Fit a Gaussian Process

Diego Garrido

In this work we fit a Gaussian Process Regressor (GPR) with different kernel to different data set. Jupyter Notebook

1 Fitting GPs over a sinusoidal function

In this exercise we fit a Gaussian Process (GP) with different kernels to a sinusoidal function (see Figure 1). Based on fit on training set, the GP with the best performance, i.e., with higher likelihood, is the GP with Exponential Sine Squared (ESS) kernel, this has a log-likelihood of 3.394. However, GPs with kernels Radial Basic Funtion (RBF), Rational Quadratic (RQ) and Mattern kernel better capture smooth pattern of sinusoidal function. This can be seen by comparing the Root Mean Square Error (RMSE) in the intervale [0,5], where the RMSE of ESS is about twice the RMSE than the other kernels mentioned. Note that the RMSE is calculated by evaluating the sinusoidal function and the posterior mean on an equispaced sample of size 100 in the interval [0,5].

2 Fitting GPs over a sample generated by a GP

In this case, we generate 20 samples from a GP with RQ kernel and fit some GPs with different kernels (see Figure 2). Similarly, based on fit on training set, the GP with the best performance is the GP with monster kernel. This is because this kernel has great flexibility compared to the other kernels used. For that reason, this kernel is more likely to overfit data. To solve that problem, we can use a metric that penalize model complexity, such as the Bayesian Information Criteria (BIC), this metric penalizes the log-likelihood by the product between the number of free parameters d (in this case, the parameters to be fitted) and the logarithm of the number of samples, mathematically is

$$BIC = logp(D|\hat{\theta}_{ML}) - \frac{d}{2}log(N) \tag{1}$$

, based in BIC the best model is the RQ kernel, the same one that generates the training data.

3 Simplify kernel complexity

When adjusting multiple kernels, a good heuristic to simplify the model is to start by eliminating those kernels

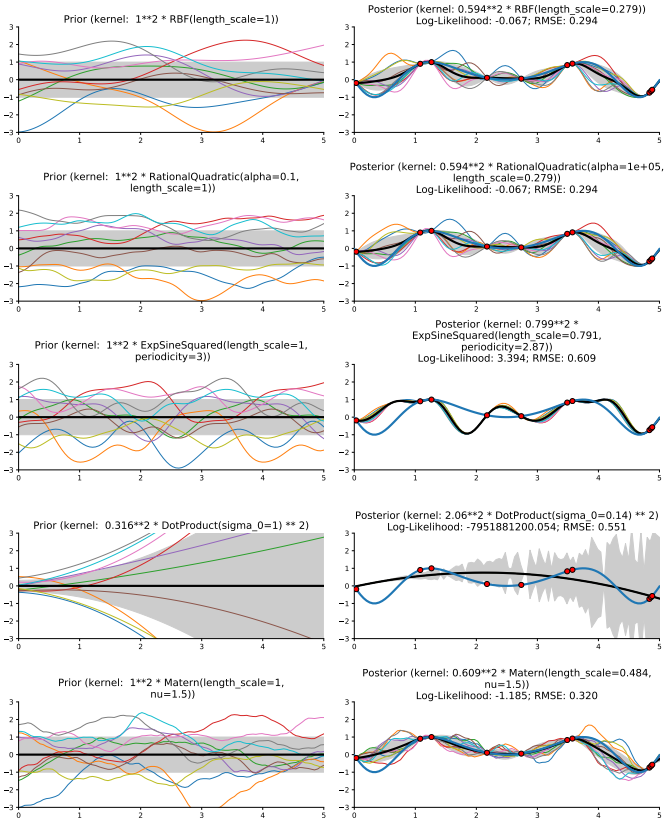

Figure 1: Prior and posterior of a Gaussian Process Regression with different kernels. Mean (black line), standard deviation (grey area), 10 samples (red points) and ground truth (blue line) are shown.

that contribute less in magnitude, that is, the kernels with the lowest scale factor ($sigma^2$). In this case for the monster kernel we have the follo parameters

, where we can see that scale factor (σ^2) from RQ and Matern are approximately five orders of magnitude lower than when used alone, and four orders less than RBF, ESS and Dot Product (DP).

4 Increasing kernel complexity

When kernels are added continuously the log-likelihood tends to increase (due to the increase in the complexity of the model), and by consequence the risk of overfitting too, this situacion is worse when we have small data or the phenomenon does not have high complexity. This problem can be solved evaluating the generalization

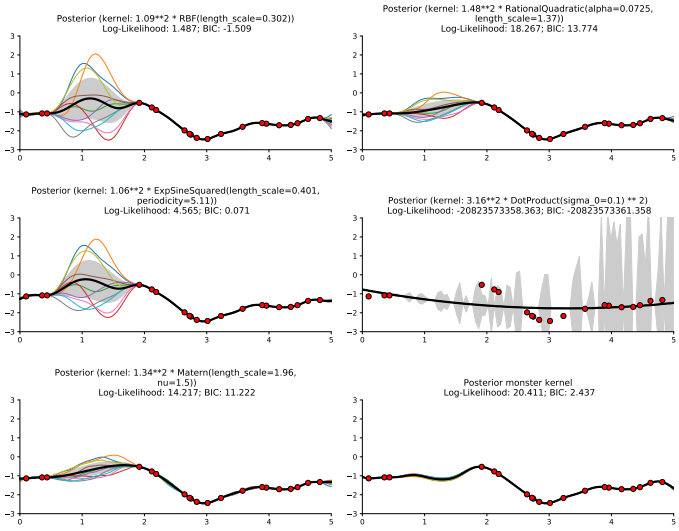

Figure 2: Posterior of a GPR with different kernels. Mean (black line), standard deviation (grey area) and 20 samples from a GP with RQ kernel (red points) are shown.

Monster kernel parameters: 0.669**2 * RBF(length_scale=1.37) + 0.00316**2 * RationalQuadratic(alpha=1.33e+03, length_scale=3.12) + 0.591**2 * ExpSineSquared(length_scale=0.837, periodicity=2.93) + 0.1**2 * DotProduct(sigma_0=1.8) ** 2 + 0.00316**2 * Matern(length_scale=0.158, nu=1.5)

Figure 3: Monster kernel parameter from of a GPR fitted to 20 samples from a GP with RQ kernel.

capacity in validation set (not seen on training time) from differents kernels, using metric like RMSE or Mean Absolute Error (MAE), since it is a regression problem. Alternatively, we can use information theory metrics such as BUC or Akaike Information Criteria (AIC), these measure penalize the log-likelihood by model complexity, in this way, a more complex model requires an increase in performance according to the increase in complexity.

5 Setting the init kernel parameter in a GP

We create a synthetic signal, where the training data $D = \{x_i, y_i\}_{i=1}^{10}$, the x_i are i.i.d as a $U(0, 5)$ and $y_i = x_i + cos(2\pi x_i)$ (see Figure 4). How the data look non-stationary and periodic we use a DP plus ESS kernel, this is illustrated in equations 2-4.

$$k_{DP}(x, x') = \sigma_0^2 + x_i x_j \tag{2}$$

$$k_{ESS}(x, x') = exp\left(\frac{2sin^2(\pi|x - x'|/p)}{l^2}\right) \tag{3}$$

$$k(x, x') = \sigma_{DP}^2 k_{DP}(x, x') + \sigma_{ESS}^2 k_{ESS}(x, x') \tag{4}$$

The objective function of a GP is not convex, this is why it becomes very important to choose the initial parameters correctly to start near a good local optimum. The init parameters for the k_{DP} were chosen using a linear regression. The linear regression was fitted on training data, with θ_0 as intercept and θ_1 as slope. Then the setting was σ_{DP}^2 equal to the slope and σ_0^2 equal to the intercept. The variance with respect to the mean estimated by the linear regression is used as σ_{ESS}^2, i.e., $\sigma_{ESS}^2 = \frac{\sum_{i=1}^{N}(\theta_0+\theta_1 x_i - y_i)^2}{N}$. The periodicity p was setted in 1 with bound between [1,10], this is due that in the interval [0,5] we has 10 samples, so the maximum peridiocity that we can observe is 10 (twice the interval size) and the minimum is 1 (twice the interval size divided by the number of samples). Finally, the ℓ parameter controls the smoothness, as this is not much, it was set to a value of 1 with a limit of [0.1,10].

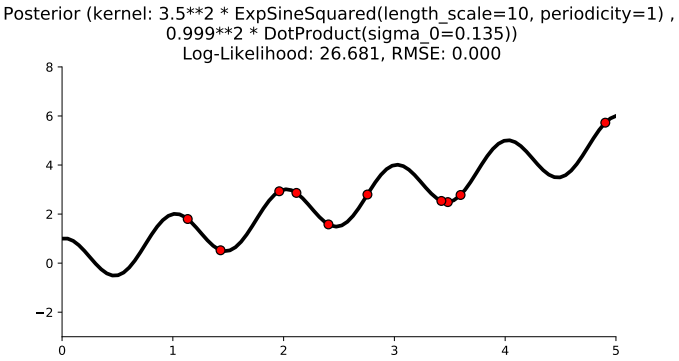

Figure 4: Posterior of a GPR with kernel DP plus ESS. Mean (black line) and 10 samples (red point).