Advanced Statistical Inference
Gaussian Processes

Maurizio Filippone
Maurizio.Filippone@eurecom.fr

Department of Data Science
EURECOM

Suggested readings

Gaussian Processes for Machine Learning

Carl E. Rasmussen and Christopher K. I. Williams

Pattern Recognition and Machine Learning

C. Bishop

Gaussian Processes

- Linear models requires specifying a set of basis functions
  - Polynomials, Trigonometric, ...??

Gaussian Processes

- Linear models requires specifying a set of basis functions
  - Polynomials, Trigonometric, . . .??
- Can we use Bayesian inference to let data tell us this?

Gaussian Processes

- Linear models requires specifying a set of basis functions
  - Polynomials, Trigonometric, ...??
- Can we use Bayesian inference to let data tell us this?
- Gaussian Processes work implicitly with an infinite set of basis functions and learn a probabilistic combination of these

Gaussian Processes

Gaussian Processes can be explained in two ways

- Weight Space View
  - Bayesian linear regression with infinite basis functions
- Function Space View
  - Defined as priors over functions

Gaussian Processes

Gaussian Processes can be explained in two ways

- **Weight Space View**
  - **Bayesian linear regression with infinite basis functions**
- Function Space View
  - Defined as priors over functions

Bayesian Linear Regression - recap

- Modeling observations as noisy realizations of a linear combination of the features:

$$p(\mathbf{t}|\mathbf{w}, \mathbf{X}, \sigma^2) = \mathcal{N}(\mathbf{X}\mathbf{w}, \sigma^2\mathbf{I})$$

Bayesian Linear Regression - recap

- Modeling observations as noisy realizations of a linear combination of the features:

$$p(\mathbf{t}|\mathbf{w}, \mathbf{X}, \sigma^2) = \mathcal{N}(\mathbf{X}\mathbf{w}, \sigma^2\mathbf{I})$$

- Gaussian prior over model parameters:

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{0}, \mathbf{S})$$

Bayesian Linear Regression - recap

- Posterior **must be** Gaussian

$$p(\mathbf{w}|\mathbf{X}, \mathbf{t}, \sigma^2) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

- Covariance:

$$\boldsymbol{\Sigma} = \left(\frac{1}{\sigma^2} \mathbf{X}^{\mathsf{T}} \mathbf{X} + \mathbf{S}^{-1} \right)^{-1}$$

- Mean:

$$\boldsymbol{\mu} = \frac{1}{\sigma^2} \boldsymbol{\Sigma} \mathbf{X}^{\mathsf{T}} \mathbf{t}$$

- Predictions

$$p(t_*|\mathbf{X}, \mathbf{t}, \mathbf{x}_*, \sigma^2) = \mathcal{N}(\mathbf{x}_*^{\mathsf{T}} \boldsymbol{\mu}, \sigma^2 + \mathbf{x}_*^{\mathsf{T}} \boldsymbol{\Sigma} \mathbf{x}_*)$$

Introducing basis functions

- Imagine transforming the inputs using a set of D functions

$$\mathbf{x} \rightarrow \boldsymbol{\phi}(\mathbf{x}) = (\phi_1(\mathbf{x}), \ldots, \phi_D(\mathbf{x}))^\top$$

- The functions $\phi_1(\mathbf{x})$ are also known as basis functions
- Define:

$$\boldsymbol{\Phi} = \begin{bmatrix} \phi_1(\mathbf{x}_1) & \ldots & \phi_D(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \phi_1(\mathbf{x}_N) & \ldots & \phi_D(\mathbf{x}_N) \end{bmatrix}$$

Introducing basis functions

- Applying Bayesian Linear Regression on the transformed features gives

$$p(\mathbf{w}|\mathbf{X}, \mathbf{t}, \sigma^2) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

- Covariance:

$$\boldsymbol{\Sigma} = \left(\frac{1}{\sigma^2} \boldsymbol{\Phi}^{\mathsf{T}} \boldsymbol{\Phi} + \mathbf{S}^{-1} \right)^{-1}$$

- Mean:

$$\boldsymbol{\mu} = \frac{1}{\sigma^2} \boldsymbol{\Sigma} \boldsymbol{\Phi}^{\mathsf{T}} \mathbf{t}$$

- Predictions:

$$p(t_*|\mathbf{X}, \mathbf{t}, \mathbf{x}_*, \sigma^2) = \mathcal{N}(\phi_*^{\mathsf{T}} \boldsymbol{\mu}, \sigma^2 + \phi_*^{\mathsf{T}} \boldsymbol{\Sigma} \phi_*)$$

Bayesian Linear Regression as a Kernel Machine

- We are going to show that predictions can be expressed exclusively in terms of scalar products as follows

$$k(\mathbf{x}, \mathbf{x}') = \boldsymbol{\psi}(\mathbf{x})^\top \boldsymbol{\psi}(\mathbf{x}')$$

- This allows us to work with either $k(\cdot, \cdot)$ or $\boldsymbol{\psi}(\cdot)$
- Why is this useful??

Bayesian Linear Regression as a Kernel Machine

- Working with $\psi(\cdot)$ costs $O(D^2)$ storage, $O(D^3)$ time
- Working with $k(\cdot, \cdot)$ costs $O(N^2)$ storage, $O(N^3)$ time

Bayesian Linear Regression as a Kernel Machine

- Working with $\psi(\cdot)$ costs $O(D^2)$ storage, $O(D^3)$ time
- Working with $k(\cdot, \cdot)$ costs $O(N^2)$ storage, $O(N^3)$ time
- Pick the one that makes computations faster . . . or

Bayesian Linear Regression as a Kernel Machine

- Working with $\psi(\cdot)$ costs $O(D^2)$ storage, $O(D^3)$ time
- Working with $k(\cdot,\cdot)$ costs $O(N^2)$ storage, $O(N^3)$ time
- Pick the one that makes computations faster … or
- What if we could pick $k(\cdot,\cdot)$ so that $\psi(\cdot)$ is infinite dimensional?

Kernels

- It is possible to show that for

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

  there exists a corresponding $\psi(\cdot)$ that is infinite dimensional!!!
- There are other kernels satisfying this property

Kernels

Proof that the Gaussian kernel induces an infinite dimensional $\psi(\cdot)$

- For simplicity consider one dimensional inputs x, y
- Expand the Gaussian kernel $k(x, y)$ as

$$\exp\left(-\frac{(x-y)^2}{2}\right) = \exp\left(-\frac{x^2}{2}\right)\exp\left(-\frac{y^2}{2}\right)\exp(xy)$$

- Focusing on the last term and applying the Taylor expansion of the $\exp(\cdot)$ function

$$\exp(xy) = 1 + (xy) + \frac{(xy)^2}{2!} + \frac{(xy)^3}{3!} + \frac{(xy)^4}{4!} + \dots$$

Kernels

Proof that the Gaussian kernel induces an infinite dimensional $\psi(\cdot)$

- Define the infinite dimensional mapping

$$\psi(x) = \exp\left(-\frac{x^2}{2}\right)\left(1, x, \frac{x^2}{\sqrt{2!}}, \frac{x^3}{\sqrt{3!}}, \frac{x^4}{\sqrt{4!}}, \ldots\right)^\top$$

- It is easy to verify that

$$k(x, y) = \exp\left(-\frac{(x-y)^2}{2}\right) = \psi(x)^\top \psi(y)$$

Bayesian Linear Regression as a Kernel Machine

Proof

- To show that Bayesian Linear Regression can be formulated through scalar products only, we need Woodbury identity:

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1}$$

- Do not memorize this!

Bayesian Linear Regression as a Kernel Machine

Proof

- Woodbury identity:

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1}$$

- We can rewrite:

$$\begin{aligned} \mathbf{\Sigma} &= \left(\frac{1}{\sigma^2}\mathbf{\Phi}^\mathsf{T}\mathbf{\Phi} + \mathbf{S}^{-1}\right)^{-1} \\ &= \mathbf{S} - \mathbf{S}\mathbf{\Phi}^\mathsf{T}\left(\sigma^2\mathbf{I} + \mathbf{\Phi}\mathbf{S}\mathbf{\Phi}^\mathsf{T}\right)^{-1}\mathbf{\Phi}\mathbf{S} \end{aligned}$$

- We set $A = \mathbf{S}$, $U = V^\top = \mathbf{\Phi}^\mathsf{T}$, and $C = \frac{1}{\sigma^2}\mathbf{I}$

Bayesian Linear Regression as a Kernel Machine

Proof

- Mean and variance of the predictions:

$$p(t_*|\mathbf{X}, \mathbf{t}, \mathbf{x}_*, \sigma^2) = \mathcal{N}(\phi_*^{\mathsf{T}}\boldsymbol{\mu}, \sigma^2 + \phi_*^{\mathsf{T}}\boldsymbol{\Sigma}\phi_*)$$

- Rewrite the variance:

$$\begin{aligned} \sigma^2 \quad &+ \quad \phi_*^{\mathsf{T}}\boldsymbol{\Sigma}\phi_* = \\ \sigma^2 \quad &+ \quad \phi_*^{\mathsf{T}}\mathbf{S}\phi_* - \phi_*^{\mathsf{T}}\mathbf{S}\boldsymbol{\Phi}^{\mathsf{T}}\left(\sigma^2\mathbf{I} + \boldsymbol{\Phi}\mathbf{S}\boldsymbol{\Phi}^{\mathsf{T}}\right)^{-1}\boldsymbol{\Phi}\mathbf{S}\phi_* \end{aligned}$$

. . . continued

Bayesian Linear Regression as a Kernel Machine

Proof

- Mean and variance of the predictions:

$$p(t_* | \mathbf{X}, \mathbf{t}, \mathbf{x}_*, \sigma^2) = \mathcal{N}(\phi_*^{\mathsf{T}} \boldsymbol{\mu}, \sigma^2 + \phi_*^{\mathsf{T}} \boldsymbol{\Sigma} \phi_*)$$

- Rewrite the variance:

$$\sigma^2 \quad + \quad \phi_*^{\mathsf{T}} \mathbf{S} \phi_* - \phi_*^{\mathsf{T}} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}} \left(\sigma^2 \mathbf{I} + \boldsymbol{\Phi} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}} \right)^{-1} \boldsymbol{\Phi} \mathbf{S} \phi_* =$$

$$\sigma^2 \quad + \quad k_{**} - \mathbf{k}_*^{\top} \left(\sigma^2 \mathbf{I} + \mathbf{K} \right)^{-1} \mathbf{k}_*$$

- Where the mapping defining the kernel is

$$\psi(\mathbf{x}) = \mathbf{S}^{1/2} \phi(\mathbf{x})$$

and

$$\begin{aligned} k_{**} &= k(\mathbf{x}_*, \mathbf{x}_*) = \psi(\mathbf{x}_*)^{\mathsf{T}} \psi(\mathbf{x}_*) \\ (\mathbf{k}_*)_i &= k(\mathbf{x}_*, \mathbf{x}_i) = \psi(\mathbf{x}_*)^{\mathsf{T}} \psi(\mathbf{x}_i) \\ (\mathbf{K})_{ij} &= k(\mathbf{x}_i, \mathbf{x}_j) = \psi(\mathbf{x}_i)^{\mathsf{T}} \psi(\mathbf{x}_j) \end{aligned}$$

Bayesian Linear Regression as a Kernel Machine

Proof

- Mean and variance of the predictions:

$$p(t_* | \mathbf{X}, \mathbf{t}, \mathbf{x}_*, \sigma^2) = \mathcal{N}(\boldsymbol{\phi}_*^{\mathsf{T}} \boldsymbol{\mu}, \sigma^2 + \boldsymbol{\phi}_*^{\mathsf{T}} \boldsymbol{\Sigma} \boldsymbol{\phi}_*)$$

- Rewrite the mean:

$$\begin{aligned}
\boldsymbol{\phi}_*^{\mathsf{T}} \boldsymbol{\mu} &= \frac{1}{\sigma^2} \boldsymbol{\phi}_*^{\mathsf{T}} \boldsymbol{\Sigma} \boldsymbol{\Phi}^{\mathsf{T}} \mathbf{t} \\
&= \frac{1}{\sigma^2} \boldsymbol{\phi}_*^{\mathsf{T}} \left(\mathbf{S} - \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}} \left(\sigma^2 \mathbf{I} + \boldsymbol{\Phi} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}} \right)^{-1} \boldsymbol{\Phi} \mathbf{S} \right) \boldsymbol{\Phi}^{\mathsf{T}} \mathbf{t} \\
&= \frac{1}{\sigma^2} \boldsymbol{\phi}_*^{\mathsf{T}} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}} \left(\mathbf{I} - \left(\sigma^2 \mathbf{I} + \boldsymbol{\Phi} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}} \right)^{-1} \boldsymbol{\Phi} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}} \right) \mathbf{t} \\
&= \frac{1}{\sigma^2} \boldsymbol{\phi}_*^{\mathsf{T}} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}} \left(\mathbf{I} - \left(\mathbf{I} + \frac{\boldsymbol{\Phi} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}}}{\sigma^2} \right)^{-1} \frac{\boldsymbol{\Phi} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}}}{\sigma^2} \right) \mathbf{t}
\end{aligned}$$

... continued

Bayesian Linear Regression as a Kernel Machine

Proof

- Define $\mathbf{H} = \frac{\mathbf{\Phi S \Phi}^{\mathrm{T}}}{\sigma^2}$
- The term in the parenthesis

$$\left(\mathbf{I} - \left(\mathbf{I} + \frac{\mathbf{\Phi S \Phi}^{\mathrm{T}}}{\sigma^2}\right)^{-1} \frac{\mathbf{\Phi S \Phi}^{\mathrm{T}}}{\sigma^2}\right)$$

  becomes

$$\left(\mathbf{I} - (\mathbf{I} + \mathbf{H})^{-1}\mathbf{H}\right) = \mathbf{I} - (\mathbf{H}^{-1} + \mathbf{I})^{-1}$$

- Using Woodbury ($A, U, V = \mathbf{I}$ and $C = \mathbf{H}^{-1}$)

$$\mathbf{I} - (\mathbf{H}^{-1} + \mathbf{I})^{-1} = (\mathbf{I} + \mathbf{H})^{-1}$$

Bayesian Linear Regression as a Kernel Machine

Proof

- Substituting into the expression of the predictive mean

$$
\begin{aligned}
\phi_*^{\mathsf{T}} \boldsymbol{\mu} &= \frac{1}{\sigma^2} \phi_*^{\mathsf{T}} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}} \left(\mathbf{I} - \left(\mathbf{I} + \frac{\boldsymbol{\Phi} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}}}{\sigma^2} \right)^{-1} \frac{\boldsymbol{\Phi} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}}}{\sigma^2} \right) \mathbf{t} \\
&= \frac{1}{\sigma^2} \phi_*^{\mathsf{T}} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}} \left(\mathbf{I} + \frac{\boldsymbol{\Phi} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}}}{\sigma^2} \right)^{-1} \mathbf{t} \\
&= \phi_*^{\mathsf{T}} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}} \left(\sigma^2 \mathbf{I} + \boldsymbol{\Phi} \mathbf{S} \boldsymbol{\Phi}^{\mathsf{T}} \right)^{-1} \mathbf{t} \\
&= \mathbf{k}_*^{\top} \left(\sigma^2 \mathbf{I} + \mathbf{K} \right)^{-1} \mathbf{t}
\end{aligned}
$$

- All definitions as in the case of the variance

$$
\begin{aligned}
\psi(\mathbf{x}) &= \mathbf{S}^{1/2} \phi(\mathbf{x}) \\
(\mathbf{k}_*)_i &= k(\mathbf{x}_*, \mathbf{x}_i) = \psi(\mathbf{x}_*)^{\mathsf{T}} \psi(\mathbf{x}_i) \\
(\mathbf{K})_{ij} &= k(\mathbf{x}_i, \mathbf{x}_j) = \psi(\mathbf{x}_i)^{\mathsf{T}} \psi(\mathbf{x}_j)
\end{aligned}
$$

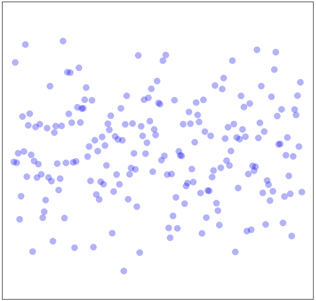
Gaussian Processes

Gaussian Processes can be explained in two ways

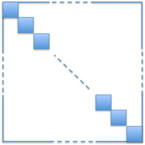
- Weight Space View
  - Bayesian linear regression with infinite basis functions
- **Function Space View**
  - **Defined as priors over functions**

Gaussian Processes - Prior over Functions

- Consider an infinite number of Gaussian random variables
- Think of them as indexed by the real line and as independent
- Denote them as $f(x)$



$K =$



Kernel

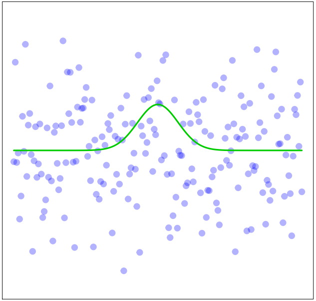
- Consider the Gaussian kernel again

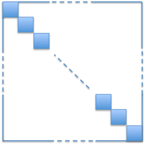
$$k(\mathbf{x}, \mathbf{x}') = \alpha \exp(-\beta \|\mathbf{x} - \mathbf{x}'\|^2)$$

- We introduced some parameters for added flexibility

Gaussian Processes - Prior over Functions

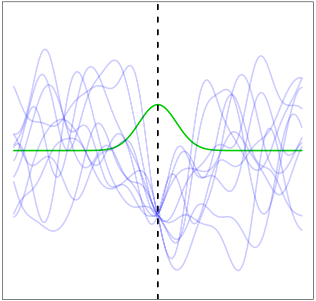
- Impose covariance using the kernel function

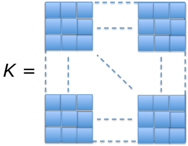


$K =$ 

Gaussian Processes - Prior over Functions

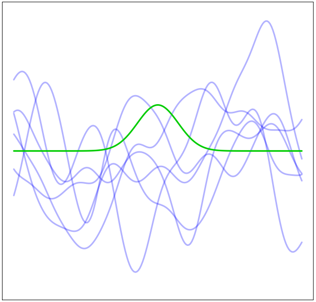
- Draw the infinite random variables again fixing one of them (the one at $x = 0$)

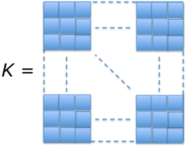




Gaussian Processes - Prior over Functions

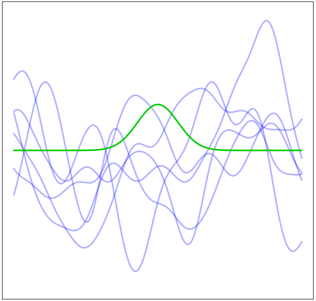
- Draw the infinite random variables again allowing the one at $x = 0$ to be random too



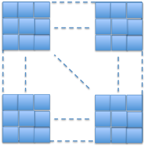


Gaussian Processes - Prior over Functions

- This can be used as a prior over functions!

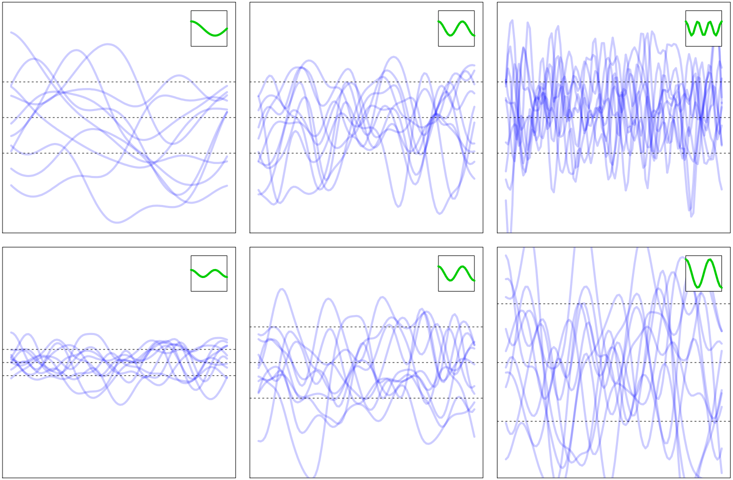


$K =$



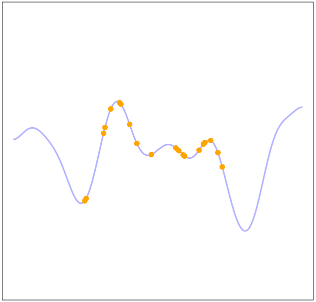
Gaussian Processes - Priors over Functions

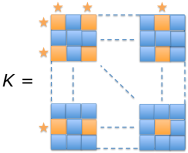
- Infinite Gaussian random variables with parameterized and input-dependent covariance



Gaussian Processes - Prior over Functions

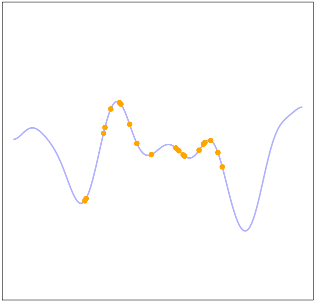
- The distribution of N random variables $f(x_1), \dots, f(x_N)$ depends exclusively on the corresponding rows and columns of the infinite by infinite kernel matrix K

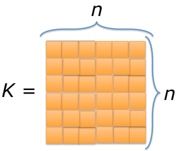




Gaussian Processes - Prior over Functions

- The distribution of N random variables $f(x_1), \ldots, f(x_N)$ depends exclusively on the corresponding rows and columns of the infinite by infinite kernel matrix K





Gaussian Processes - Prior over Functions

- The marginal distribution of $\mathbf{f} = (f(x_1), \ldots, f(x_N))^\top$ is

$$p(\mathbf{f}|\mathbf{X}) = \mathcal{N}(\mathbf{0}, \mathbf{K})$$

- The conditional distribution of f_* given $\mathbf{f}$

$$p(f_*|\mathbf{f}, \mathbf{x}_*, \mathbf{X}) = \mathcal{N}(\bar{m}, \bar{s}^2)$$

with

$$\bar{m} = \mathbf{k}_*^\top \mathbf{K}^{-1}$$

$$\bar{s}^2 = k_{**} - \mathbf{k}_*^\top \mathbf{K}^{-1} \mathbf{k}_*$$

Gaussian Processes - Prior over Functions

- Remember that when we modeled labels $\mathbf{t}$ in the linear model we assumed noise with variance σ around $\mathbf{w}^{\mathsf{T}}\mathbf{x}$
- We can do the same in Gaussian processes

$$p(\mathbf{t}|\mathbf{f}) = \prod_{i=1}^{N} p(t_i|f_i)$$

with

$$p(t_i|f_i) = \mathcal{N}(t_i|f_i, \sigma^2)$$

- Likelihood and prior are both Gaussian - conjugate!

Gaussian Processes - Prior over Functions

- Remember that when we modeled labels $\mathbf{t}$ in the linear model we assumed noise with variance σ around $\mathbf{w}^{\mathrm{T}}\mathbf{x}$
- We can do the same in Gaussian processes

$$p(\mathbf{t}|\mathbf{f}) = \prod_{i=1}^{N} p(t_i|f_i)$$

with

$$p(t_i|f_i) = \mathcal{N}(t_i|f_i, \sigma^2)$$

- Likelihood and prior are both Gaussian - conjugate!
- We can integrate out Gaussian process prior on $\mathbf{f}$

$$p(\mathbf{t}|\mathbf{X}) = \int p(\mathbf{t}|\mathbf{f})p(\mathbf{f}|\mathbf{X})d\mathbf{f}$$

- This gives

$$p(\mathbf{t}|\mathbf{X}) = \mathcal{N}(\mathbf{0}, \mathbf{K} + \sigma^2\mathbf{I})$$

Gaussian Processes - Prior over Functions

- We can derive the predictive distribution of the function also make predictions as follows:

$$p(f_*|\mathbf{t}, \mathbf{x}_* \mathbf{X}) = \int p(f_*|\mathbf{f}, \mathbf{x}_*, \mathbf{X}) p(\mathbf{f}|\mathbf{t}, \mathbf{X}) d\mathbf{f} df_* = \mathcal{N}(m, s^2)$$

  with

$$m = \mathbf{k}_*^\top \left(\mathbf{K} + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{t}$$

$$s^2 = k_{**} - \mathbf{k}_*^\top \left(\mathbf{K} + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{k}_*$$

- Same expression as in the “Weight-Space View” section

Gaussian Processes - Prior over Functions

- We can also make predictions as follows:

$$
\begin{aligned}
p(t_* | \mathbf{t}, \mathbf{x}_* \mathbf{X}) &= \int p(t_* | f_*) p(f_* | \mathbf{f}, \mathbf{x}_*, \mathbf{X}) p(\mathbf{f} | \mathbf{t}, \mathbf{X}) d\mathbf{f} df_* \\
&= \mathcal{N}(m_t, s_t^2)
\end{aligned}
$$

  with

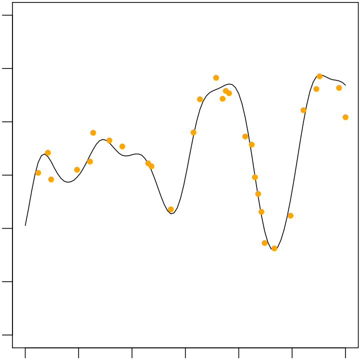
$$
m_t = \mathbf{k}_*^\top \left(\mathbf{K} + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{t}
$$

$$
s_t^2 = \sigma^2 + k_{**} - \mathbf{k}_*^\top \left(\mathbf{K} + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{k}_*
$$

- Same expression as in the "Weight-Space View" section

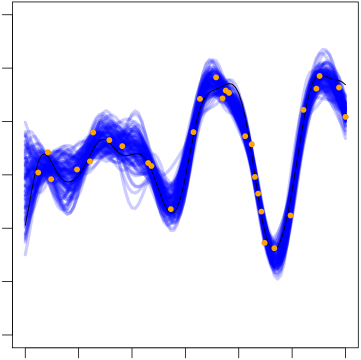
Gaussian Processes - Regression example

- Some data generated as a noisy version of some function



Gaussian Processes - Regression example

- Draws from the posterior distribution over f_* on the real line



Optimization of Gaussian Process parameters

- The kernel has parameters that have to be tuned

$$k(\mathbf{x}, \mathbf{x}') = \alpha \exp(-\beta \|\mathbf{x} - \mathbf{x}'\|^2)$$

  ... and there is also the noise parameter σ^2.

- Define $\boldsymbol{\theta} = (\alpha, \beta, \sigma^2)$
- How should we tune them?

Optimization of Gaussian Process parameters

- Define $\mathbf{K}_t = \mathbf{K} + \sigma^2 \mathbf{I}$
- Maximize the logarithm of the likelihood

$$p(\mathbf{t}|\mathbf{X}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{0}, \mathbf{K}_t)$$

  that is

$$-\frac{1}{2} \log |\mathbf{K}_t| - \frac{1}{2} \mathbf{t}^{\mathrm{T}} \mathbf{K}_t^{-1} \mathbf{t} + \text{const.}$$

- Derivatives can be useful for gradient-based optimization

$$\frac{\partial \log[p(\mathbf{t}|\mathbf{X}, \boldsymbol{\theta})]}{\partial \boldsymbol{\theta}_i}$$

Optimization of Gaussian Process parameters

- Log-likelihood

$$-\frac{1}{2}\log|\mathbf{K}_t| - \frac{1}{2}\mathbf{t}^{\mathsf{T}}\mathbf{K}_t^{-1}\mathbf{t} + \mathrm{const.}$$

- Derivatives can be useful for gradient-based optimization:

$$\frac{\partial \log[p(\mathbf{t}|\mathbf{X},\boldsymbol{\theta})]}{\partial \boldsymbol{\theta}_i} = -\frac{1}{2}\mathrm{Tr}\left(\mathbf{K}_t^{-1}\frac{\partial \mathbf{K}_t}{\partial \boldsymbol{\theta}_i}\right) + \frac{1}{2}\mathbf{t}^{\mathsf{T}}\mathbf{K}_t^{-1}\frac{\partial \mathbf{K}_t}{\partial \boldsymbol{\theta}_i}\mathbf{K}_t^{-1}\mathbf{t}$$

Summary

- Introduced Gaussian Processes
  - Weight space view
  - Function space view
- Gaussian processes for regression
- Optimization of kernel parameters
- To think about:
  - Gaussian processes for classification?
  - Scalability?