CSC 412/2506:
Probabilistic Learning and Reasoning
Week 12 - 2/2: Gaussian Processes

Murat A. Erdogdu

University of Toronto

Overview

- Continuing in our theme of probabilistic methods.
  - Building on the kernel viewpoint of regression,
  - we focus on Gaussian processes.
  - We dispense with the parametric model and define a prior distribution over functions directly.

Recap: Linear Regression

- Given a training set of inputs and targets $\{(\mathbf{x}^{(i)}, t^{(i)})\}_{i=1}^{N}$
- Linear model:

$$y = \mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x})$$

  where $\boldsymbol{\psi}(\mathbf{x})$ is the feature map.
- Vectorized, we have the design matrix $\mathbf{X}$ in input space and

$$\boldsymbol{\Psi} = \begin{bmatrix} - & \boldsymbol{\psi}(\mathbf{x}^{(1)}) & - \\ - & \boldsymbol{\psi}(\mathbf{x}^{(2)}) & - \\ & \vdots & \\ - & \boldsymbol{\psi}(\mathbf{x}^{(N)}) & - \end{bmatrix}$$

  and predictions

$$\mathbf{y} = \boldsymbol{\Psi}\mathbf{w}.$$

Recap: Linear Regression Model

- We gave linear regression a probabilistic interpretation by assuming a Gaussian noise model:

$$t \mid \mathbf{x} \sim \mathcal{N}(\mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x}),\ \sigma^2)$$

- and a Gaussian prior

$$\mathbf{w} \sim \mathcal{N}(0, \alpha^{-1} I)$$

- Prior induces a probability distribution over

$$\mathbf{y} = \boldsymbol{\Psi}\mathbf{w}$$

Distribution over prediction function

- In practice, we evaluate the prediction function $y(\mathbf{x})$ at specific points, for example at the training data points $\mathbf{x}^{(i)}$ for $i = 1, ..., N$.
- So we are interested in the joint distribution of the function values

$$y(\mathbf{x}^{(1)}), ..., y(\mathbf{x}^{(N)})$$

which we denote by the vector $\mathbf{y}$.

- We have

$$\mathbf{y} = \mathbf{\Psi}\mathbf{w} \qquad \mathbf{w} \sim \mathcal{N}(0, \alpha^{-1} I)$$

- Thus

$$\mathbf{y} \sim \mathcal{N}(0, \boldsymbol{K}) \qquad \boldsymbol{K} = \frac{1}{\alpha}\mathbf{\Psi}\mathbf{\Psi}^T$$

where $\boldsymbol{K}$ is the (scaled) Gram matrix

$$K_{ij} = \frac{1}{\alpha} k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \frac{1}{\alpha} \boldsymbol{\psi}(\mathbf{x}^{(i)})^T \boldsymbol{\psi}(\mathbf{x}^{(j)})$$

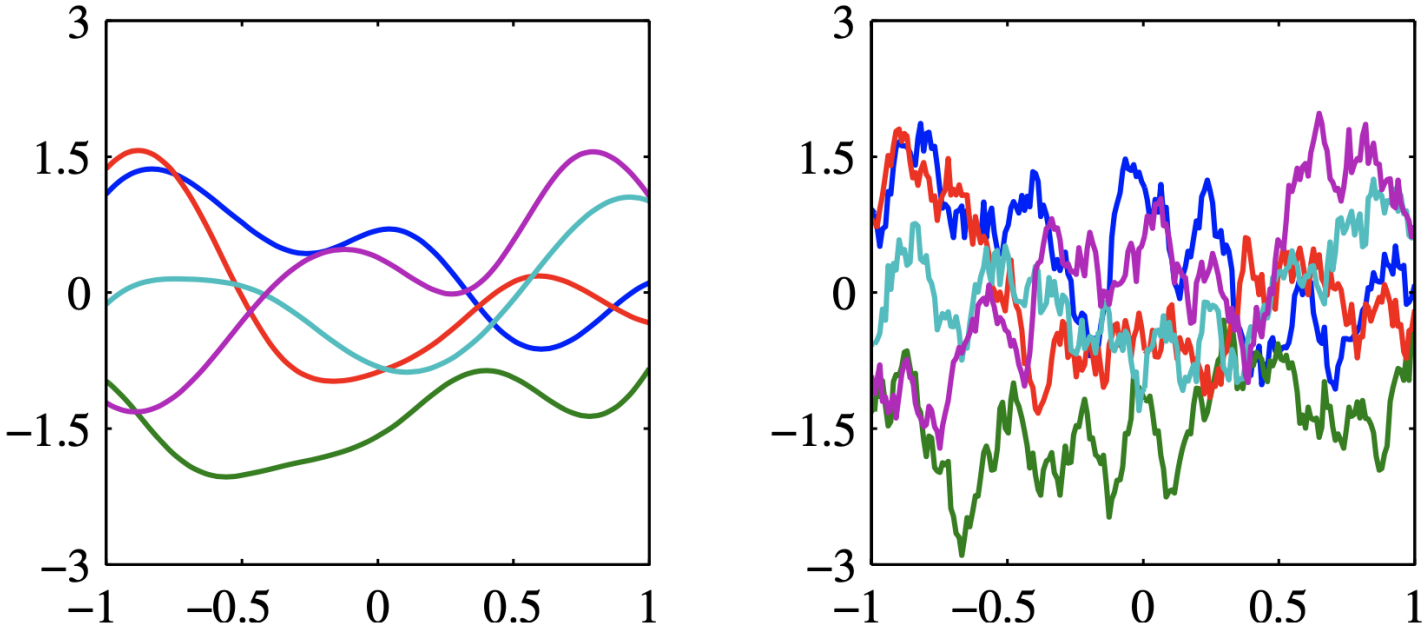
Gaussian process

- A Gaussian process is a probability distribution over functions $y(\mathbf{x})$ such that for any set of values of $y(\mathbf{x})$ evaluated at an arbitrary set of points $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ...$ is jointly Gaussian.
- The case $\mathbf{x} \in \mathbb{R}^2$ is called Gaussian random field.
- The joint distribution is specified completely by the second-order statistics, i.e. the mean and the covariance.
- In most applications, the mean of $y(\mathbf{x})$ can be set to zero.
- Thus, Gaussian process is completely specified by the covariance

$$\mathbb{E}[y(\mathbf{x}^{(i)})y(\mathbf{x}^{(j)})] = \frac{1}{\alpha}k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$$

Gaussian process

- We can directly define the kernel of a Gaussian process, not worrying about the feature map.



Samples from Gaussian processes for a Gaussian kernel (left) and an exponential kernel (right).

Gaussian processes for regression

- We have the linear model

$$t \mid \mathbf{x} \sim \mathcal{N}(y(\mathbf{x}),\ \sigma^2) \qquad y(\mathbf{x}) = \mathbf{w}^\top \boldsymbol{\psi}(\mathbf{x})$$

- Given N samples, we have

$$\mathbf{t} \mid \mathbf{y} \sim \mathcal{N}(\mathbf{y},\ \sigma^2 I)$$

- Since $\mathbf{y}$ is a Gaussian process, we have $\mathbf{y} \sim \mathcal{N}(0, \boldsymbol{K})$.
- Therefore the marginal of $\mathbf{t}$ is given by

$$\mathbf{t} \sim \mathcal{N}(0, \boldsymbol{C}) \qquad \boldsymbol{C} = \boldsymbol{K} + \sigma^2 I$$

where

$$C(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \frac{1}{\alpha} k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) + \sigma^2 \delta_{ij}$$

$\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ otw.

Gaussian processes for regression

- Let's define $\mathbf{t}_N = (t^{(1)}, t^{(2)}, ..., t^{(N)})^T$.
- We have the marginal of $\mathbf{t}_N$ given by

$$\mathbf{t}_N \sim \mathcal{N}(0, \boldsymbol{C}_N) \qquad \boldsymbol{C}_N = \boldsymbol{K}_N + \sigma^2 I$$

  where $C_N(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \frac{1}{\alpha} k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) + \sigma^2 \delta_{ij}$.
- This reflects the two Gaussian sources of randomness.
- **Goal in regression:** We want to predict for a new input $\mathbf{x}^{(N+1)}$!
- We need

$$p(t^{(N+1)} \mid \mathbf{t}_N)$$

- Note that $\mathbf{x}^{(i)}$'s are treated as constants.

Gaussian processes for regression

- We have

$$\mathbf{t}_{N+1} \sim \mathcal{N}(0, \boldsymbol{C}_{N+1}) \qquad \boldsymbol{C}_{N+1} = \boldsymbol{K}_{N+1} + \sigma^2 I$$

  where

$$C_{N+1}(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \frac{1}{\alpha} k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) + \sigma^2 \delta_{ij}$$

$$\boldsymbol{C}_{N+1} = \begin{bmatrix} \boldsymbol{C}_N & \mathbf{k} \\ \mathbf{k}^T & c \end{bmatrix}.$$

  - Here, $c = \frac{1}{\alpha} k(\mathbf{x}^{(N+1)}, \mathbf{x}^{(N+1)}) + \sigma^2$
  - $\mathbf{k}$ is a vector with entries $k_i = \frac{1}{\alpha} k(\mathbf{x}^{(i)}, \mathbf{x}^{(N+1)})$
- Since $\mathbf{t}_{N+1}$ is multivariate Gaussian, we can easily find $t^{(N+1)} \mid \mathbf{t}_N$.

Property of Multivariate Gaussian Distribution

- If we have $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \qquad \boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix} \qquad \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix}$$

- Then,

$$\mathbf{x}_1 \,|\, (\mathbf{x}_2 = \boldsymbol{a}) \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{C})$$

with

$$\boldsymbol{m} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{a} - \boldsymbol{\mu}_2) \quad \boldsymbol{C} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}.$$

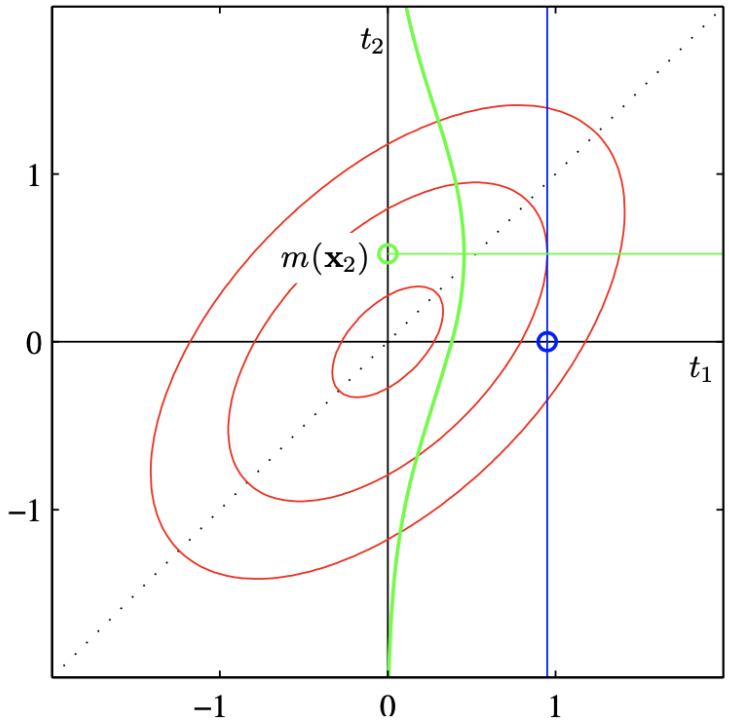
Gaussian processes for regression

- Since $\mathbf{t}_{N+1}$ is multivariate Gaussian, $t^{(N+1)} \,|\, \mathbf{t}_N$ is also Gaussian with mean and covariance

$$m(\mathbf{x}^{(N+1)}) = \mathbf{k}^T \boldsymbol{C}_N^{-1} \mathbf{t}_N \qquad \sigma^2(\mathbf{x}^{(N+1)}) = c - \mathbf{k}^T \boldsymbol{C}_N^{-1} \mathbf{k}$$

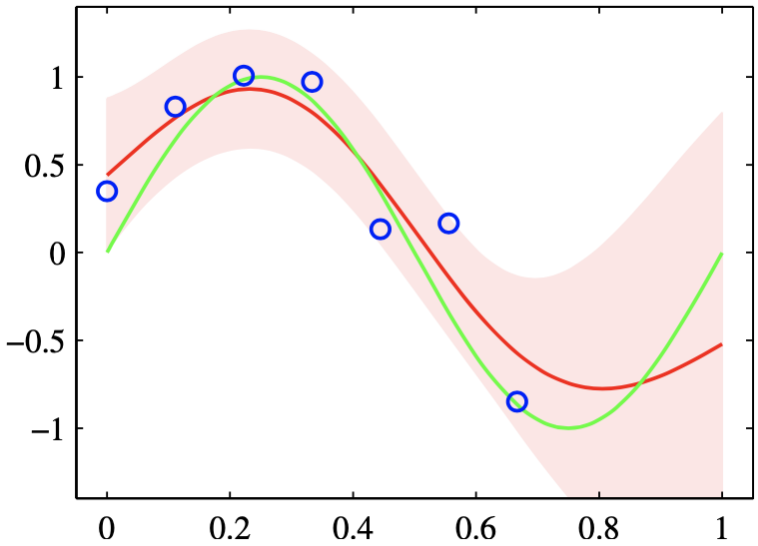
- These are the key results that define Gaussian process regression.
- The vector $\mathbf{k}$ is a function of the new test input $\mathbf{x}^{(N+1)}$.
- The predictive distribution is a Gaussian whose mean and variance both depend on $\mathbf{x}^{(N+1)}$.

GPs illustration



- One training point and one test point: red ellipses show contours of the joint $p(t^{(1)}, t^{(2)})$.
- Conditioning on $t^{(1)}$ corresponds to the vertical blue line. $p(t^{(2)}|t^{(1)})$ is shown by the green curve.

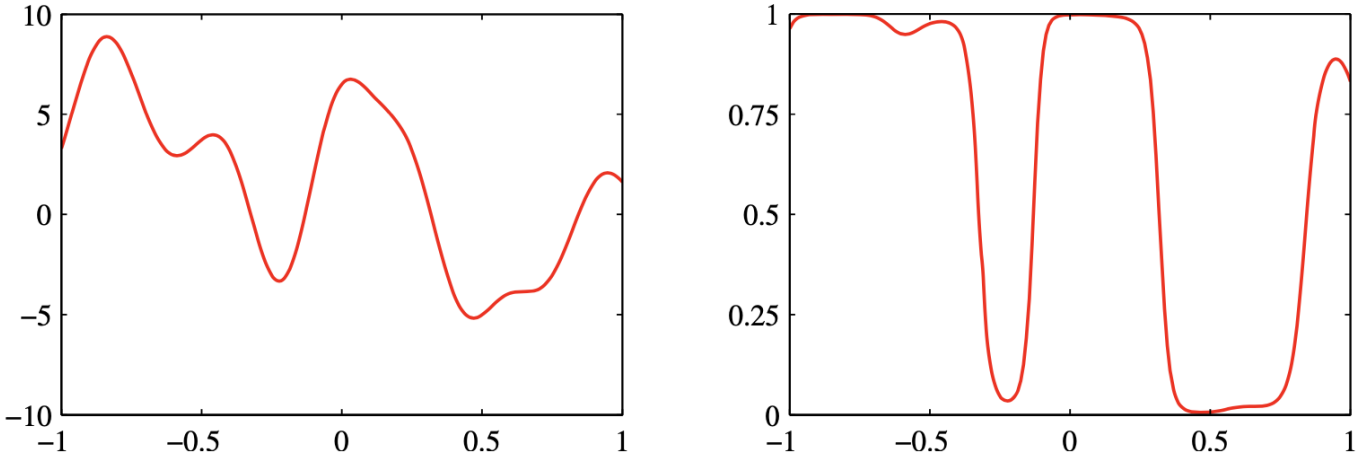
GPs for regression



- The green curve is the true sinusoidal function from which the data points, shown in blue, are obtained by sampling and addition of Gaussian noise.
- The red line shows the mean of the Gaussian process predictive distribution.
- The shaded region corresponds to plus and minus two standard deviations.

GPs for classification

- Consider a classification problem with target variables $t \in \{0, 1\}$
- We define a Gaussian process over a function $a(\mathbf{x})$ and then transform the function using sigmoid $y(\mathbf{x}) = \sigma(a(\mathbf{x}))$.
- We obtain a non-Gaussian stochastic process over functions $y(\mathbf{x}) \in (0, 1)$.



Left: $a(\mathbf{x})$ Right: $y(\mathbf{x})$

GPs for classification

- The probability distribution over target is then given by

$$p(t|a) = \sigma(a)^t(1-\sigma(a))^{1-t}$$

- We need to compute

$$p(t^{(N+1)} \,|\, \mathbf{t}_N)$$

  and notice that $a(\mathbf{x})$ is a Gaussian process but $y(\mathbf{x})$ is not. Therefore, $t^{(N+1)} \,|\, \mathbf{t}_N$ won't be Gaussian.

- We have

$$\boldsymbol{a}_{N+1} \sim \mathcal{N}(0, \boldsymbol{C}_{N+1})$$

  where

$$C_{N+1}(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \frac{1}{\alpha} k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) + \nu\delta_{ij}.$$

GPs for classification

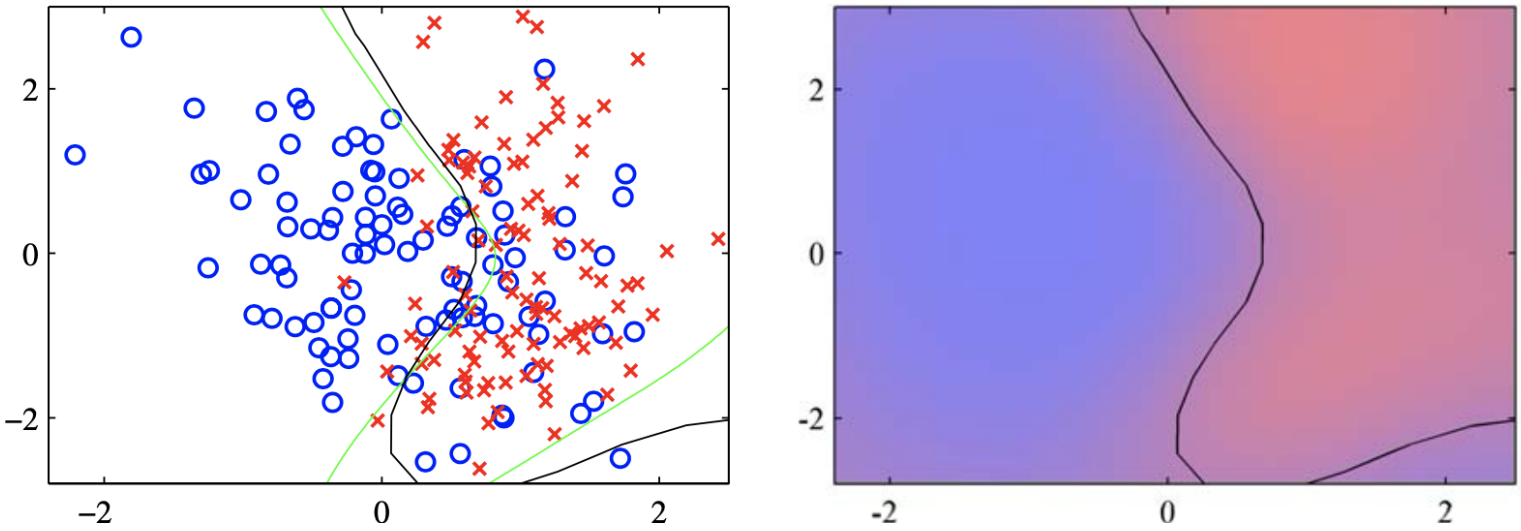
- It is sufficient to find $p(t^{(N+1)} \mid \mathbf{t}_N)$!
- So we write

$$p(t^{(N+1)} \mid \mathbf{t}_N) = \int p(t^{(N+1)} \mid \boldsymbol{a}_{N+1}) p(\boldsymbol{a}_{N+1} \mid \mathbf{t}_N) d\boldsymbol{a}_{N+1}$$

- This is an integral, so it is intractable!
- One needs to rely on MCMC based methods, or numerical integration to approximate this integral.

GPs for classification

- Illustration of GPs for classification:



- Left: optimal decision boundary from the true distribution in green, and the decision boundary from the Gaussian process classifier in black.
- Right: predicted posterior for the blue and red classes together with the Gaussian process decision boundary.

Learning the hyperparameters

- We didn't do any learning other than choosing a kernel!
- Rather than fixing the covariance function, we may prefer to use a parametric family of functions and then infer the parameter values from the data.
- Denoting the hyperparameters with θ, one can easily write down the likelihood of the Gaussian process model.

$$\log p(\mathbf{t} \mid \theta) = -\frac{1}{2} \log |\boldsymbol{C}_N| - \frac{1}{2} \mathbf{t}^T \boldsymbol{C}_N^{-1} \mathbf{t} - \frac{N}{2} \log(2\pi)$$

- The next step is standard: gradient based optimization.

Summary

- Gaussian processes are flexible tools that can be used in regression and classification tasks.
- One can simply choose a kernel and find the predictive density!
- They can be used together with modern tools, creating powerful learning methods.