ĐẠI HỌC QUỐC GIA TPHCM

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN

23TNT1

Mô hình thay thế xác suất

Đề tài: Chương 18 - Mô hình thay thế xác suất

Môn học: Cơ sở Trí tuệ nhân tạo

Sinh viên thực hiện:

23122006 - Lưu Thượng Hồng

23122020 - Nguyễn Thiên Ấn

23122034 - Lê Nguyên Khang

23122036 - Nguyễn Ngọc Khoa

Giáo viên hướng dẫn:

GS.TS. Lê Hoài Bắc

ThS. Lê Nhựt Nam

Ngày 26 tháng 12 năm 2025

Mục lục

1 Mô hình thay thế

Trước khi đi vào nội dung của Chương 18 - Mô hình thay thế xác suất, ta cần biết mô hình thay thế (surrogate model) là gì.

Mô hình thay thế $\hat{f}$ là một hàm xấp xỉ toán học được thiết kế để mô phỏng hành vi của hàm mục tiêu thực f nhưng với đặc tính mịn hơn và chi phí tính toán thấp hơn rất nhiều. Các mô hình này đóng vai trò quan trọng khi việc đánh giá hàm mục tiêu thực tế cực kỳ tốn kém, chẳng hạn như qua các thử nghiệm vật lý, mô phỏng siêu máy tính phức tạp hoặc huấn luyện mạng thần kinh sâu.

Quá trình xây dựng một mô hình thay thế thường tuân theo các bước:

- Lấy mẫu: Sử dụng các kế hoạch lấy mẫu (sampling plans) để thu thập dữ liệu ban đầu từ hàm mục tiêu thực.
- Khớp mô hình (Fitting): Sử dụng các kỹ thuật hồi quy (regression) để điều chỉnh các tham số của mô hình sao cho sai số giữa giá trị dự đoán và giá trị thực tế là nhỏ nhất.
- Sử dụng hàm cơ sở: Các mô hình phổ biến thường là tổ hợp tuyến tính của các hàm cơ sở (basis functions) như đa thức (polynomial), hình sin (sinusoidal) hoặc các hàm hướng tâm (radial basis functions - RBF).
- Lựa chọn mô hình: Để đảm bảo mô hình có khả năng dự đoán tốt trên dữ liệu mới (không bị overfitting), các kỹ thuật như kiểm chuẩn chéo (cross-validation) hoặc bootstrap được sử dụng để ước lượng sai số tổng quát hóa.

2 Phân phối Gaussian[1]

Một phân phối Gaussian (phân phối chuẩn) n-biến được đặc trưng bởi vector kì vọng $\boldsymbol{\mu}$ và ma trận hiệp phương sai $\boldsymbol{\Sigma}$. Mật độ xác suất tại $\mathbf{x}$ được định nghĩa là:

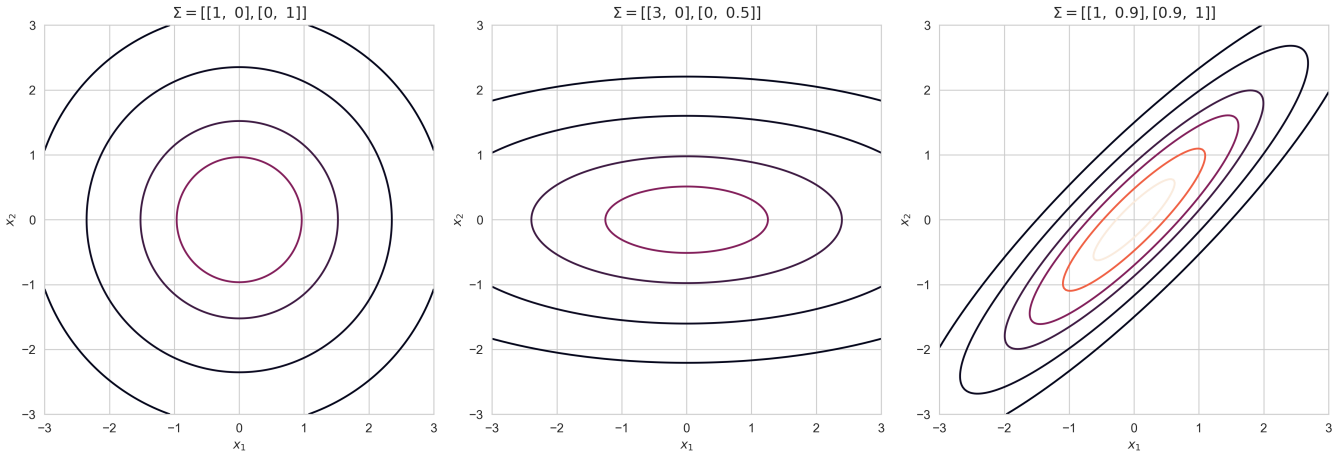
$$\mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{-n/2}|\boldsymbol{\Sigma}|^{-1/2} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

[1]Nội dung các mục trong báo cáo này được trình bày lại dựa trên Chương 18 của cuốn sách *Algorithms for Optimization* Kochenderfer and Wheeler, 2019.

Một vector ngẫu nhiên $\mathbf{x}$ tuân theo phân phối chuẩn được ký hiệu là:

$$\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

Hình 1 minh họa các đường đồng mức của các hàm mật độ xác suất với các ma trận hiệp phương sai khác nhau. Ma trận hiệp phương sai luôn là ma trận nửa xác định dương (positive semidefinite).



Hình 1: Các phân phối Gaussian đa biến với các ma trận hiệp phương sai khác nhau

Xét hai vector ngẫu nhiên Gaussian đồng thời (jointly Gaussian) $\mathbf{a}$ và $\mathbf{b}$, phân phối đồng thời của chúng có dạng:

$$\begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_\mathbf{a} \\ \boldsymbol{\mu}_\mathbf{b} \end{bmatrix}, \begin{bmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C}^\top & \mathbf{B} \end{bmatrix} \right)$$

Trong đó $\mathbf{A}$ và $\mathbf{B}$ là ma trận hiệp phương sai của riêng $\mathbf{a}$ và $\mathbf{b}$, còn $\mathbf{C}$ là ma trận hiệp phương sai chéo.

Phân phối biên (marginal distribution) của từng vector thành phần được xác định bởi:

$$\mathbf{a} \sim \mathcal{N}(\boldsymbol{\mu}_\mathbf{a}, \mathbf{A}), \quad \mathbf{b} \sim \mathcal{N}(\boldsymbol{\mu}_\mathbf{b}, \mathbf{B})$$

Phân phối có điều kiện (conditional distribution) của $\mathbf{a}$ khi biết trước $\mathbf{b}$ cũng là một phân phối Gaussian:

$$\mathbf{a} \mid \mathbf{b} \sim \mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{a}|\mathbf{b}}, \boldsymbol{\Sigma}_{\mathbf{a}|\mathbf{b}}\right)$$

Các tham số của phân phối này được tính theo công thức dạng đóng (closed-form):

- Vector kì vọng có điều kiện:

$$\boldsymbol{\mu}_{\mathbf{a}|\mathbf{b}} = \boldsymbol{\mu}_{\mathbf{a}} + \mathbf{C}\mathbf{B}^{-1}(\mathbf{b} - \boldsymbol{\mu}_{\mathbf{b}})$$

- Ma trận hiệp phương sai có điều kiện:

$$\boldsymbol{\Sigma}_{\mathbf{a}|\mathbf{b}} = \mathbf{A} - \mathbf{C}\mathbf{B}^{-1}\mathbf{C}^{\top}$$

3 Gaussian Process

Xét tập dữ liệu huấn luyện $\mathcal{D} = \{(\mathbf{x}_i, y_i) \mid i = 1, \ldots, n\}$, trong đó $X = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^\top$ là ma trận các vector đầu vào và $\mathbf{y} = [y_1, \ldots, y_n]^\top$ là vector các giá trị mục tiêu (targets) tương ứng.
Một Gaussian Process (GP) được định nghĩa là một phân phối trên các hàm số (distribution over functions). Một GP được xác định hoàn toàn bởi hàm kì vọng (mean function) $m(\mathbf{x})$ và hàm hiệp phương sai (covariance function/kernel) $k(\mathbf{x}, \mathbf{x}')$:

$$\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m(\mathbf{x}_1) \\ \vdots \\ m(\mathbf{x}_n) \end{bmatrix}, \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \cdots & k(\mathbf{x}_1, \mathbf{x}_n) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_n, \mathbf{x}_1) & \cdots & k(\mathbf{x}_n, \mathbf{x}_n) \end{bmatrix} \right)$$

hay viết gọn hơn là:

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

Trong đó:

- Hàm kì vọng $m(\mathbf{x})$ là giá trị trung bình của hàm số tại điểm đầu vào $\mathbf{x}$:

$$m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})]$$

  Hàm kì vọng có thể thể hiện kiến thức tiên nghiệm (prior knowledge) về hàm số, thường được giả định là hàm không (zero function) trong nhiều ứng dụng.

- Hàm hiệp phương sai $k(\mathbf{x}, \mathbf{x}')$ biểu diễn mối quan hệ giữa các giá trị hàm số tại hai điểm đầu vào khác nhau:

$$k(\mathbf{x}, \mathbf{x}') = \mathbb{E}\left[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))\right]$$

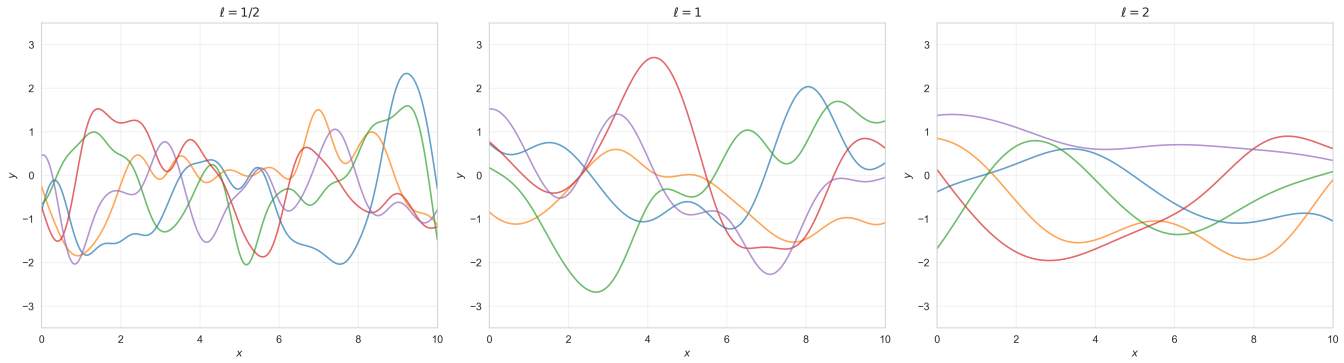
Hàm hiệp phương sai quyết định độ mượt (smoothness) và cấu trúc của các hàm số được mô hình hóa bởi GP.

Hàm hiệp phương sai

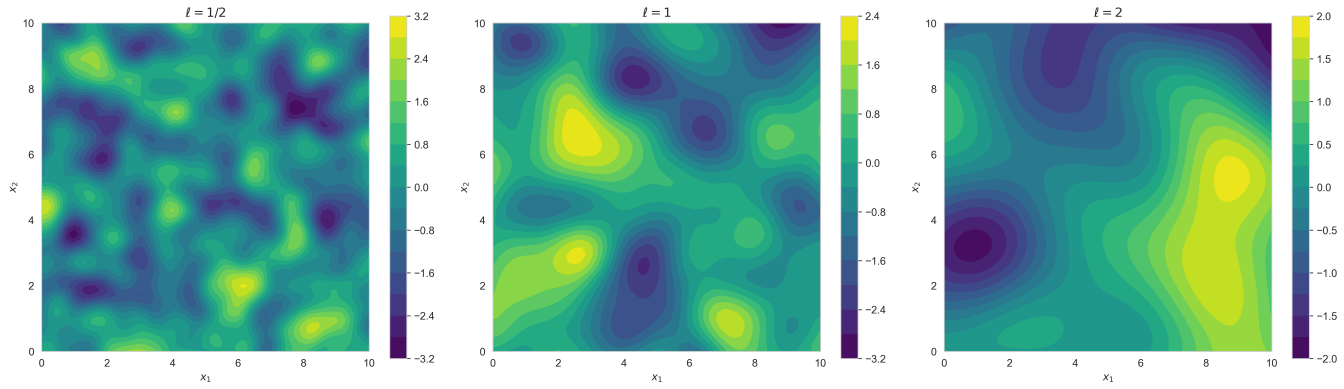
Một hàm hiệp phương sai thường dùng là hàm mũ bình phương (squared exponential kernel):

$$k\left(\mathbf{x}, \mathbf{x}'\right) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\ell^2}\right)$$

Trong đó, ℓ là chiều dài đặc trưng (length-scale) điều khiển mức độ ảnh hưởng của các điểm dữ liệu lân cận đến giá trị hàm số tại một điểm cụ thể. ℓ càng nhỏ thì hàm số càng biến động nhanh, trong khi ℓ càng lớn thì hàm số càng mượt (xem Hình 2 và Hình 3).

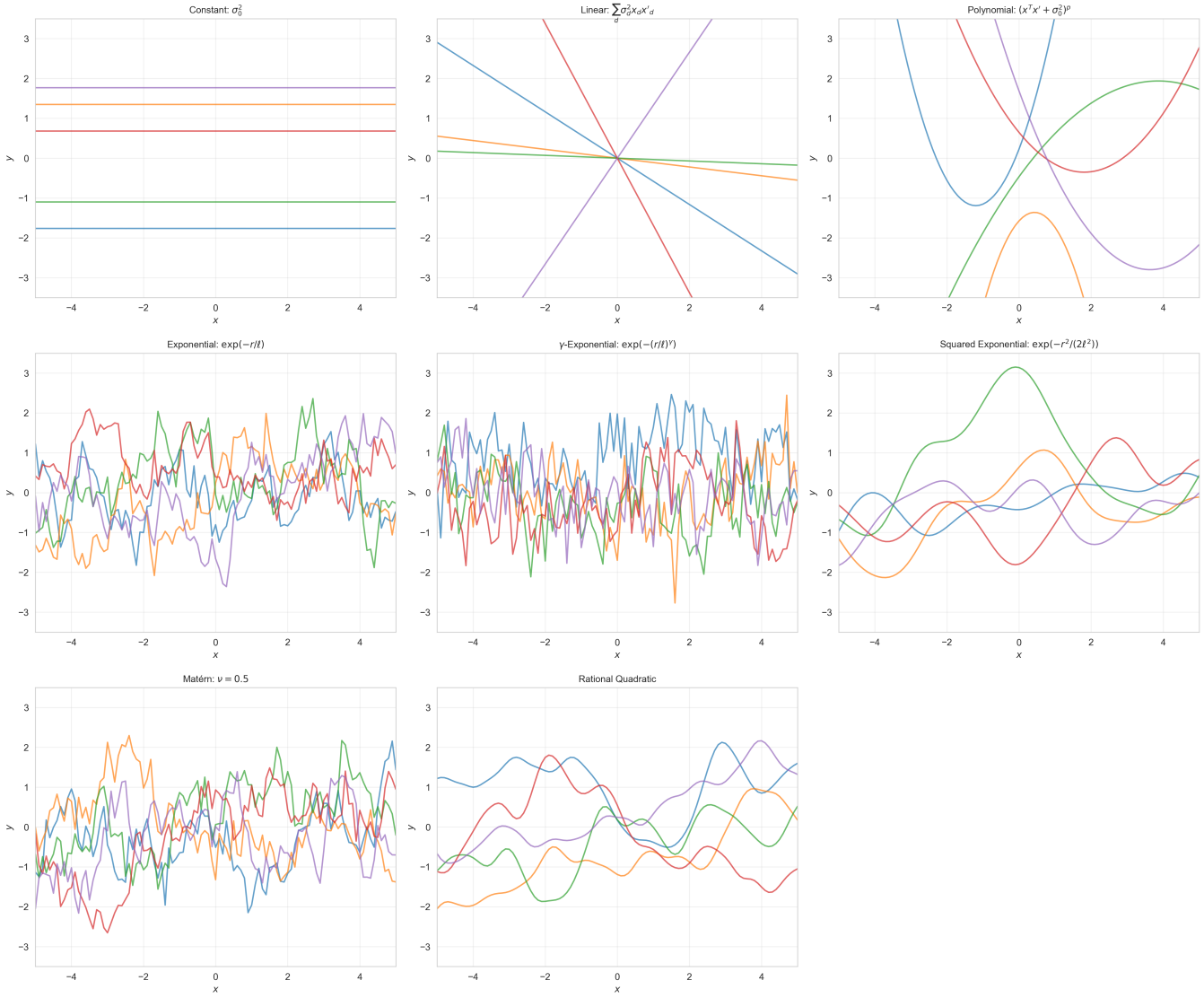


Hình 2: Các mẫu hàm số (1 chiều) được sinh từ Gaussian Process với các giá trị chiều dài đặc trưng ℓ khác nhau



Hình 3: Các mẫu hàm số (2 chiều) được sinh từ Gaussian Process với các giá trị chiều dài đặc trưng ℓ khác nhau

Ngoài ra, còn có nhiều hàm hiệp phương sai khác như hàm tuyến tính (linear kernel), hàm Matern (Matern kernel), và hàm thừa số (periodic kernel), mỗi hàm có các đặc tính riêng phù hợp với các loại dữ liệu và ứng dụng khác nhau. Hình 4 minh họa các mẫu hàm số được sinh từ GP với các hàm hiệp phương sai khác nhau.



Hình 4: Các mẫu hàm số được sinh từ Gaussian Process với các hàm hiệp phương sai khác nhau

4 Dự đoán với Gaussian Process

Một mô hình GP cho bài toán hồi quy thường bao gồm thành phần nhiễu (noise) trong quá trình quan sát. Cụ thể, mô hình được định nghĩa bởi các thành phần sau:

1. Hàm kỳ vọng (mean function) $m(\mathbf{x})$: Xu hướng chung của dữ liệu (thường giả định $m(\mathbf{x}) = 0$ để đơn giản hóa).
2. Hàm hiệp phương sai (covariance function/kernel) $k(\mathbf{x}, \mathbf{x}')$: Quy định độ trơn và hình dáng của các hàm số.
3. Dữ liệu huấn luyện $\mathcal{D} = (X, \mathbf{y})$: Các cặp input-output đã quan sát được.
4. Phương sai nhiễu ν: Độ lớn của nhiễu quan sát. (sẽ được trình bày trong phần sau)

4.1 Phân phối đồng thời

Giả sử ta muốn dự đoán giá trị đầu ra $\hat{\mathbf{y}}$ tại tập điểm X^*. Phân phối đồng thời (joint distribution) giữa dữ liệu quan sát $\mathbf{y}$ và giá trị dự đoán $\hat{\mathbf{y}}$ được mô hình hóa như sau:

$$\begin{bmatrix} \hat{\mathbf{y}} \\ \mathbf{y} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{m}(X^*) \\ \mathbf{m}(X) \end{bmatrix}, \begin{bmatrix} \mathbf{K}(X^*, X^*) & \mathbf{K}(X^*, X) \\ \mathbf{K}(X, X^*) & \mathbf{K}(X, X) \end{bmatrix} \right)$$

Trong đó:

- $\mathbf{m}(X)$ và $\mathbf{m}(X^*)$ là các vector kỳ vọng tại các điểm trong tập huấn luyện và tập dự đoán.
- $\mathbf{K}(X, X)$, $\mathbf{K}(X^*, X)$, $\mathbf{K}(X, X^*)$, và $\mathbf{K}(X^*, X^*)$ là các ma trận hiệp phương sai được xây dựng từ hàm hiệp phương sai $k(\mathbf{x}, \mathbf{x}')$.

Cụ thể, các hàm trên được tính như sau:

$$\mathbf{m}(X) = [m(\mathbf{x}_1), \ldots, m(\mathbf{x}_n)]$$

$$\mathbf{K}(X, X') = \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}'_1) & \cdots & k\left(\mathbf{x}_1, \mathbf{x}'_{|X'|}\right) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_n, \mathbf{x}'_1) & \cdots & k\left(\mathbf{x}_n, \mathbf{x}'_{|X'|}\right) \end{bmatrix}$$

4.2 Phân phối hậu nghiệm

Phân phối dự đoán của $\hat{\mathbf{y}}$ khi biết dữ liệu quan sát $\mathbf{y}$:

$$\hat{\mathbf{y}} \mid \mathbf{y} \sim \mathcal{N}(\underbrace{\mathbf{m}\left(X^*\right) + \mathbf{K}\left(X^*, X\right)\mathbf{K}(X,X)^{-1}(\mathbf{y} - \mathbf{m}(X))}_{\text{kì vọng}}, \underbrace{\mathbf{K}\left(X^*, X^*\right) - \mathbf{K}\left(X^*, X\right)\mathbf{K}(X,X)^{-1}\mathbf{K}\left(X, X^*\right)}_{\text{hiệp phương sai}})$$

Phân phối trên được gọi là phân phối hậu nghiệm (posterior distribution) của mô hình GP.
Khi dự đoán tại một điểm mới $\mathbf{x}$, ta có thể quan tâm đến các đại lượng sau:

- Kì vọng dự đoán: Đây là giá trị trung bình của hàm tại điểm $\mathbf{x}$, đại diện cho ước lượng tốt nhất của mô hình:

$$\begin{aligned}\hat{\boldsymbol{\mu}}(\mathbf{x}) &= m(\mathbf{x}) + \mathbf{K}(\mathbf{x}, X)\mathbf{K}(X,X)^{-1}(\mathbf{y} - \mathbf{m}(X)) \\ &= m(\mathbf{x}) + \boldsymbol{\Theta}^\top \mathbf{K}(X, \mathbf{x})\end{aligned}$$

  Trong đó, vector trọng số $\Theta = \mathbf{K}(X,X)^{-1}(\mathbf{y} - \mathbf{m}(X))$ có thể được tính toán trước để tái sử dụng, giúp tăng tốc độ dự đoán cho các giá trị $\mathbf{x}$ khác nhau.

- Phương sai dự đoán: Thước đo độ bất định (uncertainty) của dự đoán. Giá trị này phụ thuộc hoàn toàn vào hàm hiệp phương sai và vị trí của các điểm dữ liệu $\mathbf{x}$, không phụ thuộc vào giá trị quan sát $\mathbf{y}$:

$$\hat{v}(\mathbf{x}) = k(\mathbf{x}, \mathbf{x}) - \mathbf{K}(\mathbf{x}, X)[\mathbf{K}(X,X) + \nu\mathbf{I}]^{-1}\mathbf{K}(X, \mathbf{x})$$

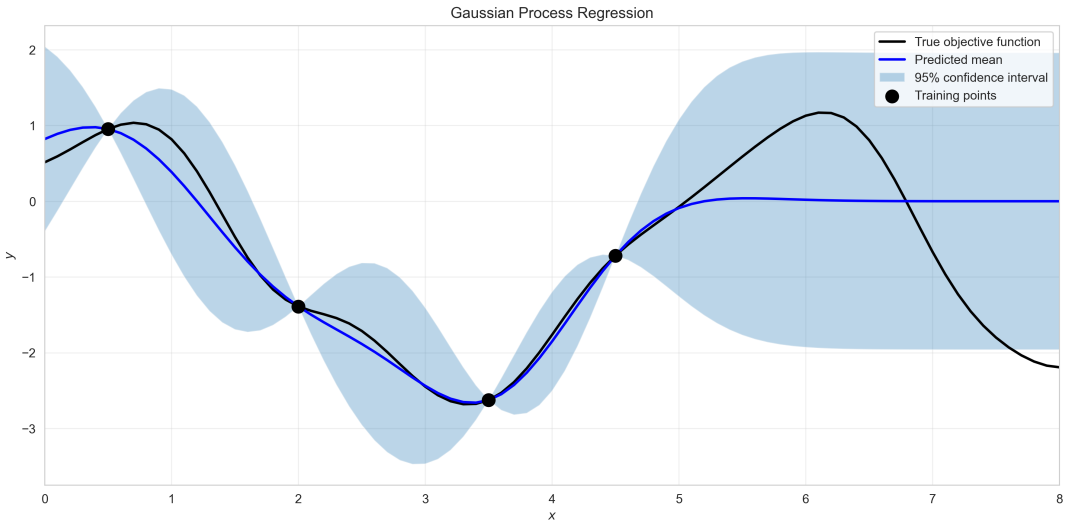
- Độ lệch chuẩn và Khoảng tin cậy: GP sử dụng độ lệch chuẩn dự đoán $\hat{\sigma}(\mathbf{x}) = \sqrt{\hat{v}(\mathbf{x})}$ để xây dựng các khoảng tin cậy (confidence regions).

  Khoảng tin cậy là một phạm vi giá trị được xây dựng xung quanh dự đoán, thể hiện mức độ chắc chắn của mô hình. Ví dụ: Thay vì đưa ra một giá trị đơn lẻ, mô hình chỉ ra rằng với xác suất 95%, giá trị thực sẽ nằm trong khoảng $[y_{\min}, y_{\max}]$.

  Độ lệch chuẩn có cùng đơn vị với kì vọng. Từ độ lệch chuẩn, ta có thể tính toán khoảng tin cậy 95% (tức khoảng giá trị chứa 95% hàm phân phối tích lũy). Với một điểm $\mathbf{x}$ cho trước, khoảng này được xác định bởi:

$$\hat{\mu}(\mathbf{x}) \pm 1.96\hat{\sigma}(\mathbf{x})$$

  Độ rộng của khoảng tin cậy tỉ lệ thuận với độ lệch chuẩn $\hat{\sigma}(\mathbf{x})$.



Hình 5: Dự đoán với Gaussian Process

Quan sát hình 5, ta có thể thấy cách một mô hình GP điều chỉnh kì vọng và khoảng tin cậy dựa trên dữ liệu quan sát. Các điểm màu đen biểu diễn dữ liệu huấn luyện, đường màu xanh lam là kì vọng dự đoán, và vùng bóng xung quanh thể hiện khoảng tin cậy 95%. Khi gần các điểm dữ liệu, mô hình trở nên chắc chắn hơn (khoảng tin cậy hẹp lại), trong khi ở những vùng xa dữ liệu, độ bất định tăng lên (khoảng tin cậy rộng hơn).

5 Kết hợp thông tin gradient

Gaussian Process có thể được mở rộng để không chỉ học từ giá trị của hàm số mà còn học từ gradient của nó tại các điểm dữ liệu. Việc này giúp mô hình nắm bắt tốt hơn xu hướng thay đổi của hàm mục tiêu, đặc biệt hữu ích trong các không gian ít dữ liệu.

5.1 Mô hình Gaussian Process với gradient

Khi có thêm thông tin về gradient $\nabla \mathbf{y}$ tại các điểm dữ liệu $\mathbf{x}_i, (i = \overline{1,n})$, ta có thể mở rộng mô hình GP để bao gồm cả giá trị hàm và gradient:

$$\begin{bmatrix} \mathbf{y} \\ \nabla \mathbf{y} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{m}_f \\ \mathbf{m}_\nabla \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{ff} & \mathbf{K}_{f\nabla} \\ \mathbf{K}_{\nabla f} & \mathbf{K}_{\nabla\nabla} \end{bmatrix} \right)$$

Trong đó:

- $\mathbf{m}_\nabla$ là hàm kỳ vọng của gradient.
- $\mathbf{K}_{f\nabla}$ và $\mathbf{K}_{\nabla f}$ là các ma trận hiệp phương sai chéo giữa giá trị hàm và gradient.
- $\mathbf{K}_{\nabla\nabla}$ là ma trận hiệp phương sai giữa các gradient.

5.2 Tính toán ma trận hiệp phương sai cho gradient

Dựa vào tính chất đạo hàm là một phép toán tuyến tính, ta có thể suy ra các hàm hiệp phương sai liên quan đến gradient từ hàm hiệp phương sai ban đầu $k(\mathbf{x}, \mathbf{x}')$:

$$
\begin{aligned}
k_{ff}\left(\mathbf{x}, \mathbf{x}'\right) &= k\left(\mathbf{x}, \mathbf{x}'\right) \\
k_{\nabla f}\left(\mathbf{x}, \mathbf{x}'\right) &= \nabla_{\mathbf{x}} k\left(\mathbf{x}, \mathbf{x}'\right) \\
k_{f\nabla}\left(\mathbf{x}, \mathbf{x}'\right) &= \nabla_{\mathbf{x}'} k\left(\mathbf{x}, \mathbf{x}'\right) \\
k_{\nabla\nabla}\left(\mathbf{x}, \mathbf{x}'\right) &= \nabla_{\mathbf{x}} \nabla_{\mathbf{x}'} k\left(\mathbf{x}, \mathbf{x}'\right)
\end{aligned}
$$

5.3 Phân phối đồng thời với gradient

Để dự đoán tại tập điểm mới X^*, ta thiết lập phân phối đồng thời giữa giá trị dự đoán $\hat{\mathbf{y}}$ và toàn bộ dữ liệu quan sát (bao gồm cả $\mathbf{y}$ và $\nabla\mathbf{y}$):

$$
\begin{bmatrix} \hat{\mathbf{y}} \\ \mathbf{y} \\ \nabla\mathbf{y} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{m}_f(X^*) \\ \mathbf{m}_f(X) \\ \mathbf{m}_{\nabla f}(X) \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{ff}(X^*, X^*) & \mathbf{K}_{ff}(X^*, X) & \mathbf{K}_{f\nabla}(X^*, X) \\ \mathbf{K}_{ff}(X, X^*) & \mathbf{K}_{ff}(X, X) & \mathbf{K}_{f\nabla}(X, X) \\ \mathbf{K}_{\nabla f}(X, X^*) & \mathbf{K}_{\nabla f}(X, X) & \mathbf{K}_{\nabla\nabla}(X, X) \end{bmatrix} \right)
$$

5.4 Phân phối hậu nghiệm với gradient

Ta có thể suy ra phân phối hậu nghiệm cho $\hat{\mathbf{y}}$ dựa trên toàn bộ dữ liệu quan sát.

$$
\hat{\mathbf{y}} \mid \mathbf{y}, \nabla\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}_\nabla, \Sigma_\nabla)
$$

Trong đó:

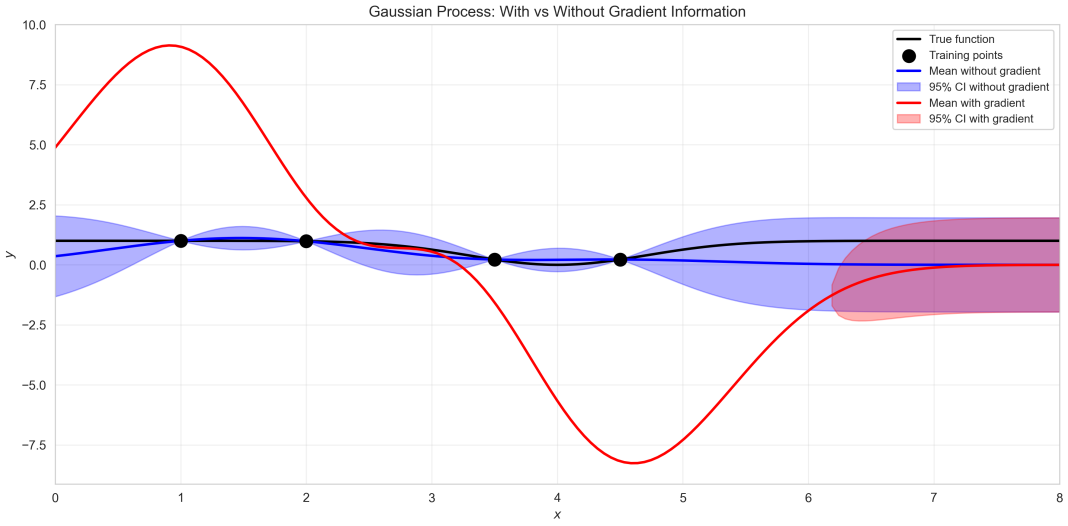
- Kỳ vọng hậu nghiệm:

$$\boldsymbol{\mu}_{\nabla} = \mathbf{m}(X^*) + \mathbf{K}(X^*, X)(\mathbf{K}(X, X) + \nu\mathbf{I})^{-1}(\mathbf{y} - \mathbf{m}(X))$$

- Hiệp phương sai hậu nghiệm:

$$\boldsymbol{\Sigma}_{\nabla} = \mathbf{K}(X^*, X^*) - \mathbf{K}(X^*, X)(\mathbf{K}(X, X) + \nu\mathbf{I})^{-1}\mathbf{K}(X, X^*)$$

Hình 6 minh họa kết quả dự đoán sử dụng Gaussian Process khi có thông tin về gradient. Ta có thể thấy rằng việc bổ sung thông tin gradient giúp mô hình GP nắm bắt tốt hơn các biến đổi của hàm mục tiêu.



Hình 6: Dự đoán với Gaussian Process có thông tin gradient

6 Kết hợp thông tin nhiễu

6.1 Mô hình Gaussian Process với nhiễu

Trong các ứng dụng thực tế, các quan sát từ hàm mục tiêu f thường không chính xác tuyệt đối mà bị ảnh hưởng bởi nhiễu. Ta mô hình hóa quá trình này thông qua công thức:

$$y = f(\mathbf{x}) + z$$

Trong đó, $f(\mathbf{x})$ là giá trị tất định của hàm số, và z là nhiễu Gaussian độc lập có kỳ vọng bằng 0 (zero-mean Gaussian noise), tức $z \sim \mathcal{N}(0, \nu)$. Tham số phương sai nhiễu ν đóng vai trò quan trọng trong việc kiểm soát độ "mượt" của mô hình và tránh hiện tượng overfitting.

6.2 Phân phối đồng thời với nhiễu

Khi bao gồm nhiễu trong mô hình GP, phân phối đồng thời giữa dữ liệu quan sát $\mathbf{y}$ và giá trị dự đoán $\hat{\mathbf{y}}$ được điều chỉnh như sau:

$$\begin{bmatrix} \hat{\mathbf{y}} \\ \mathbf{y} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{m}\left(X^*\right) \\ \mathbf{m}(X) \end{bmatrix}, \begin{bmatrix} \mathbf{K}\left(X^*, X^*\right) & \mathbf{K}\left(X^*, X\right) \\ \mathbf{K}\left(X, X^*\right) & \mathbf{K}(X, X) + \nu\mathbf{I} \end{bmatrix} \right)$$

Để ý rằng công thức trên giống với phân phối đồng thời ban đầu 4.1, nhưng ma trận hiệp phương sai của dữ liệu quan sát $\mathbf{y}$ được cộng thêm một thành phần $\nu\mathbf{I}$ để phản ánh sự hiện diện của nhiễu Gaussian độc lập với phương sai ν.

6.3 Phân phối hậu nghiệm với nhiễu

Phân phối hậu nghiệm cho $\hat{\mathbf{y}}$ khi biết dữ liệu quan sát $\mathbf{y}$ trở thành:

$$\hat{\mathbf{y}} \mid \mathbf{y}, \nu \sim \mathcal{N}(\boldsymbol{\mu}^*, \Sigma^*)$$

Trong đó:

- Kì vọng hậu nghiệm:

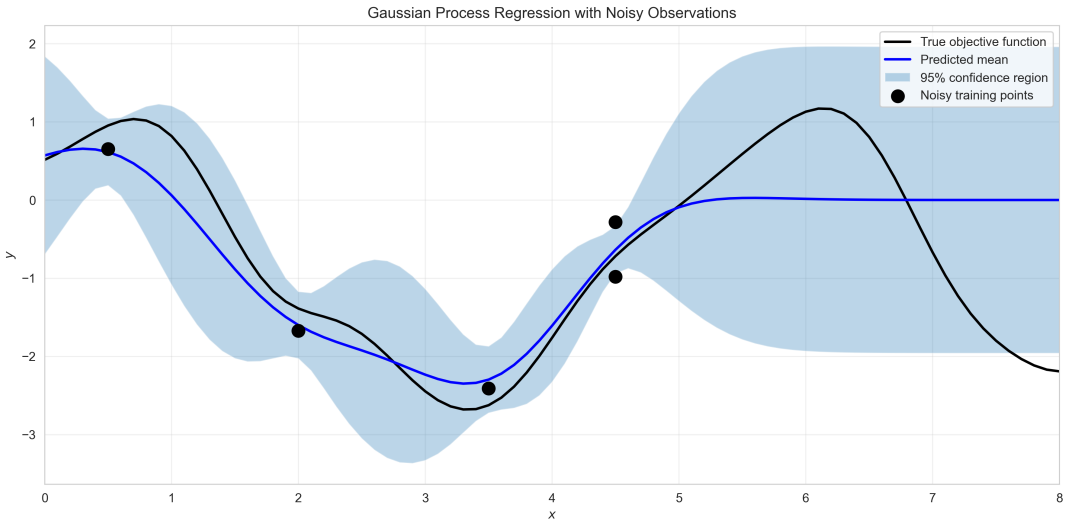
$$\boldsymbol{\mu}^* = \mathbf{m}(X^*) + \mathbf{K}(X^*, X)(\mathbf{K}(X, X) + \nu\mathbf{I})^{-1}(\mathbf{y} - \mathbf{m}(X))$$

- Hiệp phương sai hậu nghiệm:

$$\boldsymbol{\Sigma}^* = \mathbf{K}(X^*, X^*) - \mathbf{K}(X^*, X)(\mathbf{K}(X, X) + \nu\mathbf{I})^{-1}\mathbf{K}(X, X^*)$$

Hình 7 minh họa kết quả dự đoán sử dụng Gaussian Process khi dữ liệu quan sát bị nhiễu. Ta có thể thấy rằng mô hình GP vẫn có khả năng nắm bắt xu hướng chung của hàm mục tiêu mặc dù

dữ liệu bị ảnh hưởng bởi nhiễu.



Hình 7: Dự đoán với Gaussian Process có thông tin nhiễu

7 Huấn luyện Gaussian Process

Hiệu suất của mô hình GP phụ thuộc rất lớn vào việc lựa chọn hàm hiệp phương sai và các tham số đi kèm (bao gồm các siêu tham số của hàm hiệp phương sai Θ và phương sai nhiễu ν).
Thông thường, các tham số này có thể được lựa chọn bằng phương pháp cross-validation. Tuy nhiên, đối với GP, phương pháp phổ biến và hiệu quả hơn là tối đa hóa hàm log hợp lý (log likelihood). Thay vì tối thiểu hóa sai số bình phương (squared error), ta tìm tập tham số Θ và ν sao cho xác suất quan sát được dữ liệu $p(\mathbf{y} \mid X, \Theta, \nu)$ là lớn nhất.

7.1 Hàm log hợp lý

Cho tập dữ liệu $\mathcal{D}$ gồm n phần tử quan sát. Đặt $\mathbf{\Sigma}_\Theta = \mathbf{K}_\Theta(X, X) + \nu\mathbf{I}$ là ma trận hiệp phương sai của dữ liệu có nhiễu. Log likelihood được xác định bởi công thức:

$$\log p(\mathbf{y} \mid X, \nu, \Theta) = \underbrace{-\frac{1}{2}(\mathbf{y} - \mathbf{m}_\Theta(X))^\top \mathbf{\Sigma}_\Theta^{-1}(\mathbf{y} - \mathbf{m}_\Theta(X))}_{\text{Độ khớp dữ liệu (Data fit)}} \quad \underbrace{-\frac{1}{2}\log|\mathbf{\Sigma}_\Theta|}_{\text{Độ phức tạp (Complexity penalty)}} \quad -\frac{n}{2}\log 2\pi$$

Trong đó:

- Thành phần đầu tiên đo lường mức độ phù hợp của mô hình với dữ liệu (Data fit).
- Thành phần thứ hai đóng vai trò như một hình thức kiểm soát độ phức tạp (Complexity penalty), ngăn mô hình rơi vào overfitting.

7.2 Tối ưu hóa tham số

Giả sử ta sử dụng hàm kỳ vọng bằng 0 ($\mathbf{m}_\Theta(X) = \mathbf{0}$), khi đó Θ chỉ còn ảnh hưởng đến hàm hiệp phương sai. Các siêu tham số (hyperparameters) của GP có thể được ước lượng bằng phương pháp Ước lượng hợp lý cực đại (Maximum Likelihood Estimation - MLE) thông qua thuật toán Gradient Ascent.

Đạo hàm riêng của log likelihood theo một tham số Θ_j được tính như sau:

$$\frac{\partial}{\partial \Theta_j} \log p(\mathbf{y} \mid X, \Theta) = \frac{1}{2}\mathbf{y}^\top \mathbf{K}^{-1} \frac{\partial \mathbf{K}}{\partial \Theta_j} \mathbf{K}^{-1}\mathbf{y} - \frac{1}{2} \operatorname{tr}\left(\boldsymbol{\Sigma}_\Theta^{-1} \frac{\partial \mathbf{K}}{\partial \Theta_j}\right)$$

Việc tính toán gradient này cho phép ta cập nhật các siêu tham số Θ (như length-scale ℓ, signal variance σ_f^2) và phương sai nhiễu ν để mô hình GP phù hợp nhất với dữ liệu huấn luyện.

8 Tổng kết

Gaussian Process (GP) được định nghĩa là một phân phối trên không gian các hàm số, trong đó việc lựa chọn hàm nhân (kernel) đóng vai trò quyết định đến độ trơn và hình dáng của các hàm được mô hình hóa. Về mặt toán học, GP dựa trên nền tảng của phân phối chuẩn đa biến, tận dụng các tính chất của phân phối biên và phân phối có điều kiện để thực hiện suy diễn.

Khi có một tập dữ liệu huấn luyện cho trước, mô hình cho phép tính toán kỳ vọng và độ lệch chuẩn để đưa ra dự đoán về hàm mục tiêu tại các điểm dữ liệu mới. Bên cạnh đó, GP có khả năng mở rộng linh hoạt để kết hợp thông tin đạo hàm nhằm cải thiện độ chính xác, cũng như tích hợp ước lượng nhiễu để xử lý dữ liệu thực tế một cách hiệu quả. Cuối cùng, các tham số của mô hình thường được ước lượng bằng phương pháp Ước lượng hợp lý cực đại (MLE) thông qua việc tối đa hóa hàm log likelihood.

Tài liệu

Kochenderfer, M. J., & Wheeler, T. A. (2019). *Algorithms for optimization*. The MIT Press.