals being characterised by a zero-mean GP $f(x)$. On fitting the model, we jointly optimise the parameters in β alongside the parameters of the covariance function of the GP. If we take the prior on β to be a gaussian $N(b, B)$, we get

$$g(x) \sim GP(h(x)^T b, k(x, x') + h(x)^T B h(x'))$$

Upon further calculation we get

$$g(X_*) = f(X_*) + R^T\bar{\beta}$$

$$cov(g_*) = cov(f_*) + R^T(B^{-1} + HK_y^{-1}H^T)^{-1}R$$

where $\bar{\beta} = (B^{-1} + HK_y^{-1}H^T)^{-1}(HK_y^{-1}y + B^{-1}b)$ and $R = H_* - HK_y^{-1}K_*$. The marginal likelihood in this case can be written as

$$logp(y|X, b, B) = -\frac{1}{2}(H^T b - y)^T(K_y + H^T BH)^{-1}(H^T b - y) - \frac{1}{2}log|K_y + H^T BH| - \frac{n}{2}log2\pi,$$

5 Conclusions

Therefore, we see Gaussian processes are a useful alternative to simple linear regression. It is useful as it is probabilistic so can be used to predict confidence intervals as well. It is non-parametric so can be used to predict arbitrary functions. Different kernels can be used so it can make use structure of data. However the method is not sparse as it uses all the data and if the data is high-dimensional then it is not very efficient.

6 References

- Gaussian Process by Chuong B. Do
- Gaussian Processes for Machine Learning
- A Tutorial on Gaussian Process Regression
- Gaussian Processes - scikit-learn