Prediction with Gaussian Processes: Basic Ideas

Chris Williams

School of Informatics, University of Edinburgh, UK

Overview

- Bayesian Prediction
- Gaussian Process Priors over Functions
- GP regression
- GP classification

Bayesian prediction

- Define a prior over functions
- Observe data, obtain a posterior distribution over functions

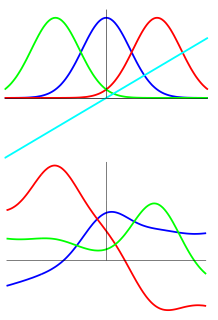
$$P(f|D) \propto P(f)P(D|f)$$

$$\text{posterior} \propto \text{prior} \times \text{likelihood}$$

- Make predictions by averaging predictions over the posterior $P(f|D)$
- Averaging mitigates *overfitting*

Bayesian Linear Regression

$$f(\mathbf{x}) = \sum_i w_i \phi_i(\mathbf{x}) \qquad \mathbf{w} \sim N(0, \Sigma)$$



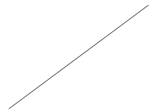
Samples from the prior

Gaussian Processes: Priors over functions

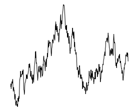
- For a stochastic process $f(\mathbf{x})$, mean function is
$$\mu(\mathbf{x}) = E[f(\mathbf{x})].$$
Assume $\mu(\mathbf{x}) \equiv 0 \; \forall \mathbf{x}$
- Covariance function
$$k(\mathbf{x}, \mathbf{x}') = E[f(\mathbf{x})f(\mathbf{x}')].$$
- Forget those weights! We should be thinking of defining priors over functions, not weights.
- Priors over function-space can be defined directly by choosing a covariance function, e.g.
$$k(\mathbf{x}, \mathbf{x}') = \exp(-w|\mathbf{x} - \mathbf{x}'|)$$
- Gaussian processes are stochastic processes defined by their mean and covariance functions.

Examples of GPs

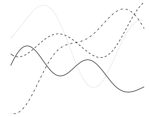
- $\sigma_0^2 + \sigma_1^2 x x'$



- $\exp -|x - x'|$



- $\exp -(x - x')^2$



Connection to feature space

A Gaussian process prior over functions can be thought of as a Gaussian prior on the coefficients $\mathbf{w} \sim N(0, \Lambda)$ where

$$f(\mathbf{x}) = \sum_{i=1}^{N_F} w_i \phi_i(\mathbf{x}) = \mathbf{w}.\Phi(\mathbf{x})$$

$$\Phi(\mathbf{x}) = \begin{pmatrix} \phi_1(\mathbf{x}) \\ \phi_2(\mathbf{x}) \\ \vdots \\ \phi_{N_F}(\mathbf{x}) \end{pmatrix}$$

In many interesting cases, $N_F = \infty$

Choose $\Phi(\cdot)$ as eigenfunctions of the kernel $k(\mathbf{x}, \mathbf{x}')$ wrt $p(\mathbf{x})$ (Mercer)

$$\int k(\mathbf{x}, \mathbf{y}) p(\mathbf{x}) \phi_i(\mathbf{x}) \, d\mathbf{x} = \lambda_i \phi_i(\mathbf{y})$$

Gaussian process regression

Dataset $\mathcal{D} = (\mathbf{x}_i, y_i)_{i=1}^n$, Gaussian likelihood $p(y_i|f_i) \sim N(0, \sigma^2)$

$$\bar{f}(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}, \mathbf{x}_i)$$

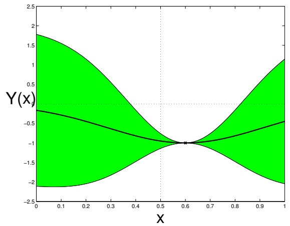
where

$$\boldsymbol{\alpha} = (K + \sigma^2 I)^{-1} \mathbf{y}$$

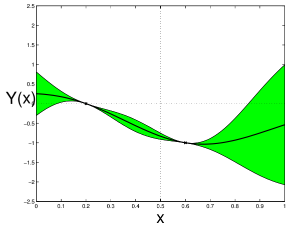
$$\text{var}(\mathbf{x}) = k(\mathbf{x}, \mathbf{x}) - \mathbf{k}^T(\mathbf{x})(K + \sigma^2 I)^{-1} \mathbf{k}(\mathbf{x})$$

in time $O(n^3)$, with $\mathbf{k}(\mathbf{x}) = (k(\mathbf{x}, \mathbf{x}_1), \ldots, \mathbf{k}(\mathbf{x}, \mathbf{x}_n))^T$

After 1 observation:



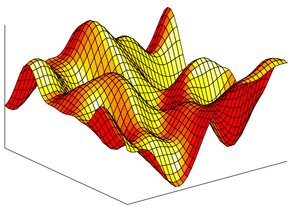
After 2 observations:



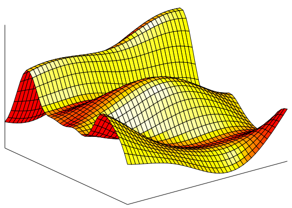
- Approximation methods can reduce $O(n^3)$ to $O(nm^2)$ for $m \ll n$
- GP regression is competitive with other kernel methods (e.g. SVMs)
- Can use non-Gaussian likelihoods (e.g. Student-t)

Adapting kernel parameters

$$k(\mathbf{x}^i, \mathbf{x}^j) = v_0 \exp -\frac{1}{2} \sum_{l=1}^{d} w_l (x_l^i - x_l^j)^2$$



$w_1 = 5.0 \quad w_2 = 5.0$



$w_1 = 5.0 \quad w_2 = 0.5$

- For GPs, the marginal likelihood (aka Bayesian evidence) $\log P(\mathbf{y}|\theta)$ can be optimized wrt the kernel parameters $\theta = (v_0, \mathbf{w})$
- For GP regression $\log P(\mathbf{y}|\theta)$ can be computed exactly

$$\log P(\mathbf{y}|\theta) == -\frac{1}{2} \log |K + \sigma^2 I| - \frac{1}{2} \mathbf{y}^T (K + \sigma^2 I)^{-1} \mathbf{y} - \frac{n}{2} \log 2\pi$$

Regularization

- $\bar{f}(\mathbf{x})$ is the (functional) minimum of

$$J[f] = \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - f(\mathbf{x}_i))^2 + \frac{1}{2}\|f\|^2_{\mathcal{H}}$$

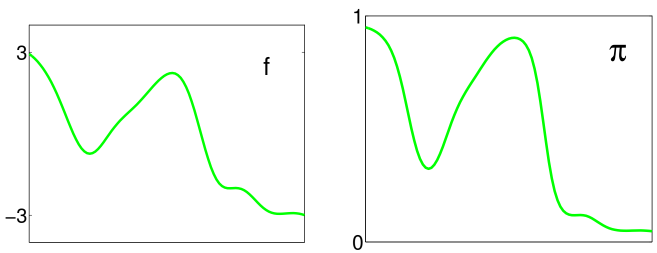
(1st term $=$ $-$ log-likelihood, 2nd term $=$ $-$ log-prior)

- However, the regularization framework does not yield predictive variance or marginal likelihood

Previous work

- Wiener-Kolmogorov prediction theory (1940's)
- Splines (Kimeldorf and Wahba, 1971; Wahba 1990)
- ARMA models for time-series
- Kriging in geostatistics (for 2-d or 3-d spaces)
- Regularization networks (Poggio and Girosi, 1989, 1990)
- Design and Analysis of Computer Experiments (Sacks et al, 1989)
- Infinite neural networks (Neal, 1995)

GP prediction for classification problems



Squash through logistic (or erf) function

- Likelihood

$$-\log P(y_i|f_i) = \log(1 + e^{-y_i f_i})$$

- Integrals can't be done analytically
  - Find *maximum a posteriori* value of $P(\mathbf{f}|\mathbf{y})$ (Williams and Barber, 1997)
  - Expectation-Propagation (Minka, 2001; Opper and Winther, 2000)
  - MCMC methods (Neal, 1997)

MAP Gaussian process classification

To obtain the MAP approximation to the GPC solution, we find $\hat{\mathbf{f}}$ that maximizes the convex function

$$\Psi(\mathbf{y}) = -\sum_{i=1}^{n} \log(1 + e^{-y_i f_i}) - \frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f} + c$$

The optimization is carried out using the Newton-Raphson iteration

$$\mathbf{f}^{new} = K(I + WK)^{-1}(W\mathbf{f} + (\mathbf{t} - \boldsymbol{\pi}))$$

where $W = \text{diag}(\pi_1(1-\pi_1), .., \pi_n(1-\pi_n))$ and $\pi_i = \sigma(\hat{f}_i)$. Basic complexity is $O(n^3)$

For a test point $\mathbf{x}_*$ we compute $\bar{f}(\mathbf{x}_*)$ and the variance, and make the prediction as

$$P(\text{class } 1|\mathbf{x}_*, \mathcal{D}) = \int \sigma(f_*) p(f_*|\mathbf{y}) df_*$$

SVMs

1-norm soft margin classifier has the form

$$f(\mathbf{x}) = \sum_{i=1}^{n} y_i \alpha_i^* k(\mathbf{x}, \mathbf{x}_i) + w_0^*$$

where $y_i \in \{-1, 1\}$ and $\boldsymbol{\alpha}^*$ optimizes the quadratic form

$$Q(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} t_i t_j \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j)$$

subject to the constraints

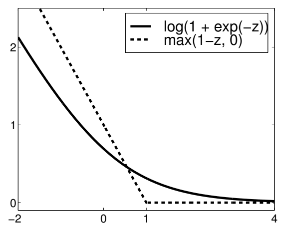
$$\sum_{i=1}^{n} y_i \alpha_i = 0$$

$$C \geq \alpha_i \geq 0, \qquad i = 1, \ldots, n$$

This is a *quadratic programming* problem. Can be solved in many ways, e.g. with interior point methods, or special purpose algorithms such as SMO.

Basic complexity is $O(n^3)$.

- Define $g_\sigma(z) = \log(1 + e^{-z})$
- SVM classifier is similar to GP classifier, but with g_σ replaced by $g_{SVM}(z) = [1 - z]_+$ (Wahba, 1999)



- Note that the MAP solution using g_σ solution is not sparse, but gives a probability output