Lecture 10

Gaussian Processes

Luigi Freda

ALCOR Lab
DIAG
University of Rome "La Sapienza"

December 20, 2016

Outline

1. Intro
   - Distribution over functions
   - Stochastic Processes
   - Gaussian Processes

2. GPs for Regression
   - GP Prior
   - Predictions Using Noisy-free Observations
   - Predictions Using Noisy Observations
   - Effect of Kernel Parameters
   - Estimating the Kernel Parameters

3. Linear Smoothers
   - Linear Smoothers

Outline

1. Intro
   - Distribution over functions
   - Stochastic Processes
   - Gaussian Processes
2. GPs for Regression
   - GP Prior
   - Predictions Using Noisy-free Observations
   - Predictions Using Noisy Observations
   - Effect of Kernel Parameters
   - Estimating the Kernel Parameters
3. Linear Smoothers
   - Linear Smoothers

Introduction

Distribution over functions

- in **supervised learning**, we observe some input vector $\mathbf{x}_i$ and some scalar outputs y_i
- we assume that $y_i = f(\mathbf{x}_i)$, for some **unknown function** f, possibly corrupted by **noise** ϵ
- the optimal approach is to infer a **distribution over functions** given the data, $p(f|\mathbf{X}, \mathbf{y})$, and then to use this to make predictions given new inputs, i.e., to compute

$$p(y^*|\mathbf{x}^*, \mathbf{X}, \mathbf{y}) = \int p(\mathbf{y}^*, f|\mathbf{x}^*, \mathbf{X}, \mathbf{y})df = \int p(\mathbf{y}^*|f, \mathbf{x}^*)p(f|\mathbf{X}, \mathbf{y})df$$

- question: how can we characterize a **distribution over functions** $p(f)$?
- in order to answer, we first need to introduce the concept of stochastic process

Outline

1. Intro
    - Distribution over functions
    - Stochastic Processes
    - Gaussian Processes
2. GPs for Regression
    - GP Prior
    - Predictions Using Noisy-free Observations
    - Predictions Using Noisy Observations
    - Effect of Kernel Parameters
    - Estimating the Kernel Parameters
3. Linear Smoothers
    - Linear Smoothers

- a **stochastic process** is a statistical model where each **observation** correspond to a **function**

more formally

- let $\mathcal{T}$ be a subset of $[0, \infty)$
- a family of random variables $\{X_t\}_{t \in \mathcal{T}}$, indexed by $\mathcal{T}$, is called a **stochastic process**
- when $\mathcal{T} = \mathbb{N}$, $\{X_t\}_{t \in \mathcal{T}}$ is said to be a **discrete-time process**
- when $\mathcal{T} = [0, \infty)$, it is called a **continuous-time process**

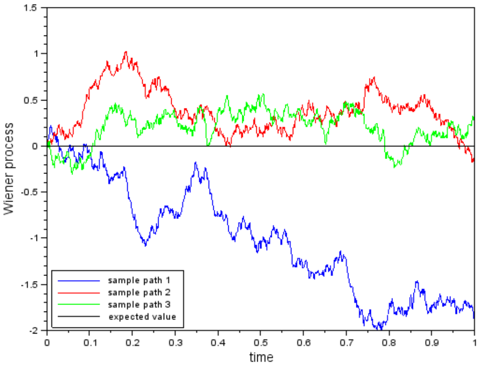
note that

- when $\mathcal{T}$ is a singleton (say $\mathcal{T} = \{1\}$), the process $\{X_t\}_{t \in \mathcal{T}} \equiv X_1$ is really just a single **random variable**
- when $\mathcal{T}$ is finite (e.g., $\mathcal{T} = \{1, 2, ..., n\}$), we get a **random vector**

Introduction

Stochastic Process

- every stochastic process can be viewed as a **function** of **two variables** t and $\omega \in \Omega$
- for each fixed $(t, \omega) \to X_t(\omega)$ is a random variable
- if we change our point of view and keep ω fixed, the stochastic process is a function mapping ω to the real-valued function $t \to X_t(\omega)$ (these functions are called the **trajectories** of the stochastic process X)

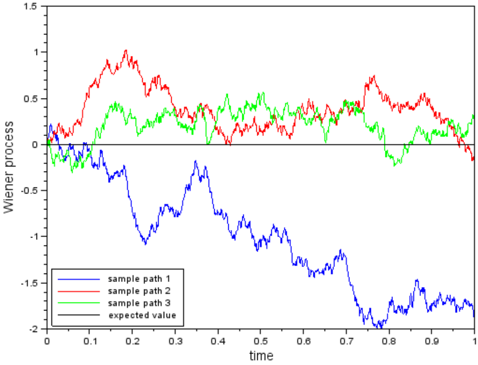


Introduction

Stochastic Process

how can we study/characterize a stochastic process $\{X_t\}_{t \in \mathcal{T}}$?

- we can start by fixing $t = t_1$ and characterizing the PDF $p_{X_1}(x_1)$ of the RV X_1
- then we can consider two values $t_1, t_2 \in \mathcal{T}$ and characterize the joint PDF $p_{X_1,X_2}(x_1, x_2)$ of the RVs X_1 and X_2
- in general we can consider any arbitrary finite set of values $t_1, ..., t_n$ and its corresponding joint PDF $p_{X_1,...,X_n}(x_1, ..., x_n)$



Outline

1. Intro
   - Distribution over functions
   - Stochastic Processes
   - Gaussian Processes
2. GPs for Regression
   - GP Prior
   - Predictions Using Noisy-free Observations
   - Predictions Using Noisy Observations
   - Effect of Kernel Parameters
   - Estimating the Kernel Parameters
3. Linear Smoothers
   - Linear Smoothers

Introduction

Gaussian processes

in a **Gaussian process** representing an unknown function f

- every point $y_i = f(\mathbf{x}_i)$ is associated with a normally distributed random variable, i.e.

$$f(\mathbf{x}_i) \sim \mathcal{N}(\mu(\mathbf{x}_i), \sigma(\mathbf{x}_i))$$

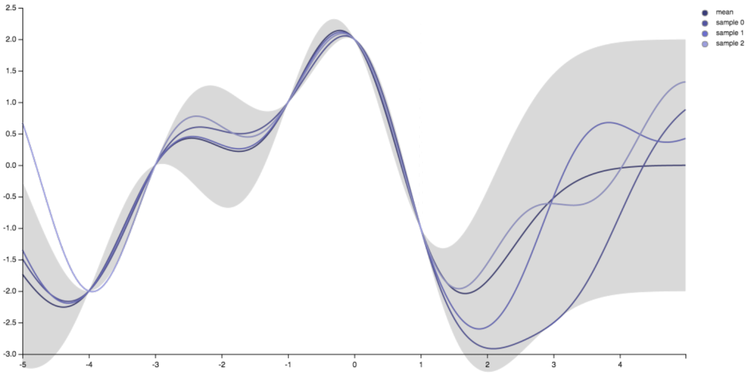
- every finite collection of random variables $f(\mathbf{x}_1), ..., f(\mathbf{x}_n)$ has a multivariate normal distribution

$$p(f(\mathbf{x}_1), ..., f(\mathbf{x}_n)) \sim \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}(\mathbf{x}_1, ..., \mathbf{x}_n), \boldsymbol{\Sigma}(\mathbf{x}_1, ..., \mathbf{x}_n))$$

  the covariance $\boldsymbol{\Sigma}(\mathbf{x}_1, ..., \mathbf{x}_n)$ has elements $\Sigma_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$ where κ is a positive definite kernel function

Introduction

An Example



different observations (trajectories) of a Gaussian process with

- mean function μ (black)
- $\mu \pm 2\sigma$ functions (95% confidence)

why should we use a Gaussian processes?

- GP based methods can be thought of as a **Bayesian alternative** to the presented kernel methods (including SVM)
- although those kernel methods are sparser and therefore faster, they do not give well-calibrated **probabilistic outputs** (i.e. estimates plus confidences)

Outline

1 Intro

- Distribution over functions
- Stochastic Processes
- Gaussian Processes

2 GPs for Regression

- GP Prior
- Predictions Using Noisy-free Observations
- Predictions Using Noisy Observations
- Effect of Kernel Parameters
- Estimating the Kernel Parameters

3 Linear Smoothers

- Linear Smoothers

GP Prior

- a GP defines a **prior** over functions, which can be converted into a **posterior** over functions once we have seen some data $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N$
- the **GP prior** on the **regression function** is denoted by

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

  where $m(\mathbf{x}) \in \mathbb{R}$ is the **mean function** and $\kappa(\mathbf{x}, \mathbf{x}') \in \mathbb{R}$ is the kernel or **covariance function**

  $$m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})]$$
  $$\kappa(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - \mathbf{m}(\mathbf{x}))(f(\mathbf{x}') - \mathbf{m}(\mathbf{x}'))^T]$$

  N.B.: $\kappa(\mathbf{x}, \mathbf{x}')$ is required to be a **positive definite kernel**
- for any finite set of points, the process defines a **joint Gaussian**

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

  where $\mathbf{f} \triangleq [f(\mathbf{x}_1), ..., f(\mathbf{x}_N)]^T \in \mathbb{R}^N$, $K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$, $\boldsymbol{\mu} = [m(\mathbf{x}_1), ..., m(\mathbf{x}_N)]^T \in \mathbb{R}^N$,
- note that it is common to use a mean function of $\mathbf{m}(\mathbf{x}) = 0$, since the GP is flexible enough to model the mean arbitrarily well
- it is also possible to consider **parametric models** for the **mean function**

Outline

1. Intro
   - Distribution over functions
   - Stochastic Processes
   - Gaussian Processes
2. GPs for Regression
   - GP Prior
   - Predictions Using Noisy-free Observations
   - Predictions Using Noisy Observations
   - Effect of Kernel Parameters
   - Estimating the Kernel Parameters
3. Linear Smoothers
   - Linear Smoothers

Predictions Using Noisy-free Observations

- suppose we observe a training set $\mathcal{D} = \{(\mathbf{x}_i, f_i), i = 1 : N\}$, where $f_i = f(\mathbf{x}_i)$ is the **noise-free** observation of the function evaluated at $\mathbf{x}_i$
- given a test set $\mathbf{X}_*$ of size $N_* \times D$, we want to predict the function outputs $\mathbf{f}_*$

what do we expect?

- we have assumed the observations are **noiseless**
- if we ask the GP to predict $f(\mathbf{x})$ for a **value** of $\mathbf{x}$ that it has **already seen**, we want the GP to return the answer $f(\mathbf{x})$ with **no uncertainty**
- in other words, it should act as an **interpolator** of the training data

Predictions Using Noisy-free Observations

- by definition of the GP, the **joint distribution** has the following form

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\mu}_* \end{bmatrix}, \begin{bmatrix} \mathbf{K} & \mathbf{K}_* \\ \mathbf{K}_*^T & \mathbf{K}_{**} \end{bmatrix} \right)$$

$\boldsymbol{\mu} = \boldsymbol{\mu}(\mathbf{X})$, $\boldsymbol{\mu}_* = \boldsymbol{\mu}(\mathbf{X}_*)$,
$\mathbf{K} = \kappa(\mathbf{X}, \mathbf{X}) \in \mathbb{R}^{N \times N}$, $\mathbf{K}_* = \kappa(\mathbf{X}, \mathbf{X}_*) \in \mathbb{R}^{N \times N_*}$, $\mathbf{K}_{**} = \kappa(\mathbf{X}_*, \mathbf{X}_*) \in \mathbb{R}^{N_* \times N_*}$

Marginals and Conditionals

Theorem 1

(Marginals and conditionals for an MVN)
Suppose $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2) \sim \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$*, i.e.* $\mathbf{x}$ *is jointly Gaussian with parameters*

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \quad \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix}, \quad \boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1} = \begin{bmatrix} \boldsymbol{\Lambda}_{11} & \boldsymbol{\Lambda}_{12} \\ \boldsymbol{\Lambda}_{21} & \boldsymbol{\Lambda}_{22} \end{bmatrix}$$

then the **marginals** *are given by*

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

and the **posterior conditional** *is given by*

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

Predictions Using Noisy-free Observations

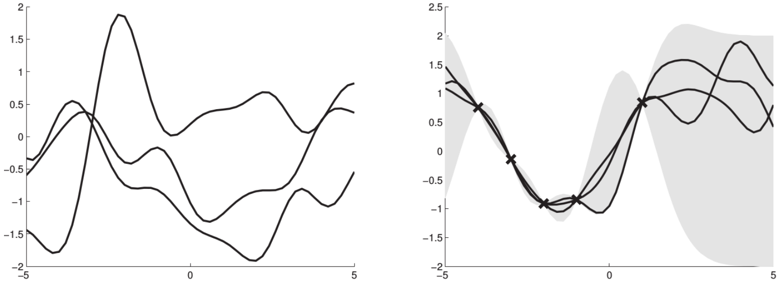
- by definition of the GP, the **joint distribution** has the following form

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\mu}_* \end{bmatrix}, \begin{bmatrix} \mathbf{K} & \mathbf{K}_* \\ \mathbf{K}_*^T & \mathbf{K}_{**} \end{bmatrix} \right)$$

- by the standard rules for conditioning Gaussians (see lec. 5), the posterior has the following form

$$\begin{aligned} p(\mathbf{f}_*|\mathbf{X}_*, \mathbf{X}, \mathbf{f}) &= \mathcal{N}(\mathbf{f}_*|\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*) \\ \boldsymbol{\mu}_* &= \boldsymbol{\mu}(\mathbf{X}_*) + \mathbf{K}_*^T \mathbf{K}^{-1}(\mathbf{f} - \boldsymbol{\mu}(\mathbf{X})) \\ \boldsymbol{\Sigma}_* &= \mathbf{K}_{**} - \mathbf{K}_*^T \mathbf{K}^{-1} \mathbf{K}_* \end{aligned}$$

Predictions Using Noisy-free Observations



- *left*: some functions sampled from a GP prior with SE (Squared Exponential) kernel
- *right*: some samples from a GP posterior, after conditioning on 5 **noise-free observations**
- the shaded area represents $\mathbb{E}[f(\mathbf{x})] \pm 2\text{std}(f(x))$
- the model perfectly interpolates the training data
- the predictive uncertainty increases as we move further away from the observed data

Squared Exponential Kernel

- in the previous 1D example, we used the **squared exponential kernel**

$$\kappa(x, x') = \sigma_f^2 \exp\left(- \frac{1}{2l^2} (x - x')^2 \right)$$

- l controls the **horizontal length scale** over which the function varies
- σ_f^2 controls the **vertical scale** (variation) of the function

Outline

1 Intro
- Distribution over functions
- Stochastic Processes
- Gaussian Processes

2 GPs for Regression
- GP Prior
- Predictions Using Noisy-free Observations
- Predictions Using Noisy Observations
- Effect of Kernel Parameters
- Estimating the Kernel Parameters

3 Linear Smoothers
- Linear Smoothers

Predictions Using Noisy Observations

- now let's consider the case where what we observe is a **noisy** version of the underlying function, i.e.
  $$y = f(\mathbf{x}) + \epsilon$$
  where $\epsilon \sim \mathcal{N}(0, \sigma_y^2)$
- in this case, the model is **not** required to interpolate the data (since they are noisy), but it must come "**close**" to the **observed data**
- one has that $y|\mathbf{x} \sim \mathcal{N}(m(\mathbf{x}), \sigma_y^2)$ since $\mathbb{E}[y|\mathbf{x}] = E[f(\mathbf{x}) + \epsilon] = E[f(\mathbf{x})] = m(\mathbf{x})$
- the covariance of the observed noisy responses is
  $$\text{cov}[y_p, y_q] = \text{cov}[f(\mathbf{x}_p) + \epsilon_p, f(\mathbf{x}_q) + \epsilon_q]$$
  which, given the noise terms ϵ_i are iid, entails
  $$\text{cov}[y_p, y_q] = \kappa(\mathbf{x}_p, \mathbf{x}_q) + \sigma_y^2 \delta_{pq}$$
  where $\delta_{pq} \triangleq \mathbb{I}(p = q)$

Predictions Using Noisy Observations

- we have $\mathbf{y} = \mathbf{f} + \boldsymbol{\epsilon}$ where $\mathbf{y} \triangleq [y_1, ..., y_N] \in \mathbb{R}^N$, $\mathbf{f} \triangleq [f(\mathbf{x}_1), ..., f(\mathbf{x}_N)]^T \in \mathbb{R}^N$, $\boldsymbol{\epsilon} \triangleq [\epsilon_1, ..., \epsilon_N] \in \mathbb{R}^N$
- this is a Gaussian linear system (see lec. 5) with $p(\mathbf{f}|\mathbf{X}) = \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}, \mathbf{K})$ and $p(\mathbf{y}|\mathbf{f}) = \prod_i \mathcal{N}(y_i|f_i, \sigma_y^2)$
- considering that $\mathbb{E}[y|\mathbf{x}] = m(\mathbf{x})$ and $\text{cov}[y_p, y_q] = \kappa(\mathbf{x}_p, \mathbf{x}_q) + \sigma_y^2 \delta_{pq}$ we obtain

$$p(\mathbf{y}|\mathbf{X}) = \mathcal{N}(\mathbf{y}|\boldsymbol{\mu}, \mathbf{K}_y) \quad \text{with} \quad \mathbf{K}_y \triangleq \text{cov}[\mathbf{y}|\mathbf{X}] = \mathbf{K} + \sigma_y^2 \mathbf{I}_N$$

Predictions Using Noisy Observations

- for notational simplicity let's assume that the mean function is zero, i.e. $m(\mathbf{x}) = 0$
- the **joint density** of the observed data $\mathbf{y}$ and the latent noise-free function on the test points $\mathbf{f}_*$ is given by

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K}_y & \mathbf{K}_* \\ \mathbf{K}_*^T & \mathbf{K}_{**} \end{bmatrix}\right)$$

- again, by the standard rules for conditioning Gaussians, we have that the **posterior predictive** density is

$$\begin{aligned} p(\mathbf{f}_*|\mathbf{X}_*, \mathbf{X}, \mathbf{y}) &= \mathcal{N}(\mathbf{f}_*|\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*) \\ \boldsymbol{\mu}_* &= \mathbf{K}_*^T \mathbf{K}_y^{-1} \mathbf{y} \\ \boldsymbol{\Sigma}_* &= \mathbf{K}_{**} - \mathbf{K}_*^T \mathbf{K}_y^{-1} \mathbf{K}_* \end{aligned}$$

- in the case of a single test input $\mathbf{x}_*$, this simplifies as follows

$$p(f_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y}) = \mathcal{N}(f_*|\mathbf{k}_*^T \mathbf{K}_y^{-1} \mathbf{y}, \mathbf{k}_{**} - \mathbf{k}_*^T \mathbf{K}_y^{-1} \mathbf{k}_*)$$

where $\mathbf{k}_* = [\kappa(\mathbf{x}_*, \mathbf{x}_1), ..., \kappa(\mathbf{x}_*, \mathbf{x}_N)]^T$ and $\mathbf{k}_{**} = \kappa(\mathbf{x}_*, \mathbf{x}_*)$

Predictions Using Noisy Observations

- in the case of a single test input $\mathbf{x}_*$

$$p(f_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y}) = \mathcal{N}(f_*|\mathbf{k}_*^T \mathbf{K}_y^{-1} \mathbf{y}, \mathbf{k}_{**} - \mathbf{k}_*^T \mathbf{K}_y^{-1} \mathbf{k}_*)$$

  where $\mathbf{k}_* = [\kappa(\mathbf{x}_*, \mathbf{x}_1), ..., \kappa(\mathbf{x}_*, \mathbf{x}_N)]^T$ and $\mathbf{k}_{**} = \kappa(\mathbf{x}_*, \mathbf{x}_*)$

- another way to write the **posterior mean** is as follows

$$\overline{f}_* = \mathbf{k}_*^T \mathbf{K}_y^{-1} \mathbf{y} = \sum_{i=1}^{N} \alpha_i \kappa(\mathbf{x}_*, \mathbf{x}_i)$$

  where $\boldsymbol{\alpha} = \mathbf{K}_y^{-1} \mathbf{y}$

Outline

1. Intro
    - Distribution over functions
    - Stochastic Processes
    - Gaussian Processes

2. GPs for Regression
    - GP Prior
    - Predictions Using Noisy-free Observations
    - Predictions Using Noisy Observations
    - Effect of Kernel Parameters
    - Estimating the Kernel Parameters

3. Linear Smoothers
    - Linear Smoothers

Effect of Kernel Parameters

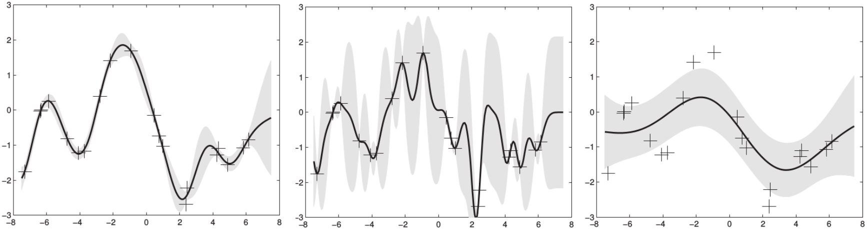
- the predictive performance of GPs depends exclusively on the **suitability** of the **chosen kernel**
- suppose we choose the following squared-exponential (SE) kernel for the 1D noisy observations

  $$\kappa_y(x_p, x_q) = \sigma_f^2 \exp\left(-\frac{1}{2l^2}(x_p - x_q)^2\right) + \sigma_y^2 \delta_{pq}$$

  where l is the **horizontal scale** over which the function changes, σ_f^2 controls the **vertical scale** of the function, and σ_y^2 is the **noise variance**

Effect of Kernel Parameters

effects of changing the parameters (l, σ_f, σ_y)



- we sampled 20 noisy data points from the SE kernel using $(l, \sigma_f, \sigma_y) = (1, 1, 0.1)$ and then made predictions changing the parameters, conditional on the data
- *left*: $(l, \sigma_f, \sigma_y) = (1, 1, 0.1)$, and the result is a good fit
- *center*: $(l, \sigma_f, \sigma_y) = (0.3, 1.08, 0.00005)$ (small l, small noise); now the function looks more "wiggly"; the uncertainty goes up faster when moving far from the training points
- *right*: $(l, \sigma_f, \sigma_y) = (3, 1.16, 0.89)$ (large l, large noise); now the function looks smoother

Outline

1. Intro
   - Distribution over functions
   - Stochastic Processes
   - Gaussian Processes

2. GPs for Regression
   - GP Prior
   - Predictions Using Noisy-free Observations
   - Predictions Using Noisy Observations
   - Effect of Kernel Parameters
   - Estimating the Kernel Parameters

3. Linear Smoothers
   - Linear Smoothers

Estimating the Kernel Parameters

- to estimate the kernel parameters, we could use **exhaustive search** over a **discrete grid** of values, with validation loss as an objective, but this can be quite **slow** (this is the approach used to tune kernels used by SVMs)
- here we consider an **empirical Bayes approach**, which will allow us to use **continuous optimization methods**, which are much faster
- in particular, we will maximize the **marginal likelihood**

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

  where $\mathbf{f} \triangleq [f(\mathbf{x}_1), ..., f(\mathbf{x}_N)]^T \in \mathbb{R}^N$, $\mathbf{y} \triangleq [y_1, ..., y_N] \in \mathbb{R}^N$, $\boldsymbol{\epsilon} \triangleq [\epsilon_1, ..., \epsilon_N] \in \mathbb{R}^N$

- we already saw that $\mathbf{y} = \mathbf{f} + \boldsymbol{\epsilon}$ is a Gaussian linear system with $p(\mathbf{f}|\mathbf{X}) = \mathcal{N}(\mathbf{f}|0, \mathbf{K})$ and $p(\mathbf{y}|\mathbf{f}) = \prod_i \mathcal{N}(y_i|f_i, \sigma_y^2)$, and we obtain

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

  where $\mathbf{K}_y = \mathbf{K} + \sigma_y^2 \mathbf{I}_N$

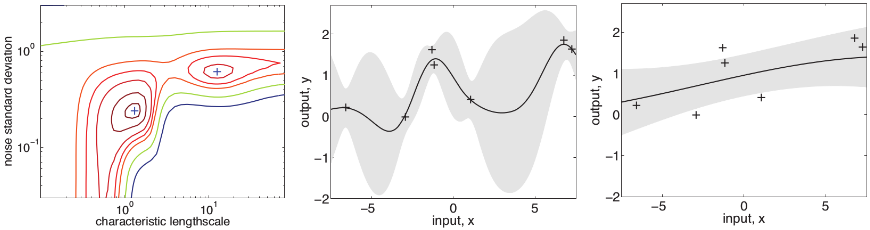
Estimating the Kernel Parameters

- hence we have to maximize the log-marginal likelihood

$$\log p(\mathbf{y}|\mathbf{X}) = \log \mathcal{N}(\mathbf{f}|0, \mathbf{K}_y) = -\frac{1}{2}\mathbf{y}^T\mathbf{K}_y^{-1}\mathbf{y} - \frac{1}{2}\log|\mathbf{K}_y| - \frac{N}{2}\log(2\pi)$$

- let $\boldsymbol{\theta}$ denote the vector of kernel parameters
- once we compute the gradient $\frac{\partial}{\partial\boldsymbol{\theta}}\log p(\mathbf{y}|\mathbf{X})$ we can we can estimate the kernel parameters using any standard gradient-based optimizer on the log marginal likelihood
- since the objective is **not convex**, **local minima** can be a problem

Estimating the Kernel Parameters



(a) (b) (c)

- (a) log marginal likelihood vs σ_y^2 and l, for fixed $\sigma_f^2 = 1$, using the 7 data points; the data was generated using $(l, \sigma_y^2) = (1, 0.1)$
- (b) the function corresponding to the lower left local minimum, $(l, \sigma_y^2) \approx (1, 0.2)$; this is quite "wiggly" and has low noise
- (c) the function corresponding to the top right local minimum, $(l, \sigma_y^2) \approx (10, 0.8)$; this is quite smooth and has high noise
-

Outline

1. Intro
   - Distribution over functions
   - Stochastic Processes
   - Gaussian Processes

2. GPs for Regression
   - GP Prior
   - Predictions Using Noisy-free Observations
   - Predictions Using Noisy Observations
   - Effect of Kernel Parameters
   - Estimating the Kernel Parameters

3. Linear Smoothers
   - Linear Smoothers

GP as Linear Smoothers

- a **linear smoother** is a regression function which is a linear function of the training outputs
$$\hat{f}(\mathbf{x}_*) = \sum_i w_i(\mathbf{x}_*) y_i$$
where $w_i(\mathbf{x}_*)$ is the i-th weight function [1]
- GP regression is a linear smoother (there are a variety of linear smoothers, such as kernel regression, locally weighted regression, smoothing splines, etc)
- to see that GP regression is a linear smoother, note that the mean of the posterior predictive distribution of a GP is
$$\overline{f}(\mathbf{x}_*) = \mathbf{k}_*^T \mathbf{K}_y^{-1} \mathbf{y} = \mathbf{k}_*^T (\mathbf{K} + \sigma_y \mathbf{I})^{-1} \mathbf{y} = \sum_{i=1}^{N} w_i(\mathbf{x}_*) y_i$$
with $w_i(\mathbf{x}_*) = [(\mathbf{K} + \sigma_y \mathbf{I})^{-1} \mathbf{k}_*]_i$

[1] do not confuse this model with the linear model $\hat{f}(\mathbf{x}_*) = \mathbf{w}^T \mathbf{x}$

GP as Linear Smoothers

- GP regression as a linear smoother

$$\overline{f}(\mathbf{x}_*) = \sum_{i=1}^{N} w_i(\mathbf{x}_*) y_i$$

  with $w_i(\mathbf{x}_*) = [(\mathbf{K} + \sigma_y \mathbf{I})^{-1} \mathbf{k}_*]_i$

- for certain GP kernel functions, one can show that $\sum_{i=1}^{N} w_i(\mathbf{x}_*) = 1$, although we may have $w_i(\mathbf{x}_*) < 0$, so we are computing a **linear combination** but not a convex combination of the y_i
- more interestingly, $w_i(\mathbf{x}_*)$ is a **local function**, even if the original kernel used by the GP is not local
- furthermore the effective bandwidth of the equivalent kernel of a GP automatically decreases as the sample size N increases

Credits

- Kevin Murphy's book