STA 437/2005: Methods for Multivariate Data

Week 4: Gaussian Processes

Piotr Zwiernik

University of Toronto

Table of contents

Marginal distribution of MVN

Consider the following reformulation of the earlier result:
Suppose $X \sim N_m(\mu, \Sigma)$. Let $T := \{1, \dots, m\}$ and define

- $m : T \to \mathbb{R}$ such that $m(t) := \mu_t$ (mean function)
- $k : T \times T \to \mathbb{R}$ such that $k(s,t) := \Sigma_{st}$ (kernel function)

Then for every $A = \{t_1, \dots, t_n\} \subseteq T$, the vector $X_A = (X_{t_1}, \dots, X_{t_n})$ is Gaussian with

- The mean μ_A whose i-th entry is $m(t_i)$.
- The covariance matrix Σ_{AA} whose (i,j)-th entry is $k(t_i, t_j)$.

Gaussian Processes - an immediate generalization

A **Gaussian Process (GP)** is a generalization of the multivariate normal distribution to a collection of random variables indexed by an arbitrary set T.

Definition

A Gaussian Process is a collection of random variables $\{X_t\}_{t \in T}$ such that for any finite set of points $\{t_1, \ldots, t_n\} \subset T$, the corresponding vector $(X_{t_1}, \ldots, X_{t_n})$ follows a multivariate normal distribution.

In what follows we assume $T \subseteq \mathbb{R}^d$ with the Euclidean distance metric.

The mean and the kernel functions

A GP is characterized by:

- A **mean function** $m : T \to \mathbb{R}$: $\quad m(t) = \mathbb{E}[X_t]$
- A **kernel function** $k : T \times T \to \mathbb{R}$: $\quad k(t, t') = \mathrm{Cov}(X_t, X_{t'})$

Note that m is pretty much arbitrary (often set to be zero) but k is highly constrained:
POsitive semi-definitness: For any finite set $\{t_1, \ldots, t_n\}$, the covariance matrix Σ with entries $\Sigma_{ij} = k(t_i, t_j)$ is positive semi-definite.

We can use feature maps $\psi : \mathbb{R}^d \to \mathbb{R}^p$ to define kernels:

$$k(s, t) = \psi(s)^\top \psi(t).$$

Feature maps define kernels but not all kernels are like that (this can be generalized to "infinite dimensional" feature maps).

Feature map defines a kernel

- Let $k(\mathbf{x}, \mathbf{x}') = \psi(\mathbf{x})^\top \psi(\mathbf{x}')$
- The kernel matrix is given as $\Sigma_{ij} = k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$, $\Sigma = \mathbf{y}\mathbf{y}^\top$.
- We show that this matrix is positive semi-definite, $\forall \mathbf{u} \in \mathbb{R}^N$,

$$\boldsymbol{u}^\top \Sigma \boldsymbol{u} = \boldsymbol{u}^\top \mathbf{y}\mathbf{y}^\top \boldsymbol{u} = (\mathbf{y}^\top \boldsymbol{u})^\top \mathbf{y}^\top \boldsymbol{u} = \|\mathbf{y}^\top \boldsymbol{u}\|^2 \geq 0.$$

Main points:

- Forget the feature map.
- We can directly choose a kernel and work with it!
- The dimension of the feature space does not matter anymore.
- Kernels provide a measure of proximity between $\mathbf{x}$ and $\mathbf{x}'$.

Kernels: Examples

Example 1:

- D-dimensional inputs: $\mathbf{x} = (x_1, x_2, ..., x_D)^\top$ and $\mathbf{z} = (z_1, z_2, ...z_D)^\top$

$$\begin{aligned}
k(\mathbf{x}, \mathbf{z}) =&(\mathbf{x}^\top \mathbf{z})^2 = (x_1 z_1 + x_2 z_2 + ...)^2 \\
=&x_1^2 z_1^2 + 2 x_1 z_1 x_2 z_2 + x_2^2 z_2^2 + ... \\
=&(x_1^2, x_2^2, ..., \sqrt{2} x_1 x_2, ...)^\top (z_1^2, z_2^2, ..., \sqrt{2} z_1 z_2, ...) \\
=&\psi(\mathbf{x})^\top \psi(\mathbf{z})
\end{aligned}$$

Example 2 (Gaussian kernel): $k(\mathbf{x}, \mathbf{z}) = \exp(-\|\mathbf{x} - \mathbf{z}\|^2 / 2\sigma^2)$.

- The feature vector has infinite dimension here! (a bit of functional analysis)

Common Kernels in GPs

- **Squared Exponential (RBF) Kernel:**

$$k_{\mathrm{E}}(t, t') = \sigma^2 \exp\left(-\frac{\|t - t'\|^2}{2\ell^2}\right).$$

  - Controls smoothness of the functions sampled from the GP.
  - Length scale ℓ: Correlation distance.
  - Signal variance σ^2: Scale of the output.

- **Matérn Kernel:**

$$k_{\mathrm{M}}(t, t') = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\frac{\|t - t'\|}{\ell}\right)^{\nu} K_{\nu}\left(\sqrt{2\nu}\frac{\|t - t'\|}{\ell}\right).$$

  - ν: Smoothness parameter.
  - More flexible than the RBF kernel for modeling rough functions.

Constructing kernels from kernels

Given valid kernels $k_1(\mathbf{x}, \mathbf{x}')$ and $k_2(\mathbf{x}, \mathbf{x}')$, the following kernels will also be valid:

$$
\begin{aligned}
k(\mathbf{x}, \mathbf{x}') &= c k_1(\mathbf{x}, \mathbf{x}') \quad \text{for} \;\; c > 0, \\
k(\mathbf{x}, \mathbf{x}') &= f(\mathbf{x}) k_1(\mathbf{x}, \mathbf{x}') f(\mathbf{x}') \\
k(\mathbf{x}, \mathbf{x}') &= k_1(\mathbf{x}, \mathbf{x}') + k_2(\mathbf{x}, \mathbf{x}') \\
k(\mathbf{x}, \mathbf{x}') &= k_1(\mathbf{x}, \mathbf{x}') \cdot k_2(\mathbf{x}, \mathbf{x}') \\
k(\mathbf{x}, \mathbf{x}') &= \mathbf{x}^\top A \mathbf{x}' \qquad (A \text{ PSD}) \\
k(\mathbf{x}, \mathbf{x}') &= \exp(k_1(\mathbf{x}, \mathbf{x}')) \\
k(\mathbf{x}, \mathbf{x}') &= q(k_1(\mathbf{x}, \mathbf{x}'))
\end{aligned}
$$

where q polynomial with ≥ 0 coefficients.

Modelling perspective

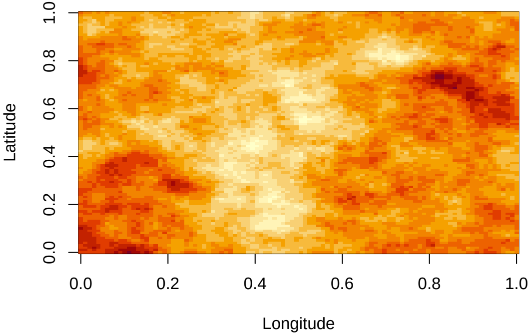
Working with Gaussian Processes we fix a kernel function.
Data: Suppose we observed $(X_{t_1}, \ldots, X_{t_n})$ for some $t_1, \ldots, t_n \in T$.

- If the kernel function comes with some hyperparameters α, we can learn them maximizing the log-likelihood.
  - By definition, $(X_{t_1}, \ldots, X_{t_n})$ is MVN with covariance that depends on α.
  - This may be a complicated optimization procedure.

- Suppose we want to predict the value of the process at t_{n+1}
  - By definition $(X_{t_1}, \ldots, X_{t_n}, X_{t_{n+1}})$ is jointly Gaussian so simply compute the conditional distribution: $X_{t_{n+1}} | X_{t_1}, \ldots, X_{t_n}$

Example: Modeling Spatial Data with GPs

GPs are widely used in spatial statistics, e.g. temperature across a grid of locations.



ullet Grid of 100^2 points.

ullet Fix the exponential kernel $\exp\{-\frac{1}{2}\|\mathbf{x} - \mathbf{x}'\|\}$

ullet Compute the $100^2 \times 100^2$ covariance matrix

ullet Get 1 sample from the corresponding distr.

Handling a 10000-dimensional Gaussian comes with its own computational challenges.

Spatial GP: Prediction

1. Combine training and test locations.
2. Compute the covariance matrix using the kernel function.
3. Use Gaussian conditioning formulas:

$$\mathbb{E}[\mathbf{y}_{\text{test}} | \mathbf{y}_{\text{train}}] = \Sigma_{\text{test,train}}^{\top} \Sigma_{\text{train,train}}^{-1} \mathbf{y}_{\text{train}},$$
$$\text{Cov}(\mathbf{y}_{\text{test}} | \mathbf{y}_{\text{train}}) = \Sigma_{\text{test,test}} - \Sigma_{\text{test,train}}^{\top} \Sigma_{\text{train,train}}^{-1} \Sigma_{\text{test,train}}.$$

Nonparametric Regression

GPs can be used for nonparametric regression:

$$y_i = f(\mathbf{x}_i) + \varepsilon_i, \quad \varepsilon_i \sim N(0, \sigma^2), \quad i = 1, \ldots, n.$$

Prior over $f : \mathbb{R}^d \to \mathbb{R}$: GP defined by $m(\mathbf{x})$ and $k(\mathbf{x}, \mathbf{x}')$.

- In this sense GP defines a distribution over (random) functions $f : \mathbb{R}^d \to \mathbb{R}$.

We have $f(\mathbf{X}) = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)) \sim N_n(\nu, C)$

- $\nu_i = m(\mathbf{x}_i)$
- $C_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$

Say $d = 1$. Given $m(x)$ and $k(x, x')$, how would you plot random samples of the corresponding random functions on $\mathbb{R}$?

Nonparametric Regression

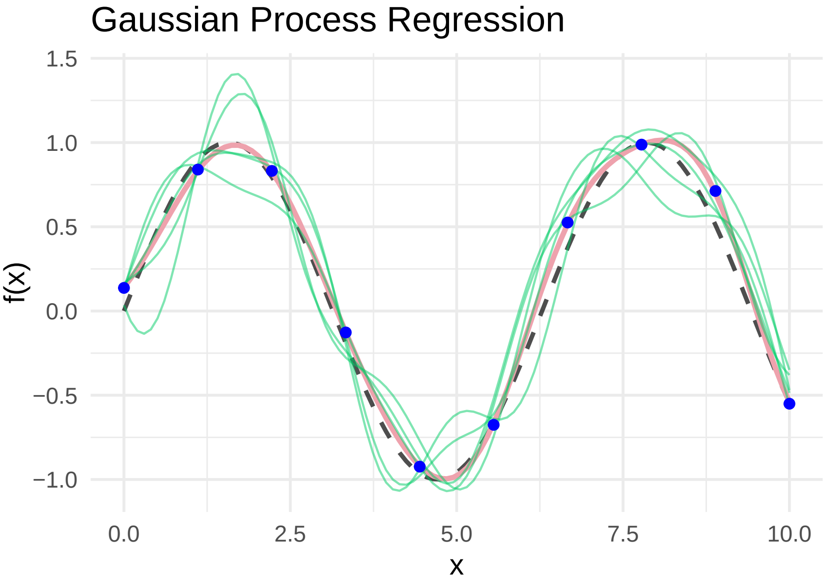
Note that $\mathbf{y} = (y_1, \ldots, y_n) = (f(\mathbf{x}_1) + \varepsilon_1, \ldots, f(\mathbf{x}_n) + \varepsilon_n)$.

The function $y(\mathbf{x})$ is also described by a GP:

- The mean is $m(\mathbf{x})$.
  - $\mathbb{E}[y(\mathbf{x}_i)] = \mathbb{E}[f(\mathbf{x}_i) + \varepsilon_i] = m(\mathbf{x}_i)$.
- The kernel is $k(\mathbf{x}, \mathbf{x}') + \sigma^2 \mathbf{1}\{\mathbf{x} = \mathbf{x}'\}$.
  - $\text{cov}[y(\mathbf{x}_i), y(\mathbf{x}_j)] = \text{cov}(f(\mathbf{x}_i) + \varepsilon_i, f(\mathbf{x}_j) + \varepsilon_j) = k(\mathbf{x}_i, \mathbf{x}_j') + \sigma^2 \mathbf{1}\{\mathbf{x}_i = \mathbf{x}_j'\}$.

Given data $(y_1, \mathbf{x}_1)$, $\ldots$, $(y_n, \mathbf{x}_n)$ we can now easily predict y at any other point $\mathbf{x}$.

Illustration



Summary

- Gaussian Processes are a versatile tool for regression and spatial modeling.
- Key components:
  - Mean function.
  - Kernel function.
- Takeaway: Conceptually it is not harder than MVNs and the same formulas apply.