02477 – Bayesian Machine Learning: Lecture 5

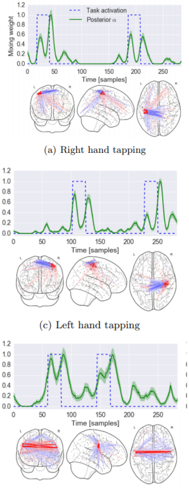
Michael Riis Andersen

Technical University of Denmark,
DTU Compute, Department of Applied Math and Computer Science

Outline

Multitude of Gaussian processes applications

- Regression (supervised learning)
  - Time series analysis
  - EEG brain imaging
  - Survival analysis for cancer data
  - Predicting rainfall
  - Robot dynamics
  - ...
- Classification (supervised learning)
  - Recognizings human movements
  - Brain decoding
  - ...
- Used as building block in more complex models
- Dimensionality reduction (unsupervised learning)
- Optimization of black functions (Bayesian optimization)
- Numerical integration (Bayesian quadrature)
- Solving differential equations (probabilistic numerics)



(a) Right hand tapping
(c) Left hand tapping
(b) Tongue wagging

Towards prior distributions for function spaces

Parametric models

- In week 3, we studied linear models of the form

$$y_n = f_n + e_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w} + e_n$$

- In week 4, we studied Bayesian logistic regression

$$y_n | f_n \sim Ber(\sigma(f_n))$$
$$f_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w}$$

- Typical workflow
  1. Specify prior $p(\boldsymbol{w})$ and likelihood $p(\boldsymbol{y}|\boldsymbol{w})$
  2. Calculate posterior distribution $p(\boldsymbol{w}|\boldsymbol{y})$
  3. Make predictions based on the predictive distribution $p(y^*|\boldsymbol{y}, \boldsymbol{x}^*)$

Parametric models

- In week 3, we studied linear models of the form

$$y_n = f_n + e_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w} + e_n$$

- In week 4, we studied Bayesian logistic regression

$$y_n | f_n \sim Ber(\sigma(f_n))$$
$$f_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w}$$

- Typical workflow
  1. Specify prior $p(\boldsymbol{w})$ and likelihood $p(\boldsymbol{y}|\boldsymbol{w})$
  2. Calculate posterior distribution $p(\boldsymbol{w}|\boldsymbol{y})$
  3. Make predictions based on the predictive distribution $p(y^*|\boldsymbol{y}, \boldsymbol{x}^*)$

- All we care about is the parameters $\boldsymbol{w}$ - Once we have calculated the posterior distribution $p(\boldsymbol{w}|\boldsymbol{y})$, we don't need the training data $\mathcal{D} = \{\boldsymbol{X}, \boldsymbol{y}\}$ anymore

Parametric models

- In week 3, we studied linear models of the form

$$y_n = f_n + e_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w} + e_n$$

- In week 4, we studied Bayesian logistic regression

$$y_n | f_n \sim Ber(\sigma(f_n))$$
$$f_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w}$$

- Typical workflow
  1. Specify prior $p(\boldsymbol{w})$ and likelihood $p(\boldsymbol{y}|\boldsymbol{w})$
  2. Calculate posterior distribution $p(\boldsymbol{w}|\boldsymbol{y})$
  3. Make predictions based on the predictive distribution $p(y^*|\boldsymbol{y}, \boldsymbol{x}^*)$

- All we care about is the parameters $\boldsymbol{w}$ - Once we have calculated the posterior distribution $p(\boldsymbol{w}|\boldsymbol{y})$, we don't need the training data $\mathcal{D} = \{\boldsymbol{X}, \boldsymbol{y}\}$ anymore

- Linear and logistic regression are both *parametric models*: probability distributions indexed by finite dimensional parameters

From parameters to functions I

- Our linear model

$$y_n = f_n + e_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w} + e_n$$

From parameters to functions I

- Our linear model

$$y_n = f_n + e_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w} + e_n$$

- Our goal was to learn the latent function

$$f_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w}$$

From parameters to functions I

- Our linear model

$$y_n = f_n + e_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w} + e_n$$

- Our goal was to learn the latent function

$$f_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w}$$

- We focussed on $\boldsymbol{w}$ and the joint distribution via the product rule

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

From parameters to functions I

- Our linear model

$$y_n = f_n + e_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w} + e_n$$

- Our goal was to learn the latent function

$$f_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w}$$

- We focussed on $\boldsymbol{w}$ and the joint distribution via the product rule

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

- Let $\boldsymbol{f}$ denote the function values, i.e. $\boldsymbol{f} = \begin{bmatrix} f(\phi(\boldsymbol{x}_1)) & f(\phi(\boldsymbol{x}_2)) & \dots & f(\phi(\boldsymbol{x}_N)) \end{bmatrix}$

$$p(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{w}) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})$$

From parameters to functions I

- Our linear model

$$y_n = f_n + e_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w} + e_n$$

- Our goal was to learn the latent function

$$f_n = \phi(\boldsymbol{x}_n)^T \boldsymbol{w}$$

- We focussed on $\boldsymbol{w}$ and the joint distribution via the product rule

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

- Let $\boldsymbol{f}$ denote the function values, i.e. $\boldsymbol{f} = \begin{bmatrix} f(\phi(\boldsymbol{x}_1)) & f(\phi(\boldsymbol{x}_2)) & \dots & f(\phi(\boldsymbol{x}_N)) \end{bmatrix}$

$$p(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{w}) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})$$

- The model is the same - we can recover the old model formulation via the sum rule

$$p(\boldsymbol{y}, \boldsymbol{w}) = \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{f}$$

From parameters to functions II

- The augmented model

$$p(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{w}) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})$$

- What if we integrate out the parameters instead?

$$p(\boldsymbol{y}, \boldsymbol{f}) = p(\boldsymbol{y}|\boldsymbol{f}) \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f})$$

where $p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w}$

From parameters to functions II

- The augmented model

$$p(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{w}) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})$$

- What if we integrate out the parameters instead?

$$p(\boldsymbol{y}, \boldsymbol{f}) = p(\boldsymbol{y}|\boldsymbol{f}) \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f})$$

where $p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w}$

- Let's study the distribution of $\boldsymbol{f} = \boldsymbol{\Phi}\boldsymbol{w}$ for our Gaussian prior on $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})$

$$p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = \int p(\boldsymbol{f}|\boldsymbol{w})\mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})\mathrm{d}\boldsymbol{w}$$

From parameters to functions II

- The augmented model

$$p(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{w}) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})$$

- What if we integrate out the parameters instead?

$$p(\boldsymbol{y}, \boldsymbol{f}) = p(\boldsymbol{y}|\boldsymbol{f}) \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f})$$

where $p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w}$

- Let's study the distribution of $\boldsymbol{f} = \boldsymbol{\Phi}\boldsymbol{w}$ for our Gaussian prior on $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})$

$$p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = \int p(\boldsymbol{f}|\boldsymbol{w})\mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})\mathrm{d}\boldsymbol{w}$$

- We could do the integral directly, let's use this result instead

$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{V}) \qquad \Rightarrow \qquad \boldsymbol{a} + \boldsymbol{B}\boldsymbol{x} \sim \mathcal{N}\left(\boldsymbol{a} + \boldsymbol{B}\boldsymbol{m}, \boldsymbol{B}\boldsymbol{V}\boldsymbol{B}^T\right)$$

From parameters to functions II

- The augmented model

$$p(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{w}) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})$$

- What if we integrate out the parameters instead?

$$p(\boldsymbol{y}, \boldsymbol{f}) = p(\boldsymbol{y}|\boldsymbol{f}) \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f})$$

where $p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w}$

- Let's study the distribution of $\boldsymbol{f} = \boldsymbol{\Phi}\boldsymbol{w}$ for our Gaussian prior on $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})$

$$p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = \int p(\boldsymbol{f}|\boldsymbol{w})\mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})\mathrm{d}\boldsymbol{w}$$

- We could do the integral directly, let's use this result instead

$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{V}) \qquad \Rightarrow \qquad \boldsymbol{a} + \boldsymbol{B}\boldsymbol{x} \sim \mathcal{N}\left(\boldsymbol{a} + \boldsymbol{B}\boldsymbol{m}, \boldsymbol{B}\boldsymbol{V}\boldsymbol{B}^T\right)$$

- What is the distribution of $\boldsymbol{f}$?

From parameters to functions II

- The augmented model

$$p(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{w}) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})$$

- What if we integrate out the parameters instead?

$$p(\boldsymbol{y}, \boldsymbol{f}) = p(\boldsymbol{y}|\boldsymbol{f}) \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f})$$

where $p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w}$

- Let's study the distribution of $\boldsymbol{f} = \boldsymbol{\Phi}\boldsymbol{w}$ for our Gaussian prior on $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})$

$$p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = \int p(\boldsymbol{f}|\boldsymbol{w})\mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})\mathrm{d}\boldsymbol{w}$$

- We could do the integral directly, let's use this result instead

$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{V}) \qquad \Rightarrow \qquad \boldsymbol{a} + \boldsymbol{B}\boldsymbol{x} \sim \mathcal{N}\left(\boldsymbol{a} + \boldsymbol{B}\boldsymbol{m}, \boldsymbol{B}\boldsymbol{V}\boldsymbol{B}^T\right)$$

- What is the distribution of $\boldsymbol{f}$?

$$\mathbb{E}[\boldsymbol{f}] = \boldsymbol{0} + \boldsymbol{\Phi}\boldsymbol{0} = \boldsymbol{0} \qquad \mathbb{V}[\boldsymbol{f}] = \boldsymbol{\Phi}\alpha^{-1}\boldsymbol{I}\boldsymbol{\Phi}^T = \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T$$

From parameters to functions II

- The augmented model

$$p(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{w}) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})$$

- What if we integrate out the parameters instead?

$$p(\boldsymbol{y}, \boldsymbol{f}) = p(\boldsymbol{y}|\boldsymbol{f}) \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f})$$

  where $p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w}$

- Let's study the distribution of $\boldsymbol{f} = \boldsymbol{\Phi}\boldsymbol{w}$ for our Gaussian prior on $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})$

$$p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = \int p(\boldsymbol{f}|\boldsymbol{w})\mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})\mathrm{d}\boldsymbol{w}$$

- We could do the integral directly, let's use this result instead

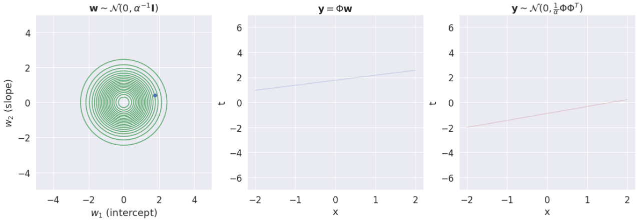
$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{V}) \qquad \Rightarrow \qquad \boldsymbol{a} + \boldsymbol{B}\boldsymbol{x} \sim \mathcal{N}\left(\boldsymbol{a} + \boldsymbol{B}\boldsymbol{m}, \boldsymbol{B}\boldsymbol{V}\boldsymbol{B}^T\right)$$

- What is the distribution of $\boldsymbol{f}$?

$$\mathbb{E}[\boldsymbol{f}] = \boldsymbol{0} + \boldsymbol{\Phi}\boldsymbol{0} = \boldsymbol{0} \qquad \mathbb{V}[\boldsymbol{f}] = \boldsymbol{\Phi}\alpha^{-1}\boldsymbol{I}\boldsymbol{\Phi}^T = \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T$$

Changing perspective from weight space to function space

$$p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = \int p(\boldsymbol{f}|\boldsymbol{w})\mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})\mathrm{d}\boldsymbol{w} = \mathcal{N}\left(\boldsymbol{f}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)$$



- Two ways to generate samples of $\boldsymbol{f} \sim p(\boldsymbol{f})$

- Weight space-perspective

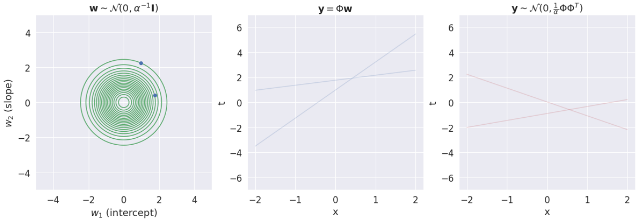
  Step 1: Generate a sample $\boldsymbol{w}^{(i)} \sim p(\boldsymbol{w})$
  Step 2: Compute $\boldsymbol{f}^{(i)} = \boldsymbol{\Phi}\boldsymbol{w}^{(i)}$

- Function space-perspective

  Step 1: Generate a sample $\boldsymbol{f}^{(i)} \sim \mathcal{N}\left(\boldsymbol{0}, \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)$

Changing perspective from weight space to function space

$$p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = \int p(\boldsymbol{f}|\boldsymbol{w})\mathcal{N}(\boldsymbol{w}|\mathbf{0}, \alpha^{-1}\boldsymbol{I})\mathrm{d}\boldsymbol{w} = \mathcal{N}\left(\boldsymbol{f}|\mathbf{0}, \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)$$



- Two ways to generate samples of $\boldsymbol{f} \sim p(\boldsymbol{f})$

- Weight space-perspective

  Step 1: Generate a sample $\boldsymbol{w}^{(i)} \sim p(\boldsymbol{w})$

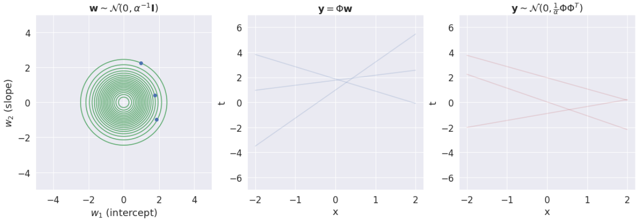
  Step 2: Compute $\boldsymbol{f}^{(i)} = \boldsymbol{\Phi}\boldsymbol{w}^{(i)}$

- Function space-perspective

  Step 1: Generate a sample $\boldsymbol{f}^{(i)} \sim \mathcal{N}\left(\mathbf{0}, \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)$

Changing perspective from weight space to function space

$$p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = \int p(\boldsymbol{f}|\boldsymbol{w})\mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})\mathrm{d}\boldsymbol{w} = \mathcal{N}\left(\boldsymbol{f}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)$$



- Two ways to generate samples of $\boldsymbol{f} \sim p(\boldsymbol{f})$

- Weight space-perspective

  Step 1: Generate a sample $\boldsymbol{w}^{(i)} \sim p(\boldsymbol{w})$

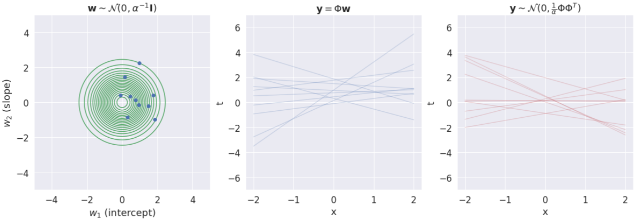
  Step 2: Compute $\boldsymbol{f}^{(i)} = \boldsymbol{\Phi}\boldsymbol{w}^{(i)}$

- Function space-perspective

  Step 1: Generate a sample $\boldsymbol{f}^{(i)} \sim \mathcal{N}\left(\boldsymbol{0}, \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)$

Changing perspective from weight space to function space

$$p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = \int p(\boldsymbol{f}|\boldsymbol{w})\mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})\mathrm{d}\boldsymbol{w} = \mathcal{N}\left(\boldsymbol{f}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)$$



- Two ways to generate samples of $\boldsymbol{f} \sim p(\boldsymbol{f})$

- Weight space-perspective

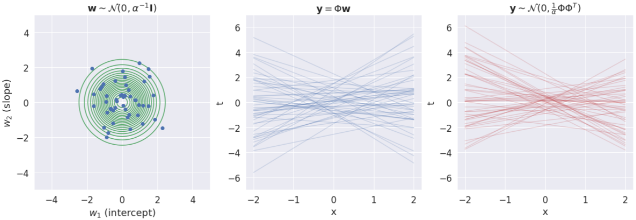
  Step 1: Generate a sample $\boldsymbol{w}^{(i)} \sim p(\boldsymbol{w})$
  Step 2: Compute $\boldsymbol{f}^{(i)} = \boldsymbol{\Phi}\boldsymbol{w}^{(i)}$

- Function space-perspective

  Step 1: Generate a sample $\boldsymbol{f}^{(i)} \sim \mathcal{N}\left(\boldsymbol{0}, \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)$

Changing perspective from weight space to function space

$$p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = \int p(\boldsymbol{f}|\boldsymbol{w})\mathcal{N}(\boldsymbol{w}|\mathbf{0}, \alpha^{-1}\boldsymbol{I})\mathrm{d}\boldsymbol{w} = \mathcal{N}\left(\boldsymbol{f}|\mathbf{0}, \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)$$



- Two ways to generate samples of $\boldsymbol{f} \sim p(\boldsymbol{f})$

- Weight space-perspective

  Step 1: Generate a sample $\boldsymbol{w}^{(i)} \sim p(\boldsymbol{w})$

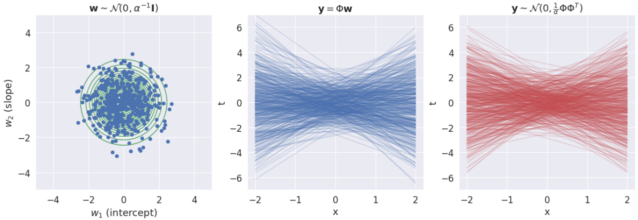
  Step 2: Compute $\boldsymbol{f}^{(i)} = \boldsymbol{\Phi}\boldsymbol{w}^{(i)}$

- Function space-perspective

  Step 1: Generate a sample $\boldsymbol{f}^{(i)} \sim \mathcal{N}\left(\mathbf{0}, \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)$

Changing perspective from weight space to function space

$$p(\boldsymbol{f}) = \int p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = \int p(\boldsymbol{f}|\boldsymbol{w})\mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{I})\mathrm{d}\boldsymbol{w} = \mathcal{N}\left(\boldsymbol{f}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)$$



- Two ways to generate samples of $\boldsymbol{f} \sim p(\boldsymbol{f})$

- Weight space-perspective

  Step 1: Generate a sample $\boldsymbol{w}^{(i)} \sim p(\boldsymbol{w})$

  Step 2: Compute $\boldsymbol{f}^{(i)} = \boldsymbol{\Phi}\boldsymbol{w}^{(i)}$

- Function space-perspective

  Step 1: Generate a sample $\boldsymbol{f}^{(i)} \sim \mathcal{N}\left(\boldsymbol{0}, \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)$

A closer look at the covariance

- A prior on linear functions: $p(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f}|\mathbf{0}, \boldsymbol{K}\right)$, where $\boldsymbol{K} = \frac{1}{\alpha}\boldsymbol{\Phi}\boldsymbol{\Phi}^T$

- A closer look on the covariance between f_i and f_j

$$\boldsymbol{K}_{ij} = \text{cov}\left(y_i, y_j\right) = \text{cov}\left(f(\boldsymbol{x}_i), f(\boldsymbol{x}_j)\right)$$

A closer look at the covariance

- A prior on linear functions: $p(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f}|\mathbf{0}, \boldsymbol{K}\right)$, where $\boldsymbol{K} = \frac{1}{\alpha}\boldsymbol{\Phi}\boldsymbol{\Phi}^T$

- A closer look on the covariance between f_i and f_j

$$\begin{aligned}\boldsymbol{K}_{ij} = \text{cov}\left(y_i, y_j\right) &= \text{cov}\left(f(\boldsymbol{x}_i), f(\boldsymbol{x}_j)\right) \\ &= \text{cov}\left(\phi(\boldsymbol{x}_i)^T\boldsymbol{w}, \phi(\boldsymbol{x}_j)^T\boldsymbol{w}\right)\end{aligned}$$

A closer look at the covariance

- A prior on linear functions: $p(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f}|\boldsymbol{0}, \boldsymbol{K}\right)$, where $\boldsymbol{K} = \frac{1}{\alpha}\boldsymbol{\Phi}\boldsymbol{\Phi}^T$

- A closer look on the covariance between f_i and f_j

$$\begin{aligned}\boldsymbol{K}_{ij} = \text{cov}\left(y_i, y_j\right) &= \text{cov}\left(f(\boldsymbol{x}_i), f(\boldsymbol{x}_j)\right)\\ &= \text{cov}\left(\phi(\boldsymbol{x}_i)^T\boldsymbol{w}, \phi(\boldsymbol{x}_j)^T\boldsymbol{w}\right)\\ &= \phi(\boldsymbol{x}_i)^T\text{cov}\left(\boldsymbol{w}, \boldsymbol{w}\right)\phi(\boldsymbol{x}_j)\end{aligned}$$

A closer look at the covariance

- A prior on linear functions: $p(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f}|\mathbf{0}, \boldsymbol{K}\right)$, where $\boldsymbol{K} = \frac{1}{\alpha}\boldsymbol{\Phi}\boldsymbol{\Phi}^T$

- A closer look on the covariance between f_i and f_j

$$\begin{aligned}
\boldsymbol{K}_{ij} = \text{cov}\left(y_i, y_j\right) &= \text{cov}\left(f(\boldsymbol{x}_i), f(\boldsymbol{x}_j)\right) \\
&= \text{cov}\left(\phi(\boldsymbol{x}_i)^T\boldsymbol{w}, \phi(\boldsymbol{x}_j)^T\boldsymbol{w}\right) \\
&= \phi(\boldsymbol{x}_i)^T\text{cov}\left(\boldsymbol{w}, \boldsymbol{w}\right)\phi(\boldsymbol{x}_j) \\
&= \phi(\boldsymbol{x}_i)^T\mathbb{V}\left(\boldsymbol{w}\right)\phi(\boldsymbol{x}_j)
\end{aligned}$$

A closer look at the covariance

- A prior on linear functions: $p(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f}|\mathbf{0}, \boldsymbol{K}\right)$, where $\boldsymbol{K} = \frac{1}{\alpha}\boldsymbol{\Phi}\boldsymbol{\Phi}^T$

- A closer look on the covariance between f_i and f_j

$$
\begin{aligned}
\boldsymbol{K}_{ij} = \mathrm{cov}\left(y_i, y_j\right) &= \mathrm{cov}\left(f(\boldsymbol{x}_i), f(\boldsymbol{x}_j)\right) \\
&= \mathrm{cov}\left(\phi(\boldsymbol{x}_i)^T\boldsymbol{w}, \phi(\boldsymbol{x}_j)^T\boldsymbol{w}\right) \\
&= \phi(\boldsymbol{x}_i)^T \mathrm{cov}\left(\boldsymbol{w}, \boldsymbol{w}\right)\phi(\boldsymbol{x}_j) \\
&= \phi(\boldsymbol{x}_i)^T \mathbb{V}\left(\boldsymbol{w}\right)\phi(\boldsymbol{x}_j) \\
&= \phi(\boldsymbol{x}_i)^T \frac{1}{\alpha}\boldsymbol{I}\phi(\boldsymbol{x}_j)
\end{aligned}
$$

A closer look at the covariance

- A prior on linear functions: $p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f}|\mathbf{0}, \boldsymbol{K})$, where $\boldsymbol{K} = \frac{1}{\alpha}\boldsymbol{\Phi}\boldsymbol{\Phi}^T$

- A closer look on the covariance between f_i and f_j

$$\begin{aligned}
\boldsymbol{K}_{ij} = \text{cov}\left(y_i, y_j\right) &= \text{cov}\left(f(\boldsymbol{x}_i), f(\boldsymbol{x}_j)\right) \\
&= \text{cov}\left(\phi(\boldsymbol{x}_i)^T \boldsymbol{w}, \phi(\boldsymbol{x}_j)^T \boldsymbol{w}\right) \\
&= \phi(\boldsymbol{x}_i)^T \text{cov}\left(\boldsymbol{w}, \boldsymbol{w}\right) \phi(\boldsymbol{x}_j) \\
&= \phi(\boldsymbol{x}_i)^T \mathbb{V}\left(\boldsymbol{w}\right) \phi(\boldsymbol{x}_j) \\
&= \phi(\boldsymbol{x}_i)^T \frac{1}{\alpha} \boldsymbol{I} \phi(\boldsymbol{x}_j) \\
&= \frac{1}{\alpha} \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j)
\end{aligned}$$

A closer look at the covariance

- A prior on linear functions: $p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f}|\mathbf{0}, \boldsymbol{K})$, where $\boldsymbol{K} = \frac{1}{\alpha}\boldsymbol{\Phi}\boldsymbol{\Phi}^T$
- A closer look on the covariance between f_i and f_j

$$\begin{aligned}
\boldsymbol{K}_{ij} = \text{cov}\left(y_i, y_j\right) &= \text{cov}\left(f(\boldsymbol{x}_i), f(\boldsymbol{x}_j)\right) \\
&= \text{cov}\left(\phi(\boldsymbol{x}_i)^T \boldsymbol{w}, \phi(\boldsymbol{x}_j)^T \boldsymbol{w}\right) \\
&= \phi(\boldsymbol{x}_i)^T \text{cov}\left(\boldsymbol{w}, \boldsymbol{w}\right) \phi(\boldsymbol{x}_j) \\
&= \phi(\boldsymbol{x}_i)^T \mathbb{V}\left(\boldsymbol{w}\right) \phi(\boldsymbol{x}_j) \\
&= \phi(\boldsymbol{x}_i)^T \frac{1}{\alpha} \boldsymbol{I} \phi(\boldsymbol{x}_j) \\
&= \frac{1}{\alpha} \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j) \\
&\equiv k(\boldsymbol{x}_i, \boldsymbol{x}_j)
\end{aligned}$$

A closer look at the covariance

- A prior on linear functions: $p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f}|\boldsymbol{0}, \boldsymbol{K})$, where $\boldsymbol{K} = \frac{1}{\alpha}\boldsymbol{\Phi}\boldsymbol{\Phi}^T$

- A closer look on the covariance between f_i and f_j

$$
\begin{aligned}
\boldsymbol{K}_{ij} = \text{cov}\left(y_i, y_j\right) &= \text{cov}\left(f(\boldsymbol{x}_i), f(\boldsymbol{x}_j)\right) \\
&= \text{cov}\left(\phi(\boldsymbol{x}_i)^T\boldsymbol{w}, \phi(\boldsymbol{x}_j)^T\boldsymbol{w}\right) \\
&= \phi(\boldsymbol{x}_i)^T\text{cov}\left(\boldsymbol{w}, \boldsymbol{w}\right)\phi(\boldsymbol{x}_j) \\
&= \phi(\boldsymbol{x}_i)^T\mathbb{V}\left(\boldsymbol{w}\right)\phi(\boldsymbol{x}_j) \\
&= \phi(\boldsymbol{x}_i)^T\frac{1}{\alpha}\boldsymbol{I}\phi(\boldsymbol{x}_j) \\
&= \frac{1}{\alpha}\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j) \\
&\equiv k(\boldsymbol{x}_i, \boldsymbol{x}_j)
\end{aligned}
$$

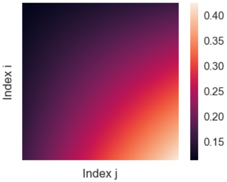
- What happens if we change the *covariance function* k?

Covariance functions

Linear

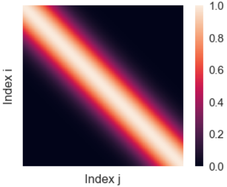
$$k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \frac{1}{\alpha}\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)$$

$\boldsymbol{K}$



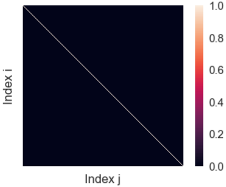
Squared exponential

$$k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\ell^2}\right)$$



White noise

$$k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \delta\left(\boldsymbol{x}_i - \boldsymbol{x}_j\right)$$



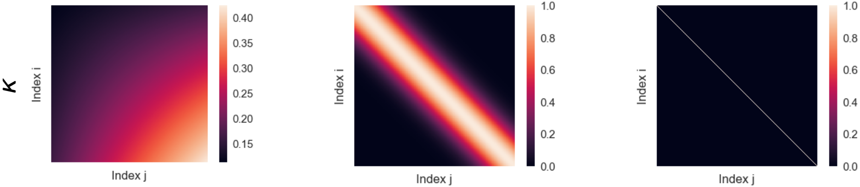
Covariance functions

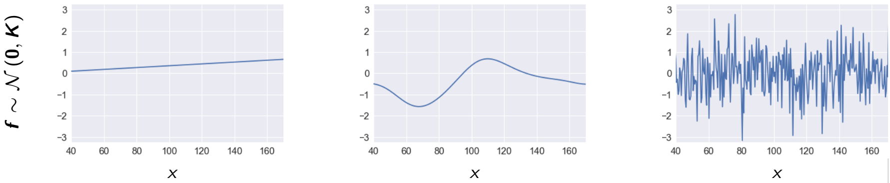
Linear | **Squared exponential** | **White noise**

$$k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \frac{1}{\alpha}\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)$$

$$k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\ell^2}\right)$$

$$k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \delta\left(\boldsymbol{x}_i - \boldsymbol{x}_j\right)$$





Covariance functions

Linear

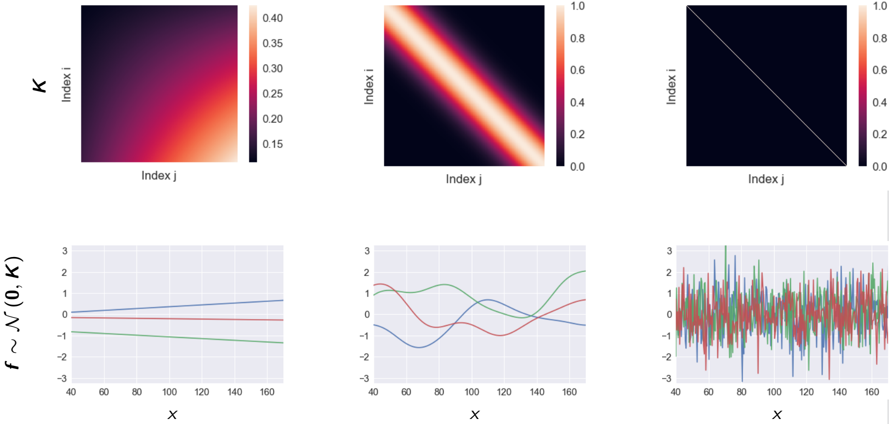
$$k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \frac{1}{\alpha}\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)$$

Squared exponential

$$k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\ell^2}\right)$$

White noise

$$k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \delta\left(\boldsymbol{x}_i - \boldsymbol{x}_j\right)$$



Covariance functions

Linear

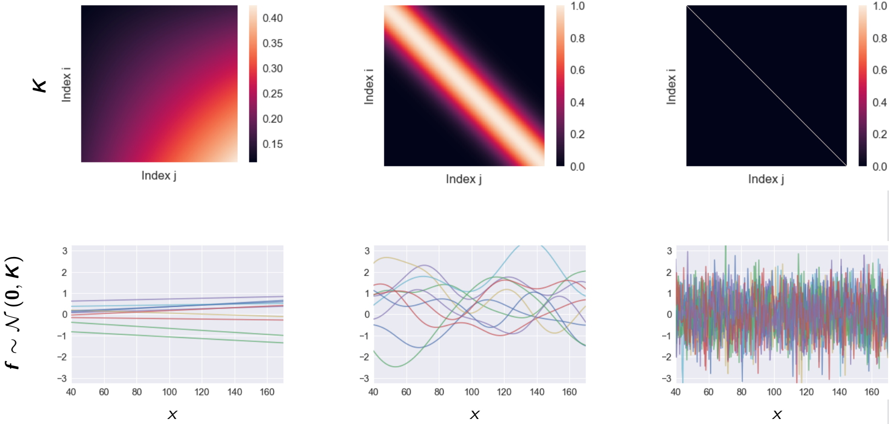
$$k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \frac{1}{\alpha}\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$$

Squared exponential

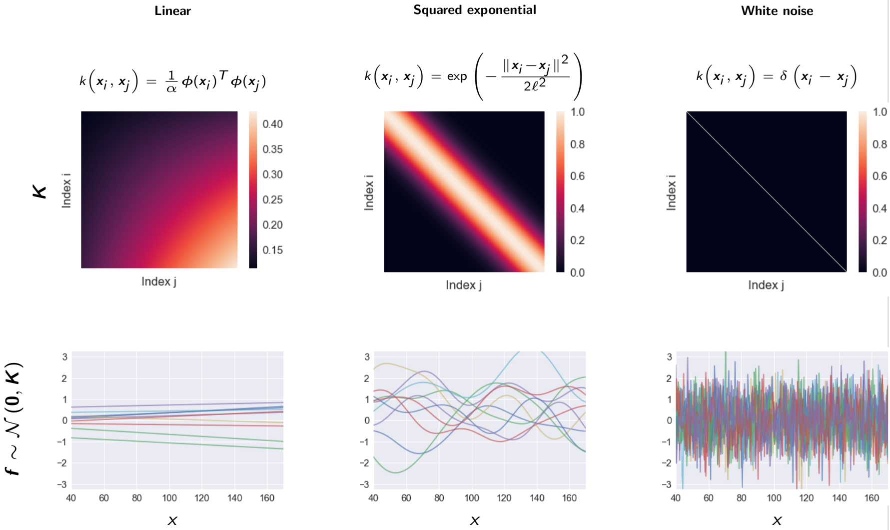
$$k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\ell^2}\right)$$

White noise

$$k\left(\mathbf{x}_i, \mathbf{x}_j\right) = \delta\left(\mathbf{x}_i - \mathbf{x}_j\right)$$



Covariance functions



The form of the covariance function determines the characteristics of the functions

The big picture: Summary so far

1. We started with a Bayesian linear model

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

2. We introduced $\boldsymbol{f}$ into the model and marginalized over the weights $\boldsymbol{w}$

$$p(\boldsymbol{y}, \boldsymbol{f}) = \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f})$$

3. This gave us a prior for linear functions in function space $p(\boldsymbol{f})$, where the covariance function for $\boldsymbol{f}$ was given by

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{1}{\alpha}\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)$$

4. By changing the form of the covariance function $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, we can model much more interesting functions

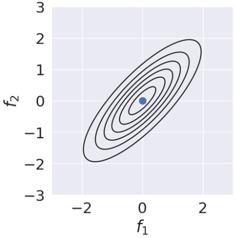
A visual approach towards Gaussian process regression

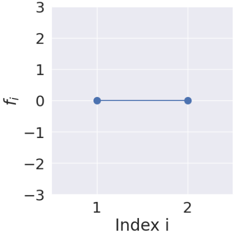
Visualizing samples in higher dimensions

- How can a multivariate normal distribution represent functions?

- Visualizations in 2D

$$\boldsymbol{K} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$



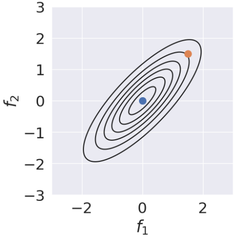


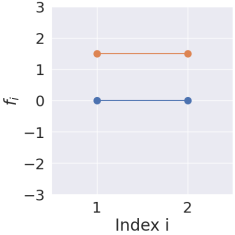
Visualizing samples in higher dimensions

- How can a multivariate normal distribution represent functions?

- Visualizations in 2D

$$\boldsymbol{K} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

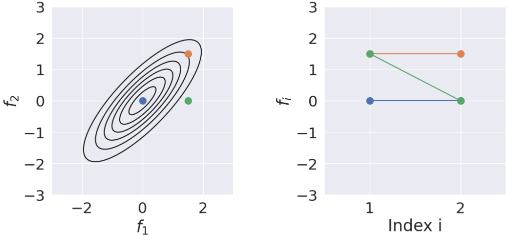




Visualizing samples in higher dimensions

- How can a multivariate normal distribution represent functions?
- Visualizations in 2D

$$\boldsymbol{K} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

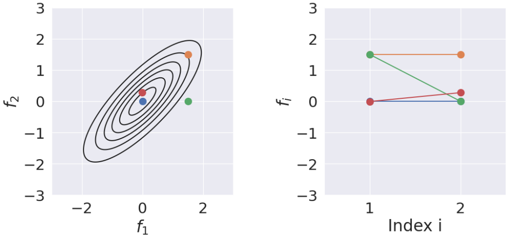


Visualizing samples in higher dimensions

- How can a multivariate normal distribution represent functions?

- Visualizations in 2D

$$\boldsymbol{K} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

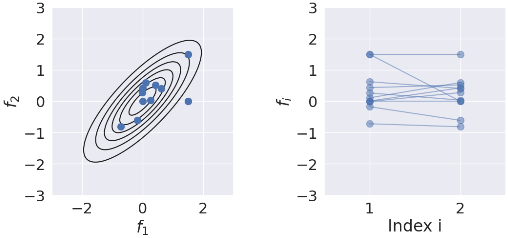


Visualizing samples in higher dimensions

- How can a multivariate normal distribution represent functions?

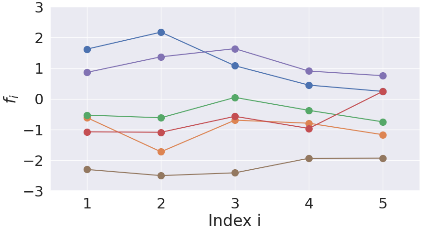
- Visualizations in 2D

$$\boldsymbol{K} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$



Visualizing samples in higher dimensions

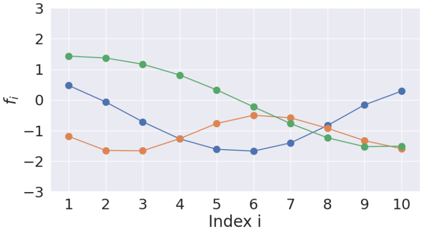
- Visualizations in 5D



$$\boldsymbol{K} = \begin{bmatrix} 1 & 0.8^1 & 0.8^2 & 0.8^3 & 0.8^4 \\ 0.8^1 & 1 & 0.8^1 & 0.8^2 & 0.8^3 \\ 0.8^2 & 0.8^1 & 1 & 0.8^1 & 0.8^2 \\ 0.8^3 & 0.8^2 & 0.8^1 & 1 & 0.8^1 \\ 0.8^4 & 0.8^3 & 0.8^2 & 0.8^1 & 1 \end{bmatrix}$$

Visualizing samples in higher dimensions

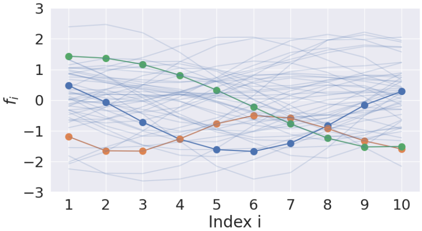
- Visualizations in 10D



$$\boldsymbol{K} = \begin{bmatrix} 1 & 0.8^1 & 0.8^2 & \dots & 0.8^9 \\ 0.8^1 & 1 & 0.8^1 & & \vdots \\ 0.8^2 & 0.8^1 & 1 & & \vdots \\ \vdots & & & \ddots & \vdots \\ 0.8^9 & \dots & \dots & \dots & 1 \end{bmatrix}$$

Visualizing samples in higher dimensions

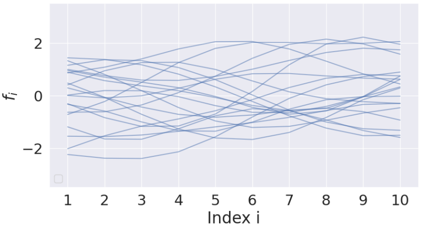
- Visualizations in 10D



$$\boldsymbol{K} = \begin{bmatrix} 1 & 0.8^1 & 0.8^2 & \dots & 0.8^9 \\ 0.8^1 & 1 & 0.8^1 & & \vdots \\ 0.8^2 & 0.8^1 & 1 & & \vdots \\ \vdots & & & \ddots & \vdots \\ 0.8^9 & \dots & \dots & \dots & 1 \end{bmatrix}$$

Conditioning

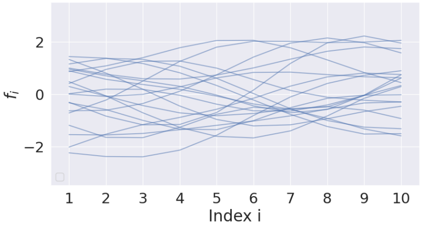
- So far, we have seen samples from the distribution $p(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f} \middle| \boldsymbol{0}, \boldsymbol{K}\right)$



Conditioning

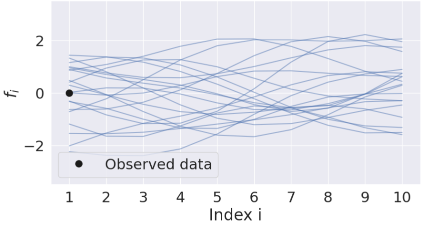
- So far, we have seen samples from the distribution $p(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f} \middle| \mathbf{0}, \boldsymbol{K}\right)$

- We can also write $p(\boldsymbol{f}) = p(f_1, \boldsymbol{f}_{2:10})$



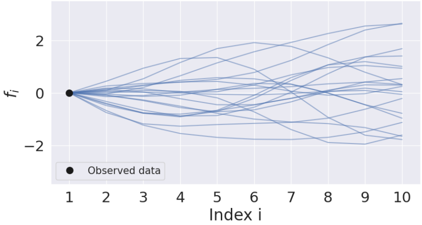
Conditioning

- So far, we have seen samples from the distribution $p(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f} \middle| \mathbf{0}, \boldsymbol{K}\right)$
- We can also write $p(\boldsymbol{f}) = p(f_1, \boldsymbol{f}_{2:10})$
- We now observe $f_1 = 0$



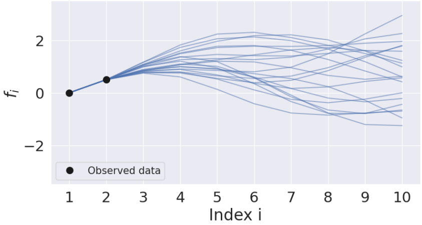
Conditioning

- So far, we have seen samples from the distribution $p(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f} \middle| \boldsymbol{0}, \boldsymbol{K}\right)$
- We can also write $p(\boldsymbol{f}) = p(f_1, \boldsymbol{f}_{2:10})$
- We now observe $f_1 = 0$
- Let's sample from the conditional distribution $p(\boldsymbol{f}_{2:10} \middle| f_1 = 0)$



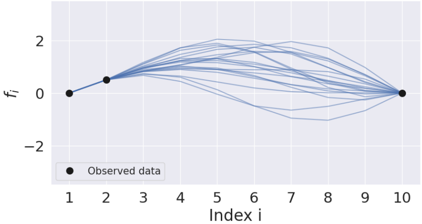
Conditioning

- So far, we have seen samples from the distribution $p\left(\boldsymbol{f}\right) = \mathcal{N}\left(\boldsymbol{f} \middle| \boldsymbol{0}, \boldsymbol{K}\right)$
- We can also write $p\left(\boldsymbol{f}\right) = p(f_1, \boldsymbol{f}_{2:10})$
- We now observe $f_1 = 0$
- Let's sample from the conditional distribution $p(\boldsymbol{f}_{2:10} \middle| f_1 = 0)$



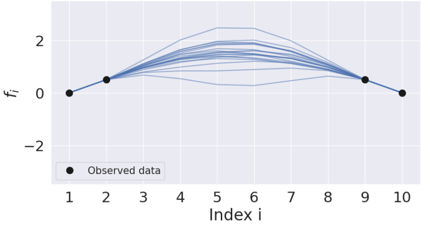
Conditioning

- So far, we have seen samples from the distribution $p(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f} \middle| \boldsymbol{0}, \boldsymbol{K}\right)$
- We can also write $p(\boldsymbol{f}) = p(f_1, \boldsymbol{f}_{2:10})$
- We now observe $f_1 = 0$
- Let's sample from the conditional distribution $p(\boldsymbol{f}_{2:10} \middle| f_1 = 0)$



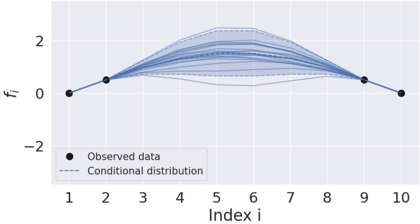
Conditioning

- So far, we have seen samples from the distribution $p\left(\boldsymbol{f}\right) = \mathcal{N}\left(\boldsymbol{f} \middle| \mathbf{0}, \boldsymbol{K}\right)$
- We can also write $p\left(\boldsymbol{f}\right) = p(f_1, \boldsymbol{f}_{2:10})$
- We now observe $f_1 = 0$
- Let's sample from the conditional distribution $p(\boldsymbol{f}_{2:10} \middle| f_1 = 0)$



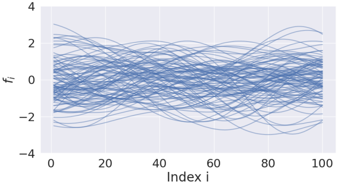
Conditioning

- So far, we have seen samples from the distribution $p\left(\boldsymbol{f}\right) = \mathcal{N}\left(\boldsymbol{f} \middle| \mathbf{0}, \boldsymbol{K}\right)$
- We can also write $p\left(\boldsymbol{f}\right) = p(f_1, \boldsymbol{f}_{2:10})$
- We now observe $f_1 = 0$
- Let's sample from the conditional distribution $p(\boldsymbol{f}_{2:10} \middle| f_1 = 0)$



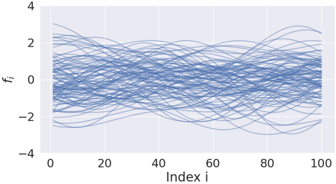
Conditioning II

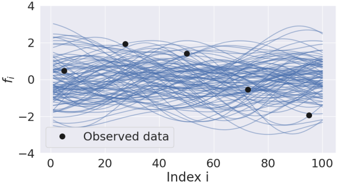
- Let's now consider a case with $\boldsymbol{f} \in \mathbb{R}^{100}$ dimensions with 5 observations



Conditioning II

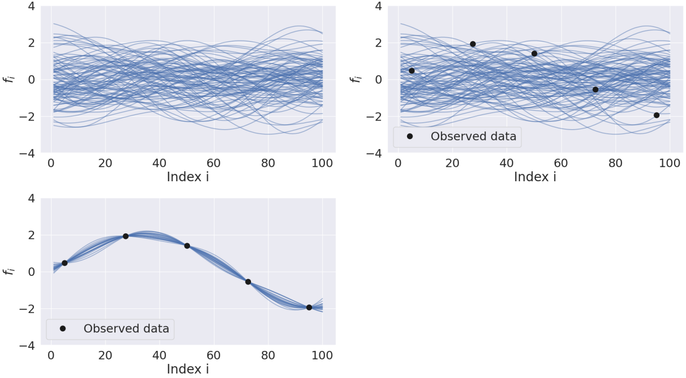
- Let's now consider a case with $\boldsymbol{f} \in \mathbb{R}^{100}$ dimensions with 5 observations





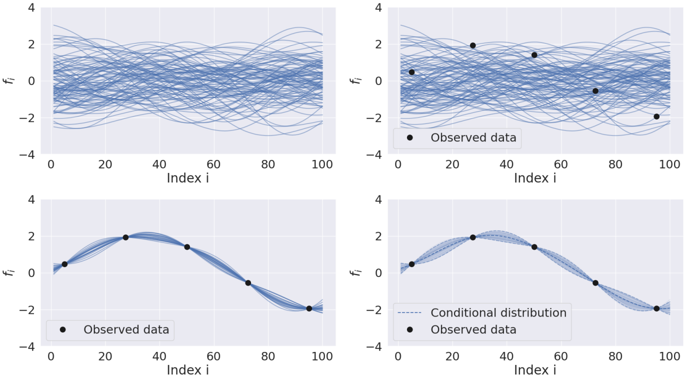
Conditioning II

- Let's now consider a case with $\boldsymbol{f} \in \mathbb{R}^{100}$ dimensions with 5 observations



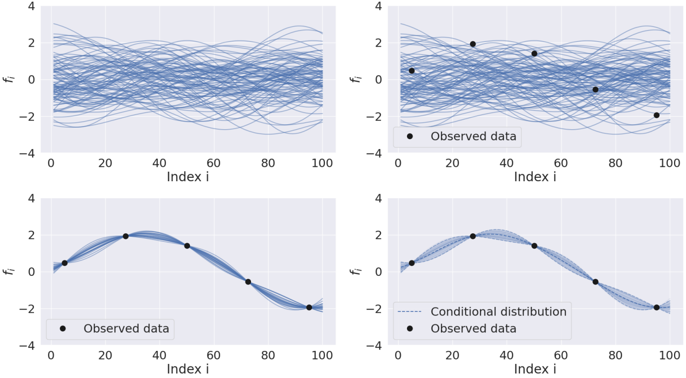
Conditioning II

- Let's now consider a case with $\boldsymbol{f} \in \mathbb{R}^{100}$ dimensions with 5 observations



Conditioning II

- Let's now consider a case with $\boldsymbol{f} \in \mathbb{R}^{100}$ dimensions with 5 observations



- Informally: We can think functions as vectors with infinite dimensions
- Using conditining in multivariate Gaussian distributions, we can do non-linear regression!

Formal definitions

Definition of the multivariate Gaussian distribution

A random vector $\boldsymbol{x} = [x_1, x_2, \cdots, x_D]$ is said to have the **multivariate Gaussian distribution** if all linear combinations of $\boldsymbol{x}$ are (univariate) Gaussian distributed:

$$f = a_1 x_1 + a_2 x_2 + \cdots + a_D x_D \sim \mathcal{N}(m, v)$$

for all $\boldsymbol{a} \in \mathbb{R}^D$

Definition of Gaussian process

A **Gaussian process** (GP) is a collection of random variables, any finite number of which have a joint Gaussian distribution.

Notation and characterization

- We'll use the notation

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

- A Gaussian process can be considered as a prior distribution over functions $f : \mathcal{X} \rightarrow \mathbb{R}$ (the domain $\mathcal{X}$ is typically $\mathbb{R}^D$)

- A Gaussian process is completely characterized by its mean function $m(\boldsymbol{x})$ and its covariance function $k\left(\boldsymbol{x}, \boldsymbol{x}'\right)$.

$$\begin{aligned} m(\boldsymbol{x}) &= \mathbb{E}\left[f(\boldsymbol{x})\right] \\ k\left(\boldsymbol{x}, \boldsymbol{x}'\right) &= \mathbb{E}\left[\left(f(\boldsymbol{x}) - m(\boldsymbol{x})\right)\left(f(\boldsymbol{x}') - m\left(\boldsymbol{x}'\right)\right)\right] \end{aligned}$$

- This means that $f(\boldsymbol{x})$ and $f(\boldsymbol{x}')$ are jointly Gaussian distributed with covariance $k\left(\boldsymbol{x}, \boldsymbol{x}'\right)$

- Not all functions are valid covariance functions - more on that later

Gaussian process regression

DTU

Recall: Linear Gaussian-systems in general (see Section 3.3 in Murphy1)

- For *linear* systems: the Gaussian distribution is *conjugate* to itself
- The *posterior* for a *linear* Gaussian model with Gaussian prior is also *Gaussian*

$$p(\boldsymbol{y}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{z} + \boldsymbol{b}, \boldsymbol{\Sigma}_y) \qquad\qquad p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$$

Recall: Linear Gaussian-systems in general (see Section 3.3 in Murphy1)

- For *linear* systems: the Gaussian distribution is *conjugate* to itself
- The *posterior* for a *linear* Gaussian model with Gaussian prior is also *Gaussian*

$$p(\boldsymbol{y}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{z} + \boldsymbol{b}, \boldsymbol{\Sigma}_y) \qquad p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$$

- The *joint* distribution $p(\boldsymbol{z}, \boldsymbol{y}) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{z}\\ \boldsymbol{y}\end{bmatrix} |\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$

$$\boldsymbol{\mu} = \begin{bmatrix}\boldsymbol{\mu}_z \\ \boldsymbol{W}\boldsymbol{\mu}_z + \boldsymbol{b}\end{bmatrix} \qquad \boldsymbol{\Sigma} = \begin{bmatrix}\boldsymbol{\Sigma}_z & \boldsymbol{\Sigma}_z\boldsymbol{W}^T \\ \boldsymbol{W}\boldsymbol{\Sigma}_Z & \boldsymbol{\Sigma}_y + \boldsymbol{W}\boldsymbol{\Sigma}_z\boldsymbol{W}^T\end{bmatrix}$$

Recall: Linear Gaussian-systems in general (see Section 3.3 in Murphy1)

- For *linear* systems: the Gaussian distribution is *conjugate* to itself

- The *posterior* for a *linear* Gaussian model with Gaussian prior is also *Gaussian*

$$p(\boldsymbol{y}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{z} + \boldsymbol{b}, \boldsymbol{\Sigma}_y) \qquad\qquad p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$$

- The *joint* distribution $p(\boldsymbol{z}, \boldsymbol{y}) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{z}\\ \boldsymbol{y}\end{bmatrix} |\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$

$$\boldsymbol{\mu} = \begin{bmatrix}\boldsymbol{\mu}_z \\ \boldsymbol{W}\boldsymbol{\mu}_z + \boldsymbol{b}\end{bmatrix} \qquad\qquad \boldsymbol{\Sigma} = \begin{bmatrix}\boldsymbol{\Sigma}_z & \boldsymbol{\Sigma}_z \boldsymbol{W}^T \\ \boldsymbol{W}\boldsymbol{\Sigma}_Z & \boldsymbol{\Sigma}_y + \boldsymbol{W}\boldsymbol{\Sigma}_z\boldsymbol{W}^T\end{bmatrix}$$

- The *posterior* distribution of $\boldsymbol{z}$ given $\boldsymbol{y}$

$$p(\boldsymbol{z}|\boldsymbol{y}) = \mathcal{N}\left(\boldsymbol{z}|\boldsymbol{\mu}_{z|y}, \boldsymbol{\Sigma}_{z|y}\right)$$
$$\boldsymbol{\Sigma}_{z|y}^{-1} = \boldsymbol{\Sigma}_z^{-1} + \boldsymbol{W}^T\boldsymbol{\Sigma}_y\boldsymbol{W}$$
$$\boldsymbol{\mu}_{z|y} = \boldsymbol{\Sigma}_{z|y}\left[\boldsymbol{W}^T\boldsymbol{\Sigma}_y^{-1}(\boldsymbol{y} - \boldsymbol{b}) + \boldsymbol{\Sigma}_z^{-1}\boldsymbol{\mu}_z\right]$$

Recall: Linear Gaussian-systems in general (see Section 3.3 in Murphy1)

- For *linear* systems: the Gaussian distribution is *conjugate* to itself
- The *posterior* for a *linear* Gaussian model with Gaussian prior is also *Gaussian*

$$p(\boldsymbol{y}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{z} + \boldsymbol{b}, \boldsymbol{\Sigma}_y) \qquad p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$$

- The *joint* distribution $p(\boldsymbol{z}, \boldsymbol{y}) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{z}\\ \boldsymbol{y}\end{bmatrix} |\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$

$$\boldsymbol{\mu} = \begin{bmatrix}\boldsymbol{\mu}_z \\ \boldsymbol{W}\boldsymbol{\mu}_z + \boldsymbol{b}\end{bmatrix} \qquad \boldsymbol{\Sigma} = \begin{bmatrix}\boldsymbol{\Sigma}_z & \boldsymbol{\Sigma}_z\boldsymbol{W}^T \\ \boldsymbol{W}\boldsymbol{\Sigma}_Z & \boldsymbol{\Sigma}_y + \boldsymbol{W}\boldsymbol{\Sigma}_z\boldsymbol{W}^T\end{bmatrix}$$

- The *posterior* distribution of $\boldsymbol{z}$ given $\boldsymbol{y}$

$$p(\boldsymbol{z}|\boldsymbol{y}) = \mathcal{N}\left(\boldsymbol{z}|\boldsymbol{\mu}_{z|y}, \boldsymbol{\Sigma}_{z|y}\right)$$
$$\boldsymbol{\Sigma}_{z|y}^{-1} = \boldsymbol{\Sigma}_z^{-1} + \boldsymbol{W}^T\boldsymbol{\Sigma}_y\boldsymbol{W}$$
$$\boldsymbol{\mu}_{z|y} = \boldsymbol{\Sigma}_{z|y}\left[\boldsymbol{W}^T\boldsymbol{\Sigma}_y^{-1}(\boldsymbol{y} - \boldsymbol{b}) + \boldsymbol{\Sigma}_z^{-1}\boldsymbol{\mu}_z\right]$$

- The *marginal* distribution $\boldsymbol{y}$

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{\mu}_z + \boldsymbol{b}, \boldsymbol{\Sigma}_y + \boldsymbol{W}\boldsymbol{\Sigma}_z\boldsymbol{W}^T)$$

Conditioning for multivariate Gaussians (Murphy1 Section 3.2.3)

- Suppose $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2)$ is jointly Gaussian $\mathcal{N}(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with mean and covariance

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

- The precision matrix $\boldsymbol{\Lambda}$

$$\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1} = \begin{bmatrix} \boldsymbol{\Lambda}_{11} & \boldsymbol{\Lambda}_{12} \\ \boldsymbol{\Lambda}_{21} & \boldsymbol{\Lambda}_{22} \end{bmatrix}$$

- The marginals are given by

$$\begin{aligned} p(\boldsymbol{y}_1) &= \mathcal{N}(\boldsymbol{y}_1|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{11}) \\ p(\boldsymbol{y}_2) &= \mathcal{N}(\boldsymbol{y}_1|\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_{22}) \end{aligned}$$

- The conditional

$$\begin{aligned} p(\boldsymbol{y}_1|\boldsymbol{y}_2) &= \mathcal{N}(\boldsymbol{y}_1|\boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2}) \\ \boldsymbol{\mu}_{1|2} &= \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{y}_2 - \boldsymbol{\mu}_2) \\ \boldsymbol{\Sigma}_{1|2} &= \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21} = \boldsymbol{\Lambda}_{11}^{-1} \end{aligned}$$

Gaussian process regression I

- Our model

$$y_n = f(\boldsymbol{x}_n) + e_n$$

Gaussian process regression I

- Our model

$$y_n = f(\boldsymbol{x}_n) + e_n$$

- Likelihood for all datapoints (assuming homoscedastic noise)

$$p(\boldsymbol{y}|\boldsymbol{f}) = \prod_{n=1}^{N} \mathcal{N}(y_n|f_n, \beta^{-1}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{f}, \beta^{-1}\boldsymbol{I})$$

Gaussian process regression I

- Our model

$$y_n = f(\boldsymbol{x}_n) + e_n$$

- Likelihood for all datapoints (assuming homoscedastic noise)

$$p(\boldsymbol{y}|\boldsymbol{f}) = \prod_{n=1}^{N} \mathcal{N}(y_n|f_n, \beta^{-1}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{f}, \beta^{-1}\boldsymbol{I})$$

- We impose a prior directly on the *function values* $\boldsymbol{f} = \begin{bmatrix} f(\boldsymbol{x}_1) & f(\boldsymbol{x}_2) & \dots & f(\boldsymbol{x}_N) \end{bmatrix}$

$$p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f}|\boldsymbol{0}, \boldsymbol{K}) \qquad \text{for} \qquad (\boldsymbol{K})_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

Gaussian process regression I

- Our model

$$y_n = f(\boldsymbol{x}_n) + e_n$$

- Likelihood for all datapoints (assuming homoscedastic noise)

$$p(\boldsymbol{y}|\boldsymbol{f}) = \prod_{n=1}^{N} \mathcal{N}(y_n|f_n, \beta^{-1}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{f}, \beta^{-1}\boldsymbol{I})$$

- We impose a prior directly on the *function values* $\boldsymbol{f} = \begin{bmatrix} f(\boldsymbol{x}_1) & f(\boldsymbol{x}_2) & \dots & f(\boldsymbol{x}_N) \end{bmatrix}$

$$p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f}|\boldsymbol{0}, \boldsymbol{K}) \qquad \text{for} \qquad (\boldsymbol{K})_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

- Goal: compute predictive distribution for $y^* = y(\boldsymbol{x}^*)$ given data $\boldsymbol{y}$, i.e. $p(y^*|\boldsymbol{y}, \boldsymbol{x}^*)$

Gaussian process regression I

- Our model

$$y_n = f(\boldsymbol{x}_n) + e_n$$

- Likelihood for all datapoints (assuming homoscedastic noise)

$$p(\boldsymbol{y}|\boldsymbol{f}) = \prod_{n=1}^{N} \mathcal{N}(y_n|f_n, \beta^{-1}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{f}, \beta^{-1}\boldsymbol{I})$$

- We impose a prior directly on the *function values* $\boldsymbol{f} = \begin{bmatrix} f(\boldsymbol{x}_1) & f(\boldsymbol{x}_2) & \dots & f(\boldsymbol{x}_N) \end{bmatrix}$

$$p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f}|\boldsymbol{0}, \boldsymbol{K}) \qquad \text{for} \qquad (\boldsymbol{K})_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

- Goal: compute predictive distribution for $y^* = y(\boldsymbol{x}^*)$ given data $\boldsymbol{y}$, i.e. $p(y^*|\boldsymbol{y}, \boldsymbol{x}^*)$

- Two-step strategy

  1. Calculate the joint Gaussian distribution $p(\boldsymbol{y}, y^*|\boldsymbol{x}^*)$
  2. Use rule for conditioning in Gaussian distributions to compute $p(y^*|\boldsymbol{y}, \boldsymbol{x}^*)$

Gaussian process regression II

- Recall: General Linear Gaussian systems (Murphy1 page 86-77).
  If

$$p(\mathbf{z}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_z, \Sigma_z)$$
$$p(\boldsymbol{y}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{z} + \boldsymbol{b}, \boldsymbol{\Sigma}_y)$$

  then

$$p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{z})p(\boldsymbol{z})\mathbf{d}\boldsymbol{z} = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{\mu}_z + \boldsymbol{b}, \boldsymbol{\Sigma}_y + \boldsymbol{W}\boldsymbol{\Sigma}_z\boldsymbol{W}^T)$$

- We can compute the marginal distribution of $\boldsymbol{y}$ using the *sum rule*

$$\begin{aligned} p(\boldsymbol{y}) &= \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f})\mathrm{d}\boldsymbol{f} \\ &= \int \mathcal{N}(\boldsymbol{y}|\boldsymbol{f}, \beta^{-1}\boldsymbol{I})\mathcal{N}(\boldsymbol{f}|\mathbf{0}, \boldsymbol{K})\mathrm{d}\boldsymbol{f} \\ &= \mathcal{N}(\boldsymbol{y}|\ ?\ ,\ ?\ ) \end{aligned}$$

- Spend 5 minutes calculating the mean and variance of $p(\boldsymbol{y})$

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\ ?\ ,\ ?\ )$$

Gaussian process regression II

- Recall: General Linear Gaussian systems (Murphy1 page 86-77).
  If

$$p(\mathbf{z}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_z, \Sigma_z)$$
$$p(\boldsymbol{y}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{z} + \boldsymbol{b}, \boldsymbol{\Sigma}_y)$$

  then

$$p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{z})p(\boldsymbol{z})\mathbf{d}\boldsymbol{z} = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{\mu}_z + \boldsymbol{b}, \boldsymbol{\Sigma}_y + \boldsymbol{W}\boldsymbol{\Sigma}_z\boldsymbol{W}^T)$$

- We can compute the marginal distribution of $\boldsymbol{y}$ using the *sum rule*

$$\begin{aligned} p(\boldsymbol{y}) &= \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f})\mathrm{d}\boldsymbol{f} \\ &= \int \mathcal{N}(\boldsymbol{y}|\boldsymbol{f}, \beta^{-1}\boldsymbol{I})\mathcal{N}(\boldsymbol{f}|\mathbf{0}, \boldsymbol{K})\mathrm{d}\boldsymbol{f} \\ &= \mathcal{N}(\boldsymbol{y}|\,?\,,\,?\,) \end{aligned}$$

- Spend 5 minutes calculating the mean and variance of $p(\boldsymbol{y})$

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\,?\,,\,?\,)$$

Gaussian process regression II

- Recall: General Linear Gaussian systems (Murphy1 page 86-77).
  If
  $$p(\mathbf{z}) = \mathcal{N}\left(\boldsymbol{z}|\boldsymbol{\mu}_z, \Sigma_z\right)$$
  $$p(\boldsymbol{y}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{z} + \boldsymbol{b}, \boldsymbol{\Sigma}_y)$$
  then
  $$p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{z})p(\boldsymbol{z})\mathbf{d}\boldsymbol{z} = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{\mu}_z + \boldsymbol{b}, \boldsymbol{\Sigma}_y + \boldsymbol{W}\boldsymbol{\Sigma}_z\boldsymbol{W}^T)$$

- We can compute the marginal distribution of $\boldsymbol{y}$ using the *sum rule*
  $$\begin{aligned} p(\boldsymbol{y}) &= \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f})\mathrm{d}\boldsymbol{f} \\ &= \int \mathcal{N}(\boldsymbol{y}|\boldsymbol{f}, \beta^{-1}\boldsymbol{I})\mathcal{N}(\boldsymbol{f}|\mathbf{0}, \boldsymbol{K})\mathrm{d}\boldsymbol{f} \\ &= \mathcal{N}(\boldsymbol{y}|\,?\,,\,?\,) \end{aligned}$$

- Spend 5 minutes calculating the mean and variance of $p(\boldsymbol{y})$
  $$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\mathbf{0}, \boldsymbol{K} + \beta^{-1}\boldsymbol{I})$$

Gaussian process regression II

- Recall: General Linear Gaussian systems (Murphy1 page 86-77).
  If
  $$p(\mathbf{z}) = \mathcal{N}\left(\boldsymbol{z}|\boldsymbol{\mu}_z, \Sigma_z\right)$$
  $$p(\boldsymbol{y}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{z} + \boldsymbol{b}, \boldsymbol{\Sigma}_y)$$
  then
  $$p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{z})p(\boldsymbol{z})\mathbf{d}\boldsymbol{z} = \mathcal{N}(\boldsymbol{y}|\boldsymbol{W}\boldsymbol{\mu}_z + \boldsymbol{b}, \boldsymbol{\Sigma}_y + \boldsymbol{W}\boldsymbol{\Sigma}_z\boldsymbol{W}^T)$$

- We can compute the marginal distribution of $\boldsymbol{y}$ using the *sum rule*
  $$\begin{aligned} p(\boldsymbol{y}) &= \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f})\mathrm{d}\boldsymbol{f} \\ &= \int \mathcal{N}(\boldsymbol{y}|\boldsymbol{f}, \beta^{-1}\boldsymbol{I})\mathcal{N}(\boldsymbol{f}|\mathbf{0}, \boldsymbol{K})\mathrm{d}\boldsymbol{f} \\ &= \mathcal{N}(\boldsymbol{y}|\,?\,,\,?\,) \end{aligned}$$

- Spend 5 minutes calculating the mean and variance of $p(\boldsymbol{y})$
  $$\begin{aligned} p(\boldsymbol{y}) &= \mathcal{N}(\boldsymbol{y}|\mathbf{0}, \boldsymbol{K} + \beta^{-1}\boldsymbol{I}) \\ &= \mathcal{N}(\boldsymbol{y}|\mathbf{0}, \boldsymbol{C}) \end{aligned}$$
  where
  $$\boldsymbol{C} = \boldsymbol{K} + \beta^{-1}\boldsymbol{I}$$

Gauasian process regression III

- The distribution of $\boldsymbol{y} \in \mathbf{R}^N$

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\mathbf{0}, \boldsymbol{C}) \qquad \text{for} \qquad \boldsymbol{C} = \boldsymbol{K} + \beta^{-1}\boldsymbol{I}$$

Suppose $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2)$ is jointly Gaussian $\mathcal{N}(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$

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

$$p(\boldsymbol{y}_1|\boldsymbol{y}_2) = \mathcal{N}(\boldsymbol{y}_1|\boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2})$$

$$\boldsymbol{\mu}_{1|2} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{y}_2 - \boldsymbol{\mu}_2)$$

$$\boldsymbol{\Sigma}_{1|2} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$$

Gauasian process regression III

- The distribution of $\boldsymbol{y} \in \mathbf{R}^N$

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\mathbf{0}, \boldsymbol{C}) \qquad \text{for} \qquad \boldsymbol{C} = \boldsymbol{K} + \beta^{-1}\boldsymbol{I}$$

- Let $\tilde{\boldsymbol{y}} = \begin{bmatrix} y(\boldsymbol{x}^*) & \boldsymbol{y} \end{bmatrix}^T \in \mathbb{R}^{N+1}$, then

Suppose $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2)$ is jointly Gaussian $\mathcal{N}(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$

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

$$p(\boldsymbol{y}_1|\boldsymbol{y}_2) = \mathcal{N}(\boldsymbol{y}_1|\boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2})$$

$$\boldsymbol{\mu}_{1|2} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{y}_2 - \boldsymbol{\mu}_2)$$

$$\boldsymbol{\Sigma}_{1|2} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$$

Gauasian process regression III

- The distribution of $\boldsymbol{y} \in \mathbf{R}^N$

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\mathbf{0}, \boldsymbol{C}) \qquad \text{for} \qquad \boldsymbol{C} = \boldsymbol{K} + \beta^{-1}\boldsymbol{I}$$

- Let $\tilde{\boldsymbol{y}} = \begin{bmatrix} y(\boldsymbol{x}^*) & \boldsymbol{y} \end{bmatrix}^T \in \mathbb{R}^{N+1}$, then

$$p(\tilde{\boldsymbol{y}}) = \mathcal{N}(\tilde{\boldsymbol{y}}|\mathbf{0}, \tilde{\boldsymbol{C}}) \qquad \text{for} \qquad \tilde{\boldsymbol{C}} = \begin{bmatrix} c & \boldsymbol{k} \\ \boldsymbol{k}^T & \boldsymbol{C} \end{bmatrix}$$

Suppose $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2)$ is jointly Gaussian $\mathcal{N}(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$

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

$$p(\boldsymbol{y}_1|\boldsymbol{y}_2) = \mathcal{N}(\boldsymbol{y}_1|\boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2})$$

$$\boldsymbol{\mu}_{1|2} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{y}_2 - \boldsymbol{\mu}_2)$$

$$\boldsymbol{\Sigma}_{1|2} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$$

Gauasian process regression III

- The distribution of $\boldsymbol{y} \in \mathbf{R}^N$

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\mathbf{0}, \boldsymbol{C}) \qquad \text{for} \qquad \boldsymbol{C} = \boldsymbol{K} + \beta^{-1}\boldsymbol{I}$$

- Let $\tilde{\boldsymbol{y}} = \begin{bmatrix} y(\boldsymbol{x}^*) & \boldsymbol{y} \end{bmatrix}^T \in \mathbb{R}^{N+1}$, then

$$p(\tilde{\boldsymbol{y}}) = \mathcal{N}(\tilde{\boldsymbol{y}}|\mathbf{0}, \tilde{\boldsymbol{C}}) \qquad \text{for} \qquad \tilde{\boldsymbol{C}} = \begin{bmatrix} c & \boldsymbol{k} \\ \boldsymbol{k}^T & \boldsymbol{C} \end{bmatrix}$$

where

$$c = k(\boldsymbol{x}^*, \boldsymbol{x}^*) + \beta^{-1}$$
$$\boldsymbol{k} = \begin{bmatrix} k(\boldsymbol{x}^*, \boldsymbol{x}_1) & k(\boldsymbol{x}^*, \boldsymbol{x}_2) & \dots & k(\boldsymbol{x}^*, \boldsymbol{x}_N) \end{bmatrix}$$

Suppose $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2)$ is jointly Gaussian $\mathcal{N}(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$

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

$$p(\boldsymbol{y}_1|\boldsymbol{y}_2) = \mathcal{N}(\boldsymbol{y}_1|\boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2})$$
$$\boldsymbol{\mu}_{1|2} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{y}_2 - \boldsymbol{\mu}_2)$$
$$\boldsymbol{\Sigma}_{1|2} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$$

Gauasian process regression III

- The distribution of $\boldsymbol{y} \in \mathbf{R}^N$

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\mathbf{0}, \boldsymbol{C}) \qquad \text{for} \qquad \boldsymbol{C} = \boldsymbol{K} + \beta^{-1}\boldsymbol{I}$$

- Let $\tilde{\boldsymbol{y}} = \begin{bmatrix} y(\boldsymbol{x}^*) & \boldsymbol{y} \end{bmatrix}^T \in \mathbb{R}^{N+1}$, then

$$p(\tilde{\boldsymbol{y}}) = \mathcal{N}(\tilde{\boldsymbol{y}}|\mathbf{0}, \tilde{\boldsymbol{C}}) \qquad \text{for} \qquad \tilde{\boldsymbol{C}} = \begin{bmatrix} c & \boldsymbol{k} \\ \boldsymbol{k}^T & \boldsymbol{C} \end{bmatrix}$$

where

$$c = k(\boldsymbol{x}^*, \boldsymbol{x}^*) + \beta^{-1}$$
$$\boldsymbol{k} = \begin{bmatrix} k(\boldsymbol{x}^*, \boldsymbol{x}_1) & k(\boldsymbol{x}^*, \boldsymbol{x}_2) & \dots & k(\boldsymbol{x}^*, \boldsymbol{x}_N) \end{bmatrix}$$

- What is the mean and variance for the following distribution?

$$p(y^*|\boldsymbol{y}) = \mathcal{N}(y^* \mid \mu_{y^*|\boldsymbol{y}} \, , \, \sigma^2_{y^*|\boldsymbol{y}})$$

Suppose $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2)$ is jointly Gaussian $\mathcal{N}(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$

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

$$p(\boldsymbol{y}_1|\boldsymbol{y}_2) = \mathcal{N}(\boldsymbol{y}_1|\boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2})$$
$$\boldsymbol{\mu}_{1|2} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{y}_2 - \boldsymbol{\mu}_2)$$
$$\boldsymbol{\Sigma}_{1|2} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$$

Gauasian process regression III

- The distribution of $\boldsymbol{y} \in \mathbf{R}^N$

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\mathbf{0}, \boldsymbol{C}) \qquad \text{for} \qquad \boldsymbol{C} = \boldsymbol{K} + \beta^{-1}\boldsymbol{I}$$

- Let $\tilde{\boldsymbol{y}} = \begin{bmatrix} y(\boldsymbol{x}^*) & \boldsymbol{y} \end{bmatrix}^T \in \mathbb{R}^{N+1}$, then

$$p(\tilde{\boldsymbol{y}}) = \mathcal{N}(\tilde{\boldsymbol{y}}|\mathbf{0}, \tilde{\boldsymbol{C}}) \qquad \text{for} \qquad \tilde{\boldsymbol{C}} = \begin{bmatrix} c & \boldsymbol{k} \\ \boldsymbol{k}^T & \boldsymbol{C} \end{bmatrix}$$

where

$$c = k(\boldsymbol{x}^*, \boldsymbol{x}^*) + \beta^{-1}$$
$$\boldsymbol{k} = \begin{bmatrix} k(\boldsymbol{x}^*, \boldsymbol{x}_1) & k(\boldsymbol{x}^*, \boldsymbol{x}_2) & \dots & k(\boldsymbol{x}^*, \boldsymbol{x}_N) \end{bmatrix}$$

- What is the mean and variance for the following distribution?

$$p(y^*|\boldsymbol{y}) = \mathcal{N}(y^* \mid \mu_{y^*|\boldsymbol{y}}\,,\ \sigma^2_{y^*|\boldsymbol{y}})$$

Suppose $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2)$ is jointly Gaussian $\mathcal{N}(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$

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

$$p(\boldsymbol{y}_1|\boldsymbol{y}_2) = \mathcal{N}(\boldsymbol{y}_1|\boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2})$$

$$\boldsymbol{\mu}_{1|2} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{y}_2 - \boldsymbol{\mu}_2)$$

$$\boldsymbol{\Sigma}_{1|2} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$$

Gauasian process regression III

- The distribution of $\boldsymbol{y} \in \mathbf{R}^N$

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\mathbf{0}, \boldsymbol{C}) \qquad \text{for} \qquad \boldsymbol{C} = \boldsymbol{K} + \beta^{-1}\boldsymbol{I}$$

- Let $\tilde{\boldsymbol{y}} = \begin{bmatrix} y(\boldsymbol{x}^*) & \boldsymbol{y} \end{bmatrix}^T \in \mathbb{R}^{N+1}$, then

$$p(\tilde{\boldsymbol{y}}) = \mathcal{N}(\tilde{\boldsymbol{y}}|\mathbf{0}, \tilde{\boldsymbol{C}}) \qquad \text{for} \qquad \tilde{\boldsymbol{C}} = \begin{bmatrix} c & \boldsymbol{k} \\ \boldsymbol{k}^T & \boldsymbol{C} \end{bmatrix}$$

where

$$c = k(\boldsymbol{x}^*, \boldsymbol{x}^*) + \beta^{-1}$$
$$\boldsymbol{k} = \begin{bmatrix} k(\boldsymbol{x}^*, \boldsymbol{x}_1) & k(\boldsymbol{x}^*, \boldsymbol{x}_2) & \dots & k(\boldsymbol{x}^*, \boldsymbol{x}_N) \end{bmatrix}$$

- What is the mean and variance for the following distribution?

$$p(y^*|\boldsymbol{y}) = \mathcal{N}(y^* \mid \mu_{y^*|\boldsymbol{y}}\,,\, \sigma^2_{y^*|\boldsymbol{y}})$$

$$\mu_{y^*|\boldsymbol{y}} = \boldsymbol{k}\boldsymbol{C}^{-1}\boldsymbol{y}$$
$$\sigma^2_{y^*|\boldsymbol{y}} = c - \boldsymbol{k}\boldsymbol{C}^{-1}\boldsymbol{k}^T$$

Suppose $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2)$ is jointly Gaussian $\mathcal{N}(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$

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

$$p(\boldsymbol{y}_1|\boldsymbol{y}_2) = \mathcal{N}(\boldsymbol{y}_1|\boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2})$$

$$\boldsymbol{\mu}_{1|2} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{y}_2 - \boldsymbol{\mu}_2)$$
$$\boldsymbol{\Sigma}_{1|2} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$$

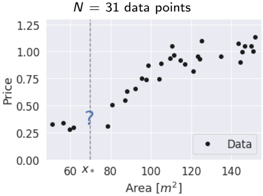
Example

Key equations for Gaussian process regression

$$p(y^*|\boldsymbol{y}) = \mathcal{N}\left(y^*\middle|\mu_{y^*|\boldsymbol{y}}, \sigma^2_{y^*|\boldsymbol{y}}\right)$$

$$\mu_{y^*|\boldsymbol{y}} = \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{y}$$

$$\sigma^2_{y^*|\boldsymbol{y}} = c - \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{k}^T$$



Example

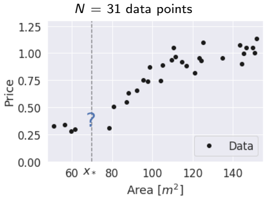
Key equations for Gaussian process regression

$$p(y^*|\mathbf{y}) = \mathcal{N}\left(y^* \middle| \mu_{y^*|\mathbf{y}}, \sigma^2_{y^*|\mathbf{y}}\right)$$

$$\mu_{y^*|\mathbf{y}} = \mathbf{k}\left(\mathbf{K} + \beta^{-1}\mathbf{I}\right)^{-1}\mathbf{y}$$

$$\sigma^2_{y^*|\mathbf{y}} = c - \mathbf{k}\left(\mathbf{K} + \beta^{-1}\mathbf{I}\right)^{-1}\mathbf{k}^T$$

- Predict $y^* = f(x_*) + e^*$ for test input $x_* = 70$



Example

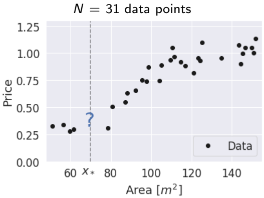
Key equations for Gaussian process regression

$$p(y^*|\boldsymbol{y}) = \mathcal{N}\left(y^*|\mu_{y^*|\boldsymbol{y}}, \sigma^2_{y^*|\boldsymbol{y}}\right)$$

$$\mu_{y^*|\boldsymbol{y}} = \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{y}$$

$$\sigma^2_{y^*|\boldsymbol{y}} = c - \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{k}^T$$

- Predict $y^* = f(x_*) + e^*$ for test input $x_* = 70$
- Observation vector $\boldsymbol{y} = [y_1, y_2, \dots, y_{31}]^T \in \mathbb{R}^{31\times 1}$



Example

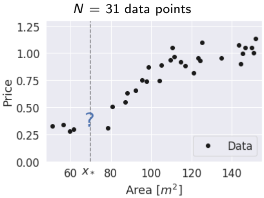
Key equations for Gaussian process regression

$$p(y^*|\boldsymbol{y}) = \mathcal{N}\left(y^* \middle| \mu_{y^*|\boldsymbol{y}}, \sigma^2_{y^*|\boldsymbol{y}}\right)$$

$$\mu_{y^*|\boldsymbol{y}} = \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{y}$$

$$\sigma^2_{y^*|\boldsymbol{y}} = c - \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{k}^T$$

- Predict $y^* = f(x_*) + e^*$ for test input $x_* = 70$
- Observation vector $\boldsymbol{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31\times 1}$
- $k(x, x') = \text{cov}(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2\cdot 20^2}\right]$



Example

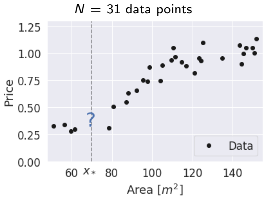
Key equations for Gaussian process regression

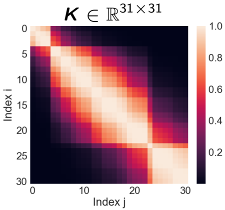
$$p(y^*|\boldsymbol{y}) = \mathcal{N}\left(y^*|\mu_{y^*|\boldsymbol{y}}, \sigma^2_{y^*|\boldsymbol{y}}\right)$$

$$\mu_{y^*|\boldsymbol{y}} = \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{y}$$

$$\sigma^2_{y^*|\boldsymbol{y}} = c - \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{k}^T$$

- Predict $y^* = f(x_*) + e^*$ for test input $x_* = 70$
- Observation vector $\boldsymbol{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31\times 1}$
- $k(x, x') = \mathrm{cov}(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2\cdot 20^2}\right]$
- Covariance matrix for training data: $[\boldsymbol{K}]_{ij} = k(x_i, x_j)$





Example

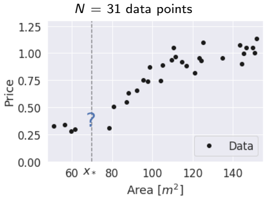
Key equations for Gaussian process regression

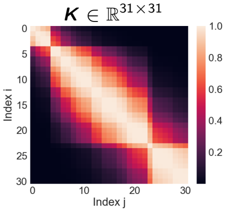
$$p(y^*|\boldsymbol{y}) = \mathcal{N}\left(y^*|\mu_{y^*|\boldsymbol{y}}, \sigma^2_{y^*|\boldsymbol{y}}\right)$$

$$\mu_{y^*|\boldsymbol{y}} = \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{y}$$

$$\sigma^2_{y^*|\boldsymbol{y}} = c - \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{k}^T$$

- Predict $y^* = f(x_*) + e^*$ for test input $x_* = 70$
- Observation vector $\boldsymbol{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31\times 1}$
- $k(x, x') = \text{cov}(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2\cdot 20^2}\right]$
- Covariance matrix for training data: $[\boldsymbol{K}]_{ij} = k(x_i, x_j)$
- Cov. between test and training $[\boldsymbol{k}]_j = k(x_*, x_j)$





Example

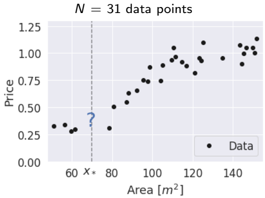
Key equations for Gaussian process regression

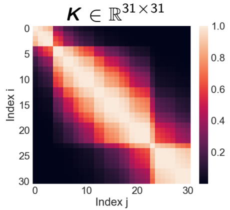
$$p(y^*|\mathbf{y}) = \mathcal{N}\left(y^* \middle| \mu_{y^*|\mathbf{y}}, \sigma^2_{y^*|\mathbf{y}}\right)$$

$$\mu_{y^*|\mathbf{y}} = \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{y}$$

$$\sigma^2_{y^*|\mathbf{y}} = c - \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{k}^T$$

- Predict $y^* = f(x_*) + e^*$ for test input $x_* = 70$
- Observation vector $\boldsymbol{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31\times 1}$
- $k(x, x') = \text{cov}(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2\cdot 20^2}\right]$
- Covariance matrix for training data: $[\boldsymbol{K}]_{ij} = k(x_i, x_j)$
- Cov. between test and training $[\boldsymbol{k}]_j = k(x_*, x_j)$
- Covariance of test point $y^* = y(x_*)$: $c = k(x_*, x_*) + \beta^{-1}$





Example

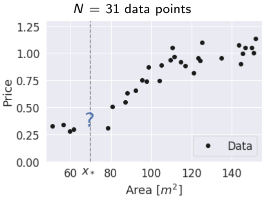
Key equations for Gaussian process regression

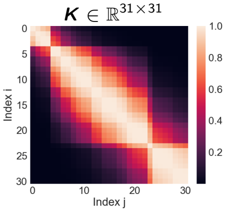
$$p(y^*|\mathbf{y}) = \mathcal{N}\left(y^* | \mu_{y^*|\mathbf{y}}, \sigma^2_{y^*|\mathbf{y}}\right)$$

$$\mu_{y^*|\mathbf{y}} = \mathbf{k}\left(\mathbf{K} + \beta^{-1}\mathbf{I}\right)^{-1}\mathbf{y}$$

$$\sigma^2_{y^*|\mathbf{y}} = c - \mathbf{k}\left(\mathbf{K} + \beta^{-1}\mathbf{I}\right)^{-1}\mathbf{k}^T$$

- Predict $y^* = f(x_*) + e^*$ for test input $x_* = 70$
- Observation vector $\mathbf{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31\times 1}$
- $k(x, x') = \text{cov}(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2\cdot 20^2}\right]$
- Covariance matrix for training data: $[\mathbf{K}]_{ij} = k(x_i, x_j)$
- Cov. between test and training $[\mathbf{k}]_j = k(x_*, x_j)$
- Covariance of test point $y^* = y(x_*)$: $c = k(x_*, x_*) + \beta^{-1}$
- Now we have all the ingredients for the key equations





Example

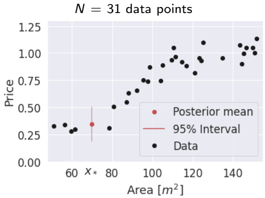
Key equations for Gaussian process regression

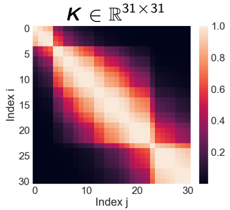
$$p(y^*|\mathbf{y}) = \mathcal{N}\left(y^*|\mu_{y^*|\mathbf{y}}, \sigma^2_{y^*|\mathbf{y}}\right)$$

$$\mu_{y^*|\mathbf{y}} = \mathbf{k}\left(\mathbf{K} + \beta^{-1}\mathbf{I}\right)^{-1}\mathbf{y}$$

$$\sigma^2_{y^*|\mathbf{y}} = c - \mathbf{k}\left(\mathbf{K} + \beta^{-1}\mathbf{I}\right)^{-1}\mathbf{k}^T$$

- Predict $y^* = f(x_*) + e^*$ for test input $x_* = 70$
- Observation vector $\mathbf{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31\times 1}$
- $k(x, x') = \text{cov}(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2\cdot 20^2}\right]$
- Covariance matrix for training data: $[\mathbf{K}]_{ij} = k(x_i, x_j)$
- Cov. between test and training $[\mathbf{k}]_j = k(x_*, x_j)$
- Covariance of test point $y^* = y(x_*)$: $c = k(x_*, x_*) + \beta^{-1}$
- Now we have all the ingredients for the key equations





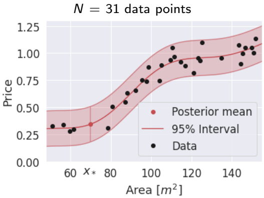
Example

Key equations for Gaussian process regression

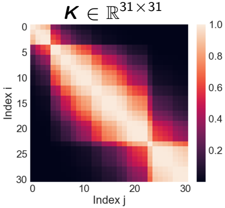
$$p(y^*|\boldsymbol{y}) = \mathcal{N}\left(y^*|\mu_{y^*|\boldsymbol{y}}, \sigma^2_{y^*|\boldsymbol{y}}\right)$$

$$\mu_{y^*|\boldsymbol{y}} = \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{y}$$

$$\sigma^2_{y^*|\boldsymbol{y}} = c - \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{k}^T$$



- Predict $y^* = f(x_*) + e^*$ for test input $x_* = 70$
- Observation vector $\boldsymbol{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31\times 1}$
- $k(x, x') = \mathrm{cov}(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2\cdot 20^2}\right]$



- Covariance matrix for training data: $[\boldsymbol{K}]_{ij} = k(x_i, x_j)$
- Cov. between test and training $[\boldsymbol{k}]_j = k(x_*, x_j)$
- Covariance of test point $y^* = y(x_*)$: $c = k(x_*, x_*) + \beta^{-1}$

- Now we have all the ingredients for the key equations

Gaussian process intuition

- Gaussian process implements the assumption

$$\boldsymbol{x} \approx \boldsymbol{x}' \quad \Rightarrow \quad f(\boldsymbol{x}) \approx f(\boldsymbol{x}')$$

- In words: If the inputs are similar, the outputs should be similar as well.

- Using the squared exponential covariance function as example

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

- Then covariance between $f(\boldsymbol{x})$ and $f(\boldsymbol{x})'$ is given by

$$\mathrm{cov}\left[f(\boldsymbol{x}), f(\boldsymbol{x}')\right] = k(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2}\right)$$

- Note: the covariance between outputs are given in terms of the inputs

True or false?

Key equations for Gaussian process regression

$$p(y^*|\boldsymbol{y}) = \mathcal{N}\left(y^* \middle| \mu_{y^*|\boldsymbol{y}}, \sigma^2_{y^*|\boldsymbol{y}}\right)$$

$$\mu_{y^*|\boldsymbol{y}} = \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{y}$$

$$\sigma^2_{y^*|\boldsymbol{y}} = c - \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{k}^T$$

True or false?

- Spend 5 minutes on the DTU Learn quiz: "Lecture 5: Key equations for GP Regression."

Covariance functions

Covariance functions

- A covariance function $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ maps a pair of inputs $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}$ from some input space $\mathcal{X}$ to the real line $\mathbb{R}$

- Recall: the covariance / kernel matrix is given by

$$\boldsymbol{K}_{ij} = \operatorname{cov}\left(f(\boldsymbol{x}_i), f(\boldsymbol{x})_j\right) = k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$$

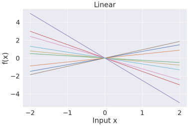
- Covariance functions must be symmetric & Positive Semi-Definite such that

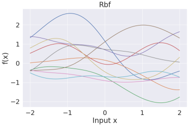
$$\text{(Symmetric)} \quad \boldsymbol{K} = \boldsymbol{K}^T$$

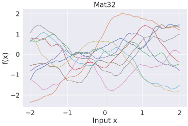
$$\text{(PSD)} \quad \forall \boldsymbol{x} \neq 0 : \quad \boldsymbol{x}^T \boldsymbol{K} \boldsymbol{x} \geq 0$$

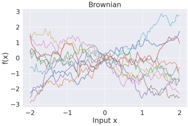
- Must hold for all possible data sets $\{\boldsymbol{x}_n\}_{n=1}^N \subset \mathcal{X}$ in the input space $\mathcal{X}$

- Covariance functions as prior information





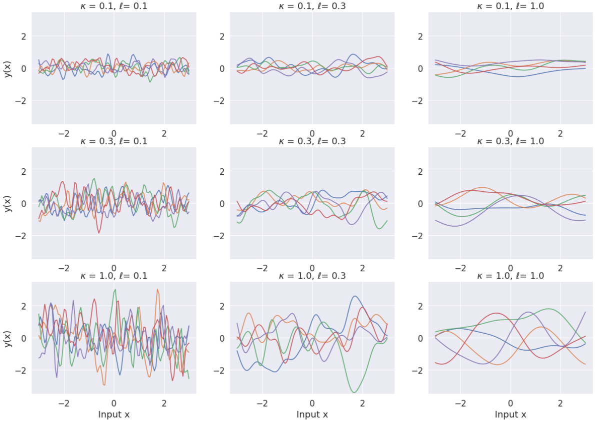




The squared exponential kernel - prior samples

$$k(\boldsymbol{x}, \boldsymbol{x}') = \kappa^2 \exp\left[-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|_2^2}{2\ell^2}\right]$$

Parameter ℓ is the called the *lengthscale* and parameter κ is the called the *magnitude*



Constructing new kernels from old ones

Techniques for Constructing New Kernels.

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

$$\begin{aligned}
k(\mathbf{x}, \mathbf{x}') &= ck_1(\mathbf{x}, \mathbf{x}') && (6.13)\\
k(\mathbf{x}, \mathbf{x}') &= f(\mathbf{x})k_1(\mathbf{x}, \mathbf{x}')f(\mathbf{x}') && (6.14)\\
k(\mathbf{x}, \mathbf{x}') &= q\left(k_1(\mathbf{x}, \mathbf{x}')\right) && (6.15)\\
k(\mathbf{x}, \mathbf{x}') &= \exp\left(k_1(\mathbf{x}, \mathbf{x}')\right) && (6.16)\\
k(\mathbf{x}, \mathbf{x}') &= k_1(\mathbf{x}, \mathbf{x}') + k_2(\mathbf{x}, \mathbf{x}') && (6.17)\\
k(\mathbf{x}, \mathbf{x}') &= k_1(\mathbf{x}, \mathbf{x}')k_2(\mathbf{x}, \mathbf{x}') && (6.18)\\
k(\mathbf{x}, \mathbf{x}') &= k_3\left(\boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{x}')\right) && (6.19)\\
k(\mathbf{x}, \mathbf{x}') &= \mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{x}' && (6.20)\\
k(\mathbf{x}, \mathbf{x}') &= k_a(\mathbf{x}_a, \mathbf{x}_a') + k_b(\mathbf{x}_b, \mathbf{x}_b') && (6.21)\\
k(\mathbf{x}, \mathbf{x}') &= k_a(\mathbf{x}_a, \mathbf{x}_a')k_b(\mathbf{x}_b, \mathbf{x}_b') && (6.22)
\end{aligned}$$

where $c > 0$ is a constant, $f(\cdot)$ is any function, $q(\cdot)$ is a polynomial with nonnegative coefficients, $\boldsymbol{\phi}(\mathbf{x})$ is a function from $\mathbf{x}$ to $\mathbb{R}^M$, $k_3(\cdot, \cdot)$ is a valid kernel in $\mathbb{R}^M$, $\mathbf{A}$ is a symmetric positive semidefinite matrix, $\mathbf{x}_a$ and $\mathbf{x}_b$ are variables (not necessarily disjoint) with $\mathbf{x} = (\mathbf{x}_a, \mathbf{x}_b)$, and k_a and k_b are valid kernel functions over their respective spaces.

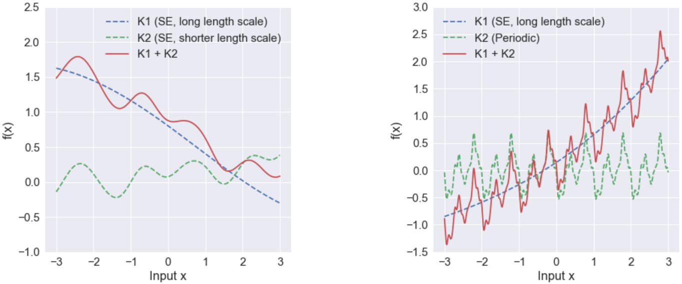
From Chris Bishops book: `https://www.microsoft.com/en-us/research/people/cmbishop`

Additive kernels

- Adding two SEs kernels to model long term trends (long length scale) and short term fluctuations (short length scale)

$$k(\boldsymbol{x}, \boldsymbol{x}') = \kappa_1^2 \exp\left[-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|_2^2}{2\ell_1^2}\right] + \kappa_2^2 \exp\left[-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|_2^2}{2\ell_2^2}\right]$$

- Adding SE and period kernels to model long term trends (long length scale) and periodic fluctuations



Hyperparameters and the marginal likelihood

The marginal likelihood I

- Let $\boldsymbol{\theta}$ denote all hyperparameters, then marginal likelihood for Gaussian likelihood

$$
\begin{aligned}
p(\boldsymbol{y}\big|\boldsymbol{\theta}) &= \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}\big|\boldsymbol{\theta}_K)\mathrm{d}\boldsymbol{f} \\
&= \int \mathcal{N}\left(\boldsymbol{y}\big|\boldsymbol{f}, \beta^{-1}\boldsymbol{I}\right)\mathcal{N}\left(\boldsymbol{f}\big|\boldsymbol{0}, \boldsymbol{K}\right)\mathrm{d}\boldsymbol{f} \\
&= \mathcal{N}\left(\boldsymbol{y}\big|\boldsymbol{0}, \beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)
\end{aligned}
$$

- We can tune the hyperparameters of the model by optimizing the marginal likelihood as we did for linear regresssion

  1. Hyperparameters of the likelihood (e.g β or σ)
  2. Hyperparameters of the kernel $\boldsymbol{\theta}_K$ (e.g. lengthscales and magnitudes)

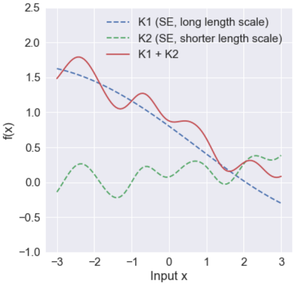
- In practice, we compute the gradient of $p(\boldsymbol{y}\big|\boldsymbol{\theta})$ wrt. $\boldsymbol{\theta}$ and use numerical optimization

The marginal likelihood II

- Suppose we have 5 hyperparameters in total

$$\boldsymbol{\theta} = \{\sigma, \kappa_1, \ell_1, \kappa_2, \ell_2\}$$

- Suppose we want to estimate those using 10-fold cross-validation and test out 10 values for each hyperparameter. How many times do we need to train the model?



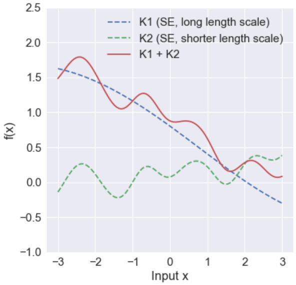
The marginal likelihood II

- Suppose we have 5 hyperparameters in total

$$\boldsymbol{\theta} = \{\sigma, \kappa_1, \ell_1, \kappa_2, \ell_2\}$$

- Suppose we want to estimate those using 10-fold cross-validation and test out 10 values for each hyperparameter. How many times do we need to train the model?

$$10 \cdot 10^5 = 10^6$$



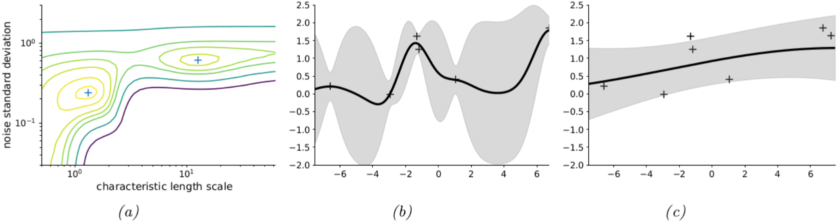
The marginal likelihood III

- The gradients of the marginal likelihood wrt. hyperparameters are given by

$$\frac{\partial}{\partial \theta_j} \log p(\boldsymbol{y}|\boldsymbol{\theta}) = \frac{1}{2}\mathrm{tr}\left((\boldsymbol{\alpha}\boldsymbol{\alpha}^T - \boldsymbol{K}^{-1}) \frac{\partial \boldsymbol{K}}{\partial \theta_j} \right),$$

  where $\boldsymbol{\alpha} = \boldsymbol{K}^{-1}\boldsymbol{y}$ and $\frac{\partial \boldsymbol{K}}{\partial \theta_j}$ depends on the specific choice of kernel.

- $\log p(\boldsymbol{y}|\boldsymbol{\theta})$ is also multimodal wrt. $\boldsymbol{\theta}$



From the Murphy1 book (p. 578)

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$

- Step 1: Compute Cholesky factorization of $\boldsymbol{C} = \beta^{-1}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute Cholesky factorization of $\boldsymbol{C} = \beta^{-1}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| =$$

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute Cholesky factorization of $\boldsymbol{C} = \beta^{-1}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right|$$

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$

- Step 1: Compute Cholesky factorization of $\boldsymbol{C} = \beta^{-1}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$

- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right| \cdot \left|\boldsymbol{L}^T\right|$$

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$

- Step 1: Compute Cholesky factorization of $\boldsymbol{C} = \beta^{-1}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$

- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right|$$

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$

- Step 1: Compute Cholesky factorization of $\boldsymbol{C} = \beta^{-1}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$

- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right| \cdot \left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn}$$

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$

- Step 1: Compute Cholesky factorization of $\boldsymbol{C} = \beta^{-1}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$

- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} = 2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn}$$

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute Cholesky factorization of $\boldsymbol{C} = \beta^{-1}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} = 2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn}$$

- Step 3: Compute quadractic term as follows

$$\boldsymbol{y}^T\boldsymbol{C}^{-1}\boldsymbol{y} =$$

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute Cholesky factorization of $\boldsymbol{C} = \beta^{-1}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} = 2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn}$$

- Step 3: Compute quadractic term as follows

$$\boldsymbol{y}^T\boldsymbol{C}^{-1}\boldsymbol{y} = \boldsymbol{y}^T\left(\boldsymbol{L}\boldsymbol{L}^T\right)^{-1}\boldsymbol{y}$$

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$

- Step 1: Compute Cholesky factorization of $\boldsymbol{C} = \beta^{-1}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$

- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} = 2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn}$$

- Step 3: Compute quadractic term as follows

$$\boldsymbol{y}^T\boldsymbol{C}^{-1}\boldsymbol{y} = \boldsymbol{y}^T\left(\boldsymbol{L}\boldsymbol{L}^T\right)^{-1}\boldsymbol{y} = \boldsymbol{y}^T\boldsymbol{L}^{-T}\boldsymbol{L}^{-1}\boldsymbol{y}$$

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\mathbf{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\mathbf{I} + \mathbf{K}\right| - \frac{1}{2}\mathbf{y}^T\left(\beta^{-1}\mathbf{I} + \mathbf{K}\right)^{-1}\mathbf{y}$$

- In numpy: $\left|0.1\mathbf{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\mathbf{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$

- Step 1: Compute Cholesky factorization of $\mathbf{C} = \beta^{-1}\mathbf{I} + \mathbf{K}$ such that $\mathbf{C} = \mathbf{L}\mathbf{L}^T$

- Step 2: Compute the log determinant term as follows

$$\ln\left|\mathbf{C}\right| = \ln\left|\mathbf{L}\mathbf{L}^T\right| = \ln\left|\mathbf{L}\right|\cdot\left|\mathbf{L}^T\right| = \ln\left|\mathbf{L}\right|^2 = 2\ln\left|\mathbf{L}\right| = 2\ln\prod_{n=1}^{N}\mathbf{L}_{nn} = 2\sum_{n=1}^{N}\ln\mathbf{L}_{nn}$$

- Step 3: Compute quadractic term as follows

$$\mathbf{y}^T\mathbf{C}^{-1}\mathbf{y} = \mathbf{y}^T\left(\mathbf{L}\mathbf{L}^T\right)^{-1}\mathbf{y} = \mathbf{y}^T\mathbf{L}^{-T}\mathbf{L}^{-1}\mathbf{y} = \left(\mathbf{L}^{-1}\mathbf{y}\right)^T\underbrace{\left(\mathbf{L}^{-1}\mathbf{y}\right)}_{=\mathbf{v}} = \mathbf{v}^T\mathbf{v}$$

The marginal likelihood: numerics

- In practice, we should avoid computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\beta^{-1}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$

- Step 1: Compute Cholesky factorization of $\boldsymbol{C} = \beta^{-1}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$

- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} = 2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn}$$

- Step 3: Compute quadractic term as follows

$$\boldsymbol{y}^T\boldsymbol{C}^{-1}\boldsymbol{y} = \boldsymbol{y}^T\left(\boldsymbol{L}\boldsymbol{L}^T\right)^{-1}\boldsymbol{y} = \boldsymbol{y}^T\boldsymbol{L}^{-T}\boldsymbol{L}^{-1}\boldsymbol{y} = \left(\boldsymbol{L}^{-1}\boldsymbol{y}\right)^T\underbrace{\left(\boldsymbol{L}^{-1}\boldsymbol{y}\right)}_{=\boldsymbol{v}} = \boldsymbol{v}^T\boldsymbol{v}$$

- Step 4: Sum components

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn} - \frac{1}{2}\boldsymbol{v}^T\boldsymbol{v}$$

- Note that we never compute the determinant or the inverse of $\boldsymbol{C}$ directly!

Computational complexity of Gaussian Processes

Key equations for Gaussian process regression

$$p(y^*|\boldsymbol{y}) = \mathcal{N}\left(y^* \middle| \mu_{y^*|\boldsymbol{y}}, \sigma^2_{y^*|\boldsymbol{y}}\right)$$
$$\mu_{y^*|\boldsymbol{y}} = \boldsymbol{k}\left(\boldsymbol{K} + \beta^{-1}\boldsymbol{I}\right)^{-1}\boldsymbol{y}$$
$$\sigma^2_{y^*|\boldsymbol{y}} = c - \boldsymbol{k}\left(\boldsymbol{K} + \sigma^2\boldsymbol{I}\right)^{-1}\boldsymbol{k}^T$$

- Gaussian processes are *non-parametric models*
- Recall: If $\boldsymbol{A} \in \mathbb{R}^{N\times M}$ and $\boldsymbol{b} \in \mathbb{R}^M$, then the cost of computing $\boldsymbol{Ab}$ is $\mathcal{O}(NM)$
- Recall: If $\boldsymbol{C} \in \mathbb{R}^{N\times N}$, then the cost of computing $\boldsymbol{C}^{-1}$ is $\mathcal{O}\left(N^3\right)$
- What is computational complexity for computing the posterior distribution for 1 test point based on a data set with N observations? $\mathcal{O}(N^3)$
- What about the memory footprint? $\mathcal{O}(N^2)$