

Gaussian Processes Part 1

Max Kochurov

MSU

2022

Agenda

① Intro

- Preliminaries
- Kernel Math
- Basic Hyper-Parameters
- Kernel Types
- Kernel Math

② Example

- Spatial Hierarchy

Non-parametrics

- Assumptions are vague
- Structure (of a function) is your prior.
- Is not only about Gaussian Processes

Non-parametrics

- Assumptions are vague
 - Priors on functions
 - Priors on time or spatial effects
 - Structure (of a function) is your prior.
-
- Is not only about Gaussian Processes

Non-parametrics

- Assumptions are vague
 - Priors on functions
 - Priors on time or spatial effects
- Structure (of a function) is your prior.
 - Does not change much
 - Volatile
 - Can take values from y_0 to y_1
 - Extrapolates periodically
 - And more structural assumptions
- Is not only about Gaussian Processes

Non-parametrics

- Assumptions are vague
 - Priors on functions
 - Priors on time or spatial effects
- Structure (of a function) is your prior.
 - Does not change much
 - Volatile
 - Can take values from y_0 to y_1
 - Extrapolates periodically
 - And more structural assumptions
- Is not only about Gaussian Processes
 - Dirichlet Processes
 - Bayesian Additive Regression Trees
 - Many Others

Notation

$$x \in \mathbb{R}^n, y \in \mathbb{R}$$

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

Notation

$$x \in \mathbb{R}^n, y \in \mathbb{R}$$

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

$$\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m(x_1) \\ \vdots \\ m(x_N) \end{bmatrix}, \begin{bmatrix} k(x_1, x_1) & \dots & k(x_1, x_N) \\ \vdots & \ddots & \vdots \\ k(x_N, x_1) & \dots & k(x_N, x_N) \end{bmatrix} \right)$$

- ① \mathcal{GP} Gaussian Process - simply, a normal distribution with special mean $m(x)$ and covariance $k(x, x')$

Notation

$$x \in \mathbb{R}^n, y \in \mathbb{R}$$

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

$$\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m(x_1) \\ \vdots \\ m(x_N) \end{bmatrix}, \begin{bmatrix} k(x_1, x_1) & \dots & k(x_1, x_N) \\ \vdots & \ddots & \vdots \\ k(x_N, x_1) & \dots & k(x_N, x_N) \end{bmatrix} \right)$$

- ① \mathcal{GP} Gaussian Process - simply, a normal distribution with special mean $m(x)$ and covariance $k(x, x')$
- ② $m(x)$ - mean function, e.g.
 - Linear regression $m(x) = x^\top \beta$
 - Simply Constant or Zero $m(x) = c$
 - Other custom functions $m(x) = \sin(x)$

Notation

$$x \in \mathbb{R}^n, y \in \mathbb{R}$$

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

$$\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m(x_1) \\ \vdots \\ m(x_N) \end{bmatrix}, \begin{bmatrix} k(x_1, x_1) & \dots & k(x_1, x_N) \\ \vdots & \ddots & \vdots \\ k(x_N, x_1) & \dots & k(x_N, x_N) \end{bmatrix} \right)$$

- ① \mathcal{GP} Gaussian Process - simply, a normal distribution with special mean $m(x)$ and covariance $k(x, x')$
- ② $m(x)$ - mean function, e.g.
 - Linear regression $m(x) = x^\top \beta$
 - Simply Constant or Zero $m(x) = c$
 - Other custom functions $m(x) = \sin(x)$
- ③ $k(x, x')$ - kernel function, simply - measure of similarity for x and x'
 - $[K]_{ij} = k(x_i, x_j)$ is an SPD matrix

Kernel Function

Recall, $\mathcal{GP}(M(x), K(x, x'))$ is a kind of normal distribution. This how a kernel might look like:

$$\begin{aligned}k(x, x') &= RBF(x, x') \\ &= \exp(\|x - x'\|/2L)\end{aligned}$$

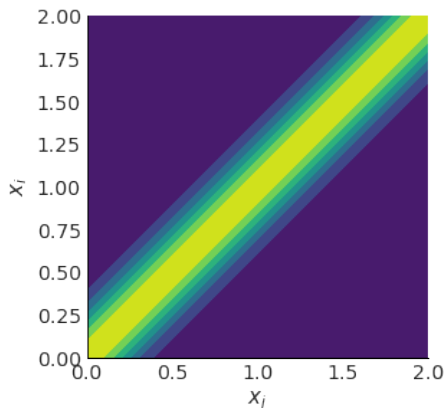


Figure: RBF kernel (data space)

Kernel Function

Recall, $\mathcal{GP}(M(x), K(x, x'))$ is a kind of normal distribution. This how a kernel might look like:

$$\begin{aligned}k(x, x') &= RBF(x, x') \\ &= \exp(\|x - x'\|/2L)\end{aligned}$$

Parameter Interpretation

L - lengthscale for x such that y does not change much

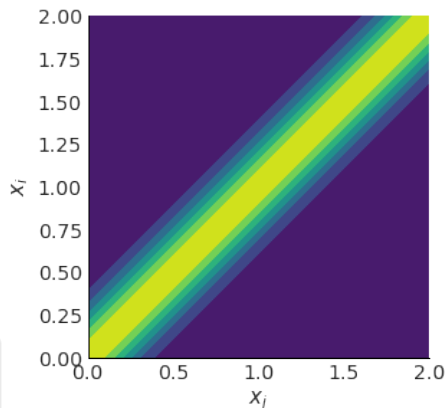


Figure: RBF kernel (data space)

Kernel Function

Recall, $\mathcal{GP}(M(x), K(x, x'))$ is a kind of normal distribution. This how a kernel might look like:

$$\begin{aligned}k(x, x') &= RBF(x, x') \\ &= \exp(\|x - x'\|/2L)\end{aligned}$$

Parameter Interpretation

L - lengthscale for x such that y does not change much

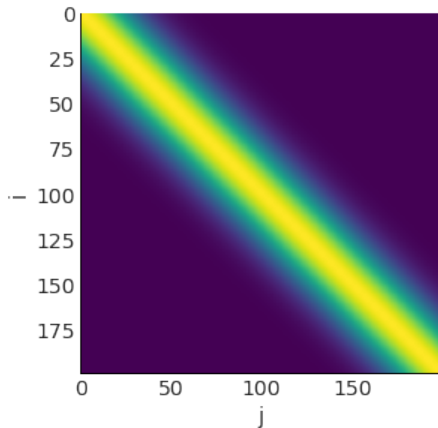


Figure: RBF kernel (covariance matrix)

Kernel Math

Kernels can be combined (read more [2]). If $k_1(x, x')$ and $k_2(x, x')$ are valid kernels, then

- ① $k_*(x, x') = a \cdot k_1(x, x') + b \cdot k_2(x, x')$ is a valid kernel
 - sum rule
 - $a, b > 0$
- ② $k_*(x, x') = k_1(x, x')^a \cdot k_2(x, x')^b$ is a valid kernel
 - product rule
 - $a, b > 0$

Kernel Math

Kernels can be combined (read more [2]). If $k_1(x, x')$ and $k_2(x, x')$ are valid kernels, then

- ① $k_*(x, x') = a \cdot k_1(x, x') + b \cdot k_2(x, x')$ is a valid kernel
 - sum rule
 - $a, b > 0$
- ② $k_*(x, x') = k_1(x, x')^a \cdot k_2(x, x')^b$ is a valid kernel
 - product rule
 - $a, b > 0$

Basic parametrisation often includes the following

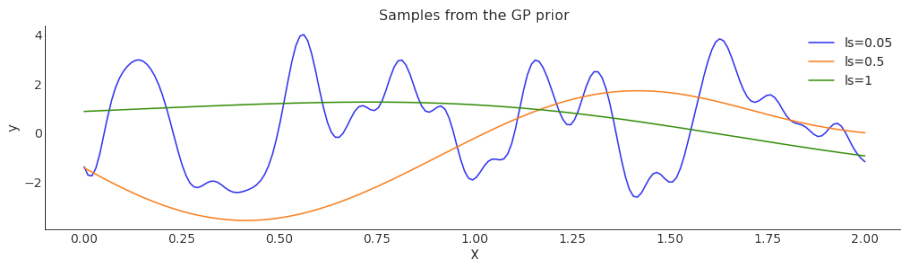
- White Noise ε
- Amplitude σ
- Lengthscale L

$$k(x, x') \cdot \sigma + \varepsilon$$

Understanding the lengthscale

- How **quickly** y is changed
- Not the magnitude!
- Often known up to a good value
- Hard to infer in practice

$$k(x, x') \cdot \sigma + \varepsilon$$



Educated guess on lenthcales

- **Granularity** of time series data
 - If data is yearly, 1y lenthscale is a good fit
 - Interpolate missing observations
 - Interpolate higher granularity (months)

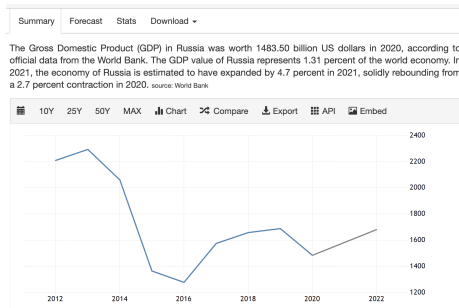


Figure: Russian GDP
(tradingeconomics.com)

Educated guess on lenthscases

- **Granularity** of time series data
 - If data is yearly, 1y lenthscale is a good fit
 - Interpolate missing observations
 - Interpolate higher granularity (months)
- **Other**
 - Spatial distance (km, m, cm)
 - Age
 - Education duration

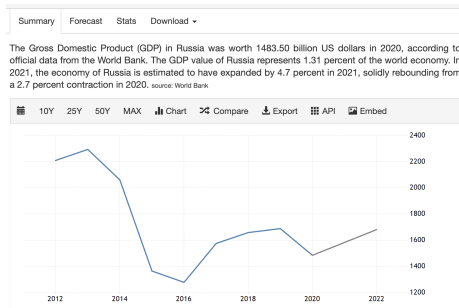


Figure: Russian GDP
(tradingeconomics.com)

Understanding Amplitude

$$k(x, x') \cdot \sigma + \varepsilon$$

- How variable are the outcomes
- Not the standard deviation (aka white noise)
- Prior can be set with prior predictive checks

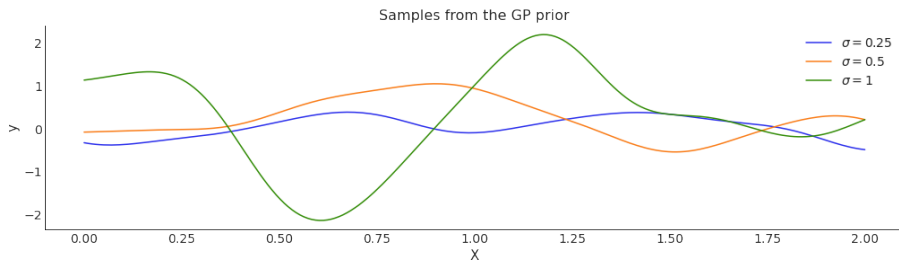


Figure: Amplitude (σ) comparison

Amplitude vs White Noise

$$k(x, x') \cdot \sigma + \epsilon$$

- White Noise is separate thing from amplitude

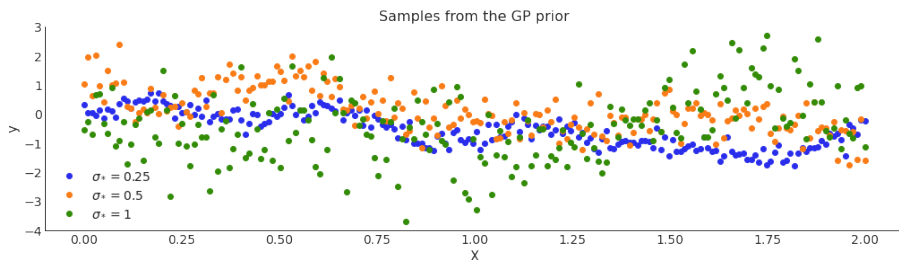


Figure: White Noise (ϵ) comparison

Putting All Together

$$\begin{aligned}k(x, x') &= RBF(x, x') \cdot \sigma + \varepsilon \\ &= \exp(\|x - x'\|/2L) \cdot \sigma + \varepsilon\end{aligned}$$

Putting All Together

- L lengthscale is input measurement unit

$$\begin{aligned}k(x, x') &= RBF(x, x') \cdot \sigma + \varepsilon \\ &= \exp(\|x - x'\|/2L) \cdot \sigma + \varepsilon\end{aligned}$$

Putting All Together

- L lengthscale is input measurement unit
- σ amplitude is output variability

$$\begin{aligned}k(x, x') &= RBF(x, x') \cdot \sigma + \varepsilon \\ &= \exp(\|x - x'\|/2L) \cdot \sigma + \varepsilon\end{aligned}$$

Putting All Together

- L lengthscale is input measurement unit
- σ amplitude is output variability
- ϵ white noise is output noise

$$\begin{aligned}k(x, x') &= RBF(x, x') \cdot \sigma + \epsilon \\ &= \exp(\|x - x'\|/2L) \cdot \sigma + \epsilon\end{aligned}$$

Putting All Together

- L lengthscale is input measurement unit
- σ amplitude is output variability
- ε white noise is output noise

$$\begin{aligned}k(x, x') &= RBF(x, x') \cdot \sigma + \varepsilon \\ &= \exp(\|x - x'\|/2L) \cdot \sigma + \varepsilon\end{aligned}$$

Note

Lengthscales can be put out of the kernel and are not their intrinsic property (for most of them)

$$\exp(\|x - x'\|/2L) = \exp(\|x/\textcolor{red}{L} - x'/\textcolor{red}{L}\|/2)$$

Kernel Types

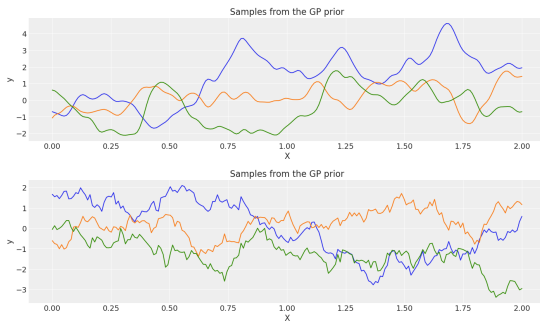
Every kernel is a structural assumption

- Stationary
- Periodic/Circular
- Linear/Polynomial (non stationary)

Kernel Types

Every kernel is a structural assumption

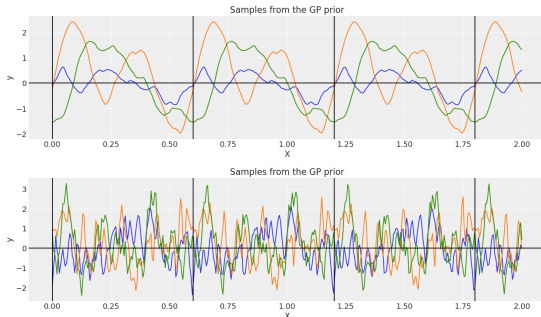
- Stationary
 - "If I extrapolate, I go back to prior"
- Periodic/Circular
- Linear/Polynomial (non stationary)



Kernel Types

Every kernel is a structural assumption

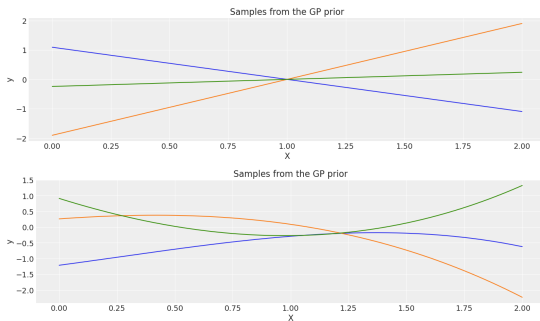
- Stationary
 - "If I extrapolate, I go back to prior"
- Periodic/Circular
 - "Things are similar over time"
- Linear/Polynomial (non stationary)



Kernel Types

Every kernel is a structural assumption

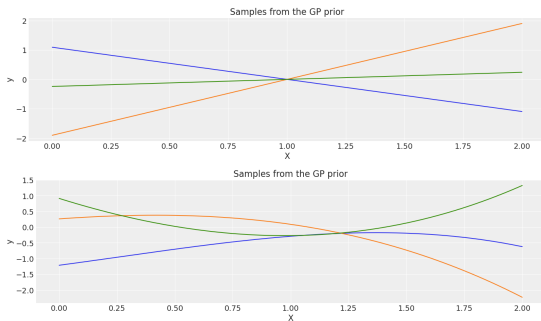
- Stationary
 - "If I extrapolate, I go back to prior"
- Periodic/Circular
 - "Things are similar over time"
- Linear/Polynomial (non stationary)
 - Linear Regression
 - Polynomial Regression



Kernel Types

Every kernel is a structural assumption

- Stationary
 - "If I extrapolate, I go back to prior"
- Periodic/Circular
 - "Things are similar over time"
- Linear/Polynomial (non stationary)
 - Linear Regression
 - Polynomial Regression



Kernel math power

You can combine basic properties of the kernels together. Examples [here](#).
Combining kernels is art. Art is for the seminar.

Combining Kernels

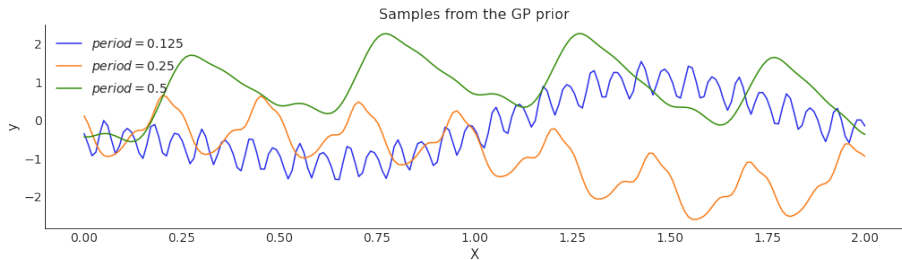


Figure: Exponential and Periodic kernel

Combining Kernels

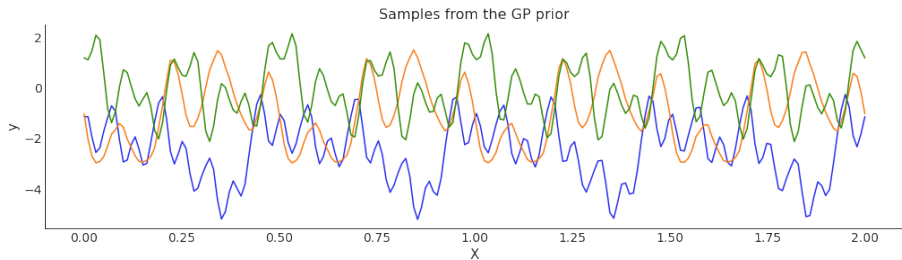


Figure: Multiple Periodic kernels

Combining Kernels

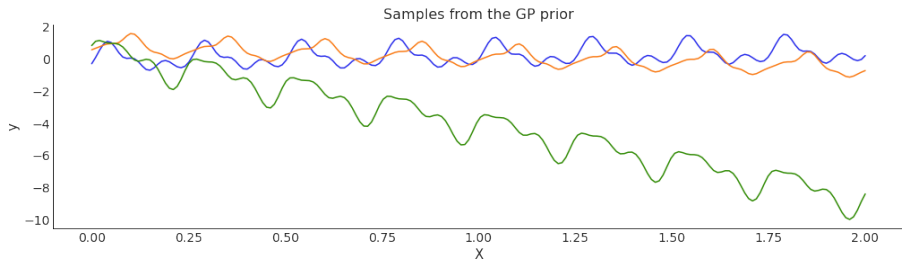


Figure: Linear and Periodic kernels

Summary

- Kernels represent structural patterns
- Patterns can be learned from data
- Combining kernels you combine patterns that can be learned

Motivation

There are cases where GP is a sharp knife to solve the problem. They look like

- My parameter changes over time [[3](#)]
- I have a time series [[1](#)]
- I have spatial data
- I have spatial data and time series

Our Example

The favorite 8 schools

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\theta_i \sim \text{Normal}(\mu, \tau)$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

Where data are pairs $\{(y_i, \sigma_i)\}$

Our Example

The favorite 8 schools

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

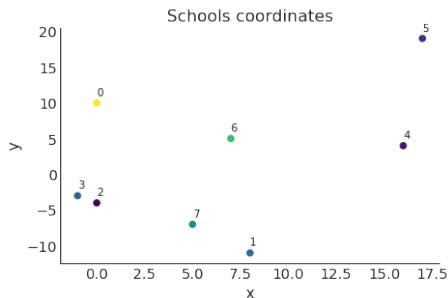
$$\theta_i \sim \text{Normal}(\mu, \tau)$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

Where data are pairs $\{(y_i, \sigma_i)\}$

But:

- What if we have additional information?
- Coordinates of schools?



Our Example

The favorite 8 schools

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

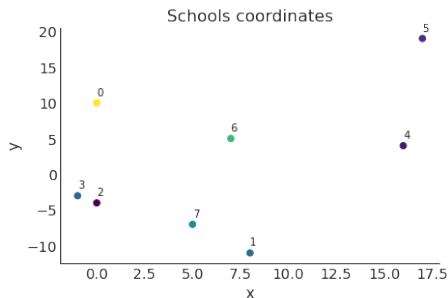
$$\theta_i \sim \text{Normal}(\mu, \tau)$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

Where data are pairs $\{(y_i, \sigma_i)\}$

But:

- What if we have additional information?
- Coordinates of schools?

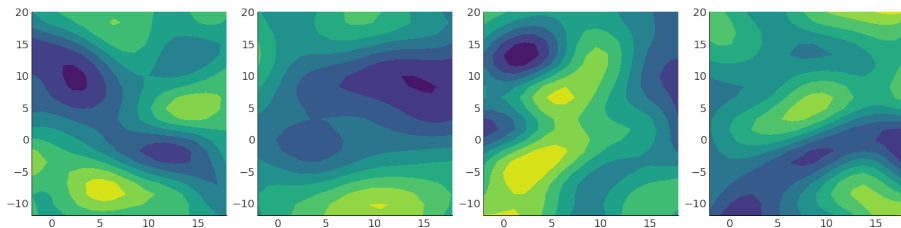


Assumption

Neighboring schools are similar

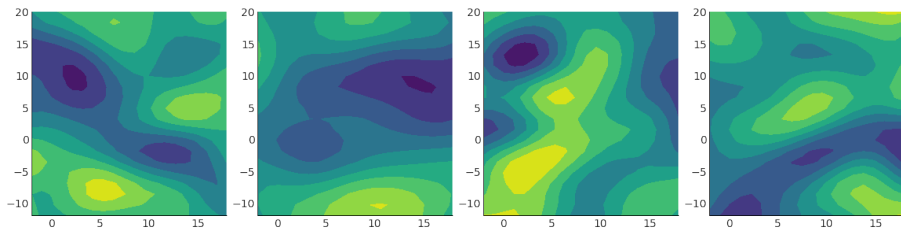
Spatial Gaussian Process

- A smooth function over 2d space
- Lengthscale is important but can be inferred



Spatial Gaussian Process

- A smooth function over 2d space
- Lengthscale is important but can be inferred



Idea

Instead of independent hierarchy, use GP hierarchy!

The GP Hierarchy

Before:

(centered)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\theta_i \sim \text{Normal}(\mu, \tau)$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

(noncentered)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\bar{\theta}_i \sim \text{Normal}(0, 1)$$

$$\theta_i = \mu + \tau \cdot \bar{\theta}_i$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

After:

(noncentered + GP)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\bar{\theta}_i \sim \mathcal{GP}(x_i)$$

$$\theta_i = \mu + \tau \cdot \bar{\theta}_i$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

The GP Hierarchy

Before:

(centered)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\theta_i \sim \text{Normal}(\mu, \tau)$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

(noncentered)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\bar{\theta}_i \sim \text{Normal}(0, 1)$$

$$\theta_i = \mu + \tau \cdot \bar{\theta}_i$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

After:

(noncentered + GP)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\bar{\theta}_i \sim \mathcal{GP}(x_i)$$

$$\theta_i = \mu + \tau \cdot \bar{\theta}_i$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

Comments

Centered parametrization has geometry issues (lecture 2)

The GP Hierarchy

Before:

(centered)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\theta_i \sim \text{Normal}(\mu, \tau)$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

(noncentered)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\bar{\theta}_i \sim \text{Normal}(0, 1)$$

$$\theta_i = \mu + \tau \cdot \bar{\theta}_i$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

After:

(noncentered + GP)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\bar{\theta}_i \sim \mathcal{GP}(x_i)$$

$$\theta_i = \mu + \tau \cdot \bar{\theta}_i$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

Comments

Issues can be resolved with noncentered parametrization

The GP Hierarchy

Before:

(centered)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\theta_i \sim \text{Normal}(\mu, \tau)$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

(noncentered)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\bar{\theta}_i \sim \text{Normal}(0, 1)$$

$$\theta_i = \mu + \tau \cdot \bar{\theta}_i$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

After:

(noncentered + GP)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\bar{\theta}_i \sim \mathcal{GP}(x_i)$$

$$\theta_i = \mu + \tau \cdot \bar{\theta}_i$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

Comments

In the original model, θ_i (or $\bar{\theta}_i$) is independent per school

The GP Hierarchy

Before:

(centered)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\theta_i \sim \text{Normal}(\mu, \tau)$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

(noncentered)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\bar{\theta}_i \sim \text{Normal}(0, 1)$$

$$\theta_i = \mu + \tau \cdot \bar{\theta}_i$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

After:

(noncentered + GP)

$$\mu \sim \text{Normal}(0, 5)$$

$$\tau \sim \text{HalfCauchy}(5)$$

$$\bar{\theta}_i \sim \mathcal{GP}(x_i)$$

$$\theta_i = \mu + \tau \cdot \bar{\theta}_i$$

$$y_i \sim \text{Normal}(\theta_i, \sigma_i)$$

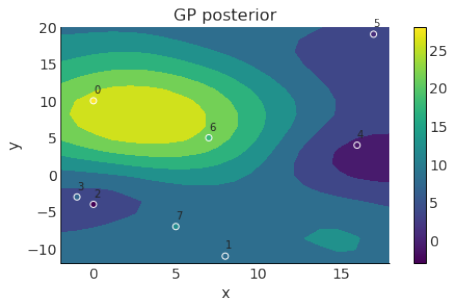
Comments

Gaussian Process adds dependencies between schools so close ones are similar. $\sigma_{\mathcal{GP}} = 1$

Results and Takeaways

GP Gotchas

- 1 Flexible structure
- 2 Smart hierarchy
- 3 Predictions for new objects



References I



G. Corani, A. Benavoli, and M. Zaffalon.

Time series forecasting with gaussian processes needs priors.

In Y. Dong, N. Kourtellis, B. Hammer, and J. A. Lozano, editors, *Machine Learning and Knowledge Discovery in Databases. Applied Data Science Track*, pages 103–117, Cham, 2021. Springer International Publishing.



M. Cuturi.

Positive definite kernels in machine learning, 2009.



T. Wiecki.

Rolling regressino in pymc-devs/pymc-examples, Jan. 2022.