Becs-114.4613 - Special Course in Bayesian Modelling

Regression in Gaussian Processes

SRIKANTH GADICHERLA

29TH JANUARY 2016

Agenda

- Why Gaussian Processes?
- What are Gaussian Processes?
- Weight Space view
  - Bayesian way of Regression
- Function Space view
  - Prior distribution over functions + calculating posterior distribution

Why Gaussian Processes?

What are Gaussian Processes?

- A Gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution.
- One can loosely think of a function as a very long vector, each entry in the vector specifying the function value $f(x)$.
- A Gaussian process is completely defined by mean $m(x)$ and covariance function $k(x, x')$.

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

$$m(x) = E[f(x)]$$

$$k(x, x') = E[(f(x) - m(x))\,(f(x') - m(x'))]$$

Gaussians provide the linear algebra of inference

- products of Gaussians are Gaussians
- marginals of Gaussians are Gaussians
- conditionals of Gaussians are Gaussians
- linear projections of Gaussians are Gaussians

Ridge Regression

Training set:

$$\mathcal{D} = \{(x_i, y_i) \mid i = 1, .., n\}$$

Linear regression:

$$f(x) = \langle w, \phi(x) \rangle$$

Ridge regression:

$$\widehat{w} = \underset{w \in \mathcal{K}}{\operatorname{argmin}} \sum_{i=1}^{m} (y_i - \langle \phi(x_i), w \rangle)^2 + \lambda \, \|w\|^2$$

The Gaussian Process is a Bayesian Generalization of the Ridge Regression!

Weight-view Space

Bayesian Way of Regression

- So, how do Bayesians use the Gaussian Processes?
  - Start with a prior
  - See your data, get likelihood.
  - Compute the posterior
- Get inferences from the posterior

Bayesian Linear Model – Gaussian Noise

- Consider a training set $\mathcal{D}$ of n observations,

$$\mathcal{D} = \{(x_i, y_i) \mid i = 1, .., n\}$$

where $x \in \mathbb{R}^D$ denotes the input vector (covariates), y denotes the target variable.

Design Matrix : $X = [\, x_1 \mid ... \mid x_n] \quad \in \mathbb{R}^D$

$$y = \begin{pmatrix} y_1 \\ \cdot \\ \cdot \\ y_n \end{pmatrix} \in \mathbb{R}^n$$

Bayesian Linear Model – Gaussian Noise

- Standard linear model can be written as

$$f(x) = x^T . w, \qquad y = f(x) + \varepsilon,$$

- Assumed that observed y differs from function value $f(x)$ by additive noise ε, further assumed that it's i.i.d with σ_n^2 is noise level

$$\varepsilon \sim \mathcal{N}(0, \sigma^2)$$

Bayesian Linear Model – Gaussian Noise

- The noise assumption together with the model directly gives rise to the likelihood.

$$p(y|X,w) = \prod_{i=1}^{n} p(y_i|x_i,w)$$

$$= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\,\sigma_n} \exp\left(-\frac{\left(y_i - x_i^T w\right)^2}{2\sigma_n^2}\right)$$

$$= \frac{1}{(2\pi\sigma_n^2)^{n/2}} \exp\left(-\frac{|\boldsymbol{y} - X^T w|^2}{2\sigma_n^2}\right)$$

$$= \mathcal{N}(X^T w, \sigma^2 I)$$

Bayesian Linear Model – Gaussian Noise

- Bayesian formalism have to specify prior over the parameters. Taking zero mean Gaussian with $\sum_p$ covariance.

$$w \sim \mathcal{N}(\,0, \sum_p)$$

- Calculate the posterior

$$p(w|X,y) = \frac{p(y|X,w)p(w)}{p(y|X)}$$

$$= \frac{p(y|X,w)p(w)}{\int p(y|X,w)dw}$$

$$= \frac{\mathcal{N}_y(X^T w, \sigma^2 I_n)\mathcal{N}(\,0, \sum_p)}{\int \mathcal{N}_y(X^T w, \sigma^2 I_n)\mathcal{N}(\,0, \sum_p)dw}$$

$$\propto \mathcal{N}_y(X^T w, \sigma^2 I_n)\mathcal{N}(\,0, \sum_p)$$

Bayesian Linear Model – Gaussian Noise

Ridge Regression

$$p(w|X,y) \propto \mathcal{N}_y(X^T w, \sigma^2 I_n)\mathcal{N}(0, \Sigma_p)$$

$$\propto \exp\left\{\frac{-1}{2\sigma^2}(y - X^T w)^T (y - X^T w)\right\} \exp\{w^T \Sigma_p{}^{-1} w\}$$

$$\propto \exp\left\{\frac{-1}{2}(\mathrm{w} - \bar{w})^T \underbrace{\left(\frac{1}{\sigma^2} XX^T + \Sigma_p{}^{-1}\right)}_{A} (\mathrm{w} - \bar{w})\right\}$$

$$\propto \boxed{\mathcal{N}_w(-\bar{w},\ A^{-1})}$$ (*"Completing the square"*)

where $\boxed{\bar{w} = \sigma^{-2} \underbrace{(\sigma^{-2} XX^T + \Sigma_p{}^{-1})^{-1}}_{A^{-1} \in \mathbb{R}^{D \times D}} Xy} \in \mathbb{R}^D$

$$\boxed{A = (\sigma^{-2} XX^T + \Sigma_p{}^{-1})} \in \mathbb{R}^{D \times D}$$

Bayesian Linear Model – Gaussian Noise

- To make predictions for a test case we average over all possible parameter values, weighted by their posterior probability.
- The predictive distribution $f_* \triangleq f(x_*)$ at x_* is given by averaging the output of all possible linear models w.r.t to Gaussian posterior

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

$$= \mathcal{N}_{f_*}\left(\frac{1}{\sigma_n^2}\, {x_*}^T A^{-1} X y, {x_*}^T A^{-1} x_*\right)$$

$$= \boxed{\mathcal{N}_{f_*}\left({x_*}^T \bar{w}, {x_*}^T A^{-1} x_*\right)}$$

The predictive uncertainties grow with the magnitude of the test input, as one would expect in a linear model.

$Note: Posterior\ covariance\ doesn't\ depend\ on\ y.$

Projection into Feature Space

Bayesian linear model suffers from limited expressiveness.

To overcome this problem, inputs are projected to feature space and then linear model is applied.

explicit features : $\emptyset(x) = [1,\ x,\ x^2,\ x^3, \ldots]$ or Chebyshev Polynomials

Implicit features : $k(\bar{x}, \bar{y}) = \exp(-\|\bar{x} - \bar{y}\|^2)$ "kernels"

Computational savings is high when dimensionality of feature space large when compared to data samples

- "Linear in the parameters"

  As long as the projections are fixed functions (i.e., independent of parameter **w**) the model is *'linear in parameters'* and therefore analytically tractable.

Projection into Feature Space

- **Explicit features** : Now, the model is

$$y = \emptyset(x)^T w + \epsilon$$

- The predictive distribution is

$$p(\underbrace{f_*|x_*}_{\emptyset(x_*)^T \mathrm{w}}, X, y) = \mathcal{N}_{f_*}(\emptyset(x_*)^T \overline{w}, \emptyset(x_*)^T A^{-1} \emptyset(x_*))$$

$$\overline{w} = \sigma^{-2} \underbrace{(\sigma^{-2}\emptyset(X)\emptyset(X)^T + \Sigma_p{}^{-1})^{-1}}_{A^{-1} \in \mathbb{R}^{N \times N}} \emptyset(X) y \in \mathbb{R}^D$$

$$A = (\sigma^{-2}\emptyset(X)\emptyset(X)^T + \Sigma_p{}^{-1}) \in \mathbb{R}^{N \times N}$$

Projection into Feature Space

- Alternative formalism, to avoid $N \times N$ matrix inversion

$$\mathcal{N}_{f_*}\left(\underbrace{(\emptyset_*^T \Sigma_p \phi)}_{\mathbb{R}^{N\times N}}\underbrace{(K + \sigma^2 I_n)}_{\mathbb{R}^{N\times N}}{}^{-1} y, \underbrace{(\emptyset_*^T \Sigma_p \phi)}_{\mathbb{R}^{N\times N}} - \underbrace{(\emptyset_*^T \Sigma_p \phi)}_{\mathbb{R}^{N\times N}}\underbrace{(K + \sigma^2 I_n)}_{\mathbb{R}^{N\times N}}{}^{-1}\underbrace{(\phi^T \Sigma_p \phi_*)}_{\mathbb{R}^{N\times N}}\right)$$

Function-view Space

Motivation

- Get inferences directly in the function space.
- Gaussian process is used to describe a distribution over functions.

Gaussian Processes in Function Space

- For each $x \in \mathbb{R}^D$ a Guassian variable $f(x)$ is associated such that

$$f(x) \sim \mathcal{N}_{f(x)}\left(m(x), k(x,x)\right)$$

$$\begin{bmatrix} f(x) \\ f(\tilde{x}) \end{bmatrix} \sim \mathcal{N}_{\begin{bmatrix} f(x) \\ f(\tilde{x}) \end{bmatrix}} \left\{ \begin{bmatrix} m(x) \\ \widetilde{m(x)} \end{bmatrix}, \begin{bmatrix} k(x,x) & k(\tilde{x},x) \\ k(x,\tilde{x}) & k(\tilde{x},\tilde{x}) \end{bmatrix} \right\}$$

Bayesian Linear Model…revisited!

- "Bayesian linear model is a Gaussian Process"

$$f(x) = \emptyset(x)^T w \quad \in \mathbb{R}$$

$$w \sim \mathcal{N}(0, \Sigma_p)$$

where $\emptyset(x), \mathrm{w} \in \mathbb{R}^N$

- $[f(x_1), \ldots, f(x_k)]$ are jointly Gaussian $\forall\, x_1 \ldots. x_k$, thus f is a GP.

$$\mathbb{E}[f(x)] = \emptyset(x)^T\, \mathbb{E}[w] = 0$$

$$\mathbb{E}[f(x)f(x')] = \emptyset(x)^T\, \mathbb{E}[ww^T]\emptyset(x') = \emptyset(x)^T \textstyle\sum_p \emptyset(x') = k(x, x')$$

Bayesian Linear Model…revisited!

- Covariance function : Squared Exponential

$$cov\left(f(x_p), f(x_q)\right) = k(x_p, x_q) = \exp(-\frac{1}{2}\left|x_p - x_q\right|^2)$$

Intuition: variables close to each other are highly correlated (close to unity) than others.

Note, that covariance between output is written as a function of the inputs

Noiseless Data Predictions

- Consider a training set $\mathcal{D}$ of n noise-free observations,

$$\mathcal{D} = \{(x_i, y_i) \mid i = 1, .., n\}$$

- $$\left.X = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix} \in \mathbb{R}^{n \times D}; \quad f = \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} \in \mathbb{R}^n \right\} n\ training\ inputs\ and\ targets$$

- $$\left.X_* = \begin{bmatrix} x_{*1}^T \\ \vdots \\ x_{*m}^T \end{bmatrix} \in \mathbb{R}^{m \times D}; \quad f_* = \begin{bmatrix} f_{*1} \\ \vdots \\ f_{*m} \end{bmatrix} \in \mathbb{R}^m \right\} m\ testing\ inputs\ and\ targets$$

Noiseless Data Predictions

$$\begin{bmatrix} f \\ f_* \end{bmatrix} \sim \mathcal{N}_{\begin{bmatrix} f \\ f_* \end{bmatrix}} \left\{ \begin{bmatrix} 0_n \\ 0_m \end{bmatrix}, \underbrace{\begin{bmatrix} k(X,X) & k(X,X_*) \\ k(X_*,X) & k(X_*,X_*) \end{bmatrix}}_{\in \mathbb{R}^{m+n \times m+n}} \right\}$$

- Aim : to get the posterior $\mathrm{p}(f_* \mid X_*, X, f)$

Computationally inefficient

- One way is to restrict the joint prior distribution to contain only those functions which agree with the data
- Calculate posterior analytically

Noiseless Data Predictions

$$p(f_*|X_*, X, f) \sim \mathcal{N}_{f_*}(k(X_*, X)k(X, X)^{-1}f,\ k(X_*, X_*) - k(X_*, X)k(X, X)^{-1}k(X, X_*))$$

Remarks:

1. Covariance here is zero as the data is noise free.
2. The result is similar to one derived in explicit feature.

$$\mathcal{N}_{f_*}((\emptyset_*^T \sum_p \phi)(K + \sigma^2 I_n)^{-1}y, k(X_*, X_*) - (\emptyset_*^T \sum_p \phi)(K + \sigma^2 I_n)^{-1}k(X, X_*)(\phi^T \sum_p \phi_*))$$

Noisy Data Predictions

- Model : $y = f(x) + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2)$

  Assuming, the noise is i.i.d Gaussian noise with variance σ_n^2.

- The joint distribution of the observed target values and test functions

$$\begin{bmatrix} y \\ f_* \end{bmatrix} \sim \mathcal{N}_{\begin{bmatrix} y \\ f_* \end{bmatrix}} \left\{ \begin{bmatrix} 0_n \\ 0_m \end{bmatrix}, \underbrace{\begin{bmatrix} k(X,X) + \sigma^2 I_n & k(X,X_*) \\ k(X_*,X) & k(X_*,X_*) \end{bmatrix}}_{\in \mathbb{R}^{m+n \times m+n}} \right\}$$

Noisy Data Predictions

- The posterior for the noisy data is

$$p(f_*|X, y, X_*) \sim \mathcal{N}_{f_*}\left(\bar{f}_*, cov(f_*)\right)$$

where $f_* \triangleq E(f_*|X, y, X_*) = k(X_*, X)(k(X, X) + \sigma^2 I_n)^{-1}\mathrm{y}$

$$\mathrm{cov}(f_*) = k(X_*, X_*) - k(X_*, X)(k(X, X) + \sigma^2 I_n)^{-1} k(X, X_*)$$

Noisy Data Predictions

- Numerical computation considerations

input: X (inputs), $\mathbf{y}$ (targets), k (covariance function), σ_n^2 (noise level), $\mathbf{x}_*$ (test input)

$L := \text{cholesky}(K + \sigma_n^2 I)$

$\boldsymbol{\alpha} := L^\top \backslash (L \backslash \mathbf{y})$

$\bar{f}_* := \mathbf{k}_*^\top \boldsymbol{\alpha}$ } predictive mean eq. (2.25)

$\mathbf{v} := L \backslash \mathbf{k}_*$

$\mathbb{V}[f_*] := k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{v}^\top \mathbf{v}$ } predictive variance eq. (2.26)

$\log p(\mathbf{y}|X) := -\frac{1}{2}\mathbf{y}^\top \boldsymbol{\alpha} - \sum_i \log L_{ii} - \frac{n}{2} \log 2\pi$ eq. (2.30)

return: $\bar{f}_*$ (mean), $\mathbb{V}[f_*]$ (variance), $\log p(\mathbf{y}|X)$ (log marginal likelihood)

Varying the Hyperparameters

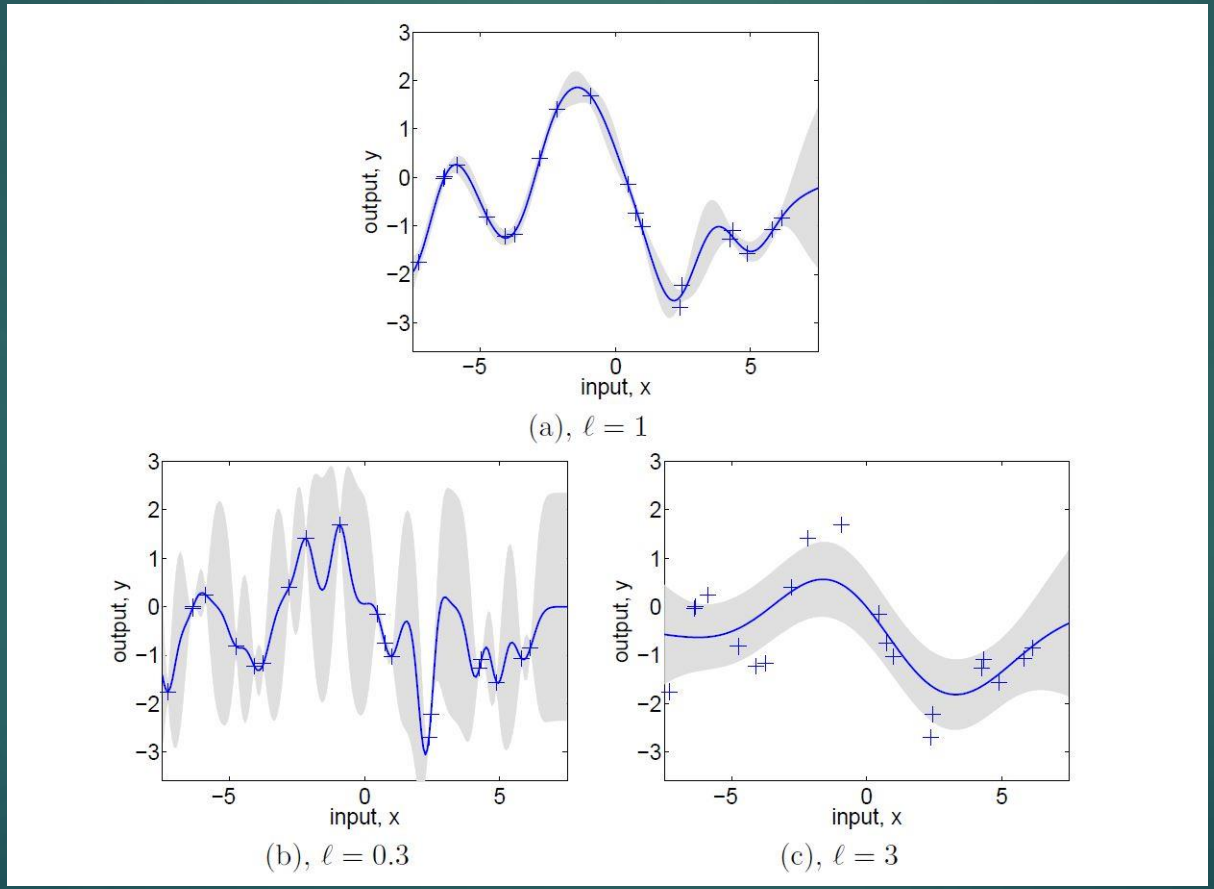
Covariance Function

- Consider the covariance function,

$$k_y(x_p, x_q) = \sigma_f^2 \exp\left(-\frac{1}{2l^2}(x_p - x_q)^2\right) + \sigma_n^2 \delta_{pq}$$

- The above equation has a few free parameters.

Covariance Function



Decision Theory of Regression

- Define a loss function, $\mathcal{L}(y_{true}, y_{guess})$
- Goal is to make the point prediction y_{guess} which incurs smallest loss.

How is it possible if we don't know y_{true} ?

- Instead, we minimize the expected loss by averaging w.r.t our model opinion as to what the truth might be

$$\tilde{R}_{\mathcal{L}}(y_{guess}|x_*) = \int \mathcal{L}(y_*, y_{guess})\ p(y_*|x_*, \mathcal{D})$$