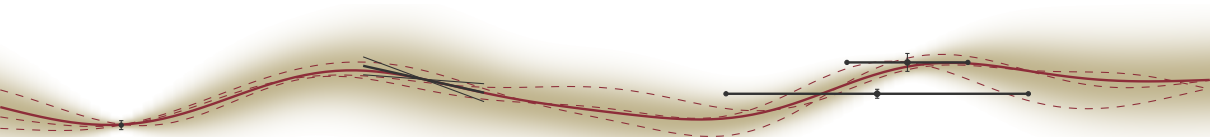
Probabilistic Inference and Learning
Lecture 13
Continuous Time Series

Philipp Hennig

28 November 2018

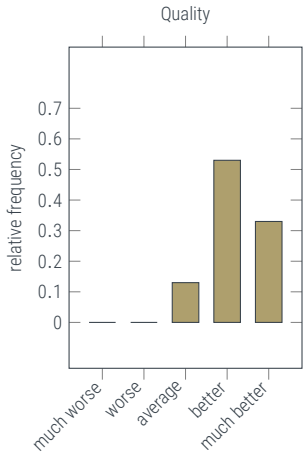
Eberhard Karls
Universität
Tübingen

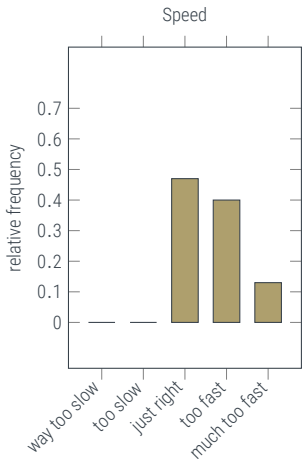
Faculty of Science
Department of Computer Science
Chair for the Methods of Machine Learning

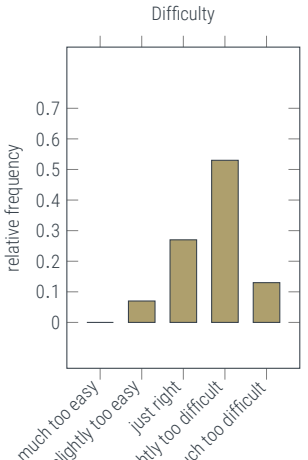


Last Lecture: Debrief

Feedback dashboard







Last Lecture: Debrief

Detailed Feedback

Things you did not like:

- too fast
- exercises are again too hard

Things you did not understand:

- the Sobolev stuff
- do we have to understand the proofs?
- I don't know what Statistical Learning Theory is

Things you enjoyed:

- **comparison of Bayesianism and Frequentism** !!
- moral discussion at the end
- π-example

Overview of Lectures so far:

0. Introduction to Reasoning under Uncertainty
1. Probabilistic Reasoning
2. Probabilities over Continuous Variables
3. Gaussian Probability Distributions
4. Gaussian Parametric Regression
5. More on Parametric Regression
6. Gaussian Processes
7. More on Kernels & GPs
8. A practical GP example
9. Markov Chains, Time Series, Filtering
10. Classification
11. Empirical Example of Classification
12. Bayesianism and Frequentism
13. Stochastic Differential Equations (today)
14. beyond Gaussians (WED)

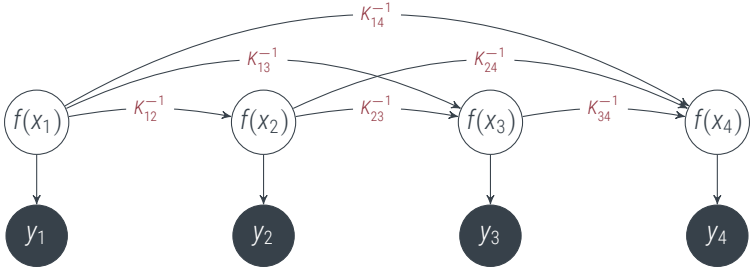
Today:

- Recap of Kalman Filters
- How to get from a Filter (A, Q) to a Gaussian process (m, k)

Graphical View I: "Full" GP

Recall from Lecture 1: Complexity of Inference can be controlled by **Conditional Independence**

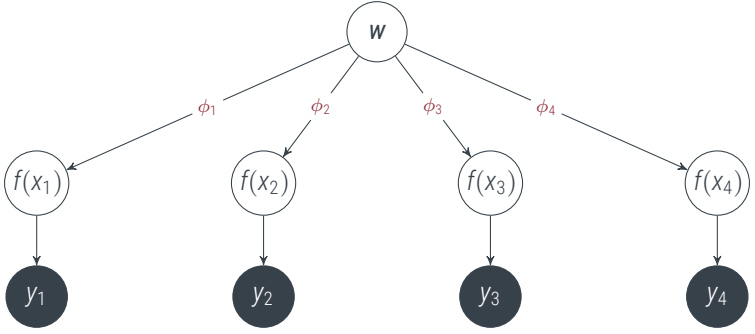
$$p(f) = \mathcal{GP}(f; 0, k) \quad p\left(\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}\right) = \mathcal{N}\left(0, \begin{bmatrix} K_{11}^{-1} & K_{12}^{-1} & K_{13}^{-1} & K_{14}^{-1} \\ & K_{22}^{-1} & K_{23}^{-1} & K_{24}^{-1} \\ & & K_{33}^{-1} & K_{34}^{-1} \\ & & & K_{44}^{-1} \end{bmatrix}^{-1}\right) \quad p(\boldsymbol{y} \mid \boldsymbol{f}) = \prod_i \mathcal{N}(y_i; f_i, \sigma^2)$$



Graphical View II: Parametric Model

Recall from Lecture 1: Complexity of Inference can be controlled by **Conditional Independence**

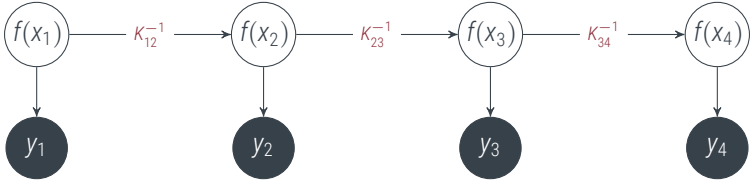
$$p(f) = \mathcal{GP}(f; 0, \Phi_X^\intercal \Sigma \Phi_X) \quad p\left(\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \middle| \, \boldsymbol{w} \right) = \prod_i \delta(f_i - \phi_i^\intercal \boldsymbol{w}) \quad p(\boldsymbol{y} \mid \boldsymbol{f}) = \prod_i \mathcal{N}(y_i; f_i, \sigma^2)$$



Graphical View III: Markov Chain

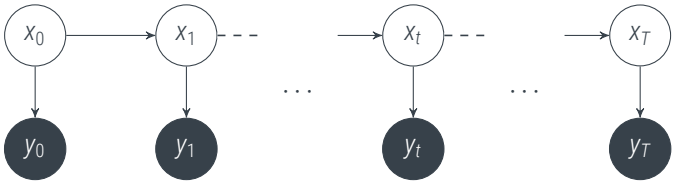
Recall from Lecture 1: Complexity of Inference can be controlled by **Conditional Independence**

$$p(f) = \mathcal{GP}(f; 0, k) \quad p\left(\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}\right) = \mathcal{N}\left(0, \begin{bmatrix} K_{11}^{-1} & K_{12}^{-1} & 0 & 0 \\ K_{12}^{-1} & K_{22}^{-1} & K_{23}^{-1} & 0 \\ 0 & K_{23}^{-1} & K_{33}^{-1} & K_{34}^{-1} \\ 0 & 0 & K_{34}^{-1} & K_{44}^{-1} \end{bmatrix}^{-1}\right) \quad p(\boldsymbol{y} \mid \boldsymbol{f}) = \prod_i \mathcal{N}(y_i; f_i, \sigma^2)$$



Assume:

$$p(x_t \mid X_{0:t-1}) = p(x_t \mid x_{t-1})$$
$$\text{and} \quad p(y_t \mid X) = p(y_t \mid x_t)$$



Filtering: $\mathcal{O}(T)$

predict: $$p(x_t \mid Y_{0:t-1}) = \int p(x_t \mid x_{t-1}) p(x_{t-1} \mid Y_{0:t-1})\, dx_{t-1}$$ (Chapman-Kolmogorov Eq.)

update: $$p(x_t \mid Y_{0:t}) = \frac{p(y_t \mid x_t) p(x_t \mid Y_{0:t-1})}{p(y_t)}$$

Smoothing: $\mathcal{O}(T)$

smooth: $$p(x_t \mid Y) = p(x_t \mid Y_{0:t}) \int p(x_{t+1} \mid x_t) \frac{p(x_{t+1} \mid Y)}{p(x_{t+1} \mid Y_{1:t})}\, dx_{t+1}$$

Time Series:

- **Markov Chains** formalize the notion of a stochastic process with a *local finite memory*
- Inference over Markov Chains separates into three operations, that can be performed in *linear* time:

Filtering: $\mathcal{O}(T)$

predict: $$p(x_t \mid Y_{0:t-1}) = \int p(x_t \mid x_{t-1}) p(x_{t-1} \mid Y_{0:t-1})\, dx_{t-1}$$ (Chapman-Kolmogorov Eq.)

update: $$p(x_t \mid Y_{0:t}) = \frac{p(y_t \mid x_t) p(x_t \mid Y_{0:t-1})}{p(y_t)}$$

Smoothing: $\mathcal{O}(T)$

smooth: $$p(x_t \mid Y) = p(x_t \mid Y_{0:t}) \int p(x_{t+1} \mid x_t) \frac{p(x_{t+1} \mid Y)}{p(x_{t+1} \mid Y_{1:t})}\, dx_{t+1}$$

procedure INFERENCE($Y, p(x_0), p(x_t \mid x_{t-1})\ \forall t, p(y_t \mid x_t)\ \forall t$)

  for i=1,…,n **do**

    $p(x_t \mid y_{1:t-1}) = \int p(x_t \mid x_{t-1}) p(x_{t-1} \mid Y_{0:t-1})\, dx_{t-1}$ // Chapman-Kolmogorov eq.

    $p(x_t \mid y_{1:t}) = p(y_t \mid x_t) p(x_t \mid Y_{0:t-1}) / p(y_t)$ // Update

  end for

  for i=n-1,…,0 **do** // Smoothing

    $p(x_t \mid Y) = p(x_t \mid Y_{0:t}) \int p(x_{t+1} \mid x_t) p(x_{t+1} \mid Y) p(x_{t+1} \mid Y_{1:t})\, dx_{t+1}$

  end for

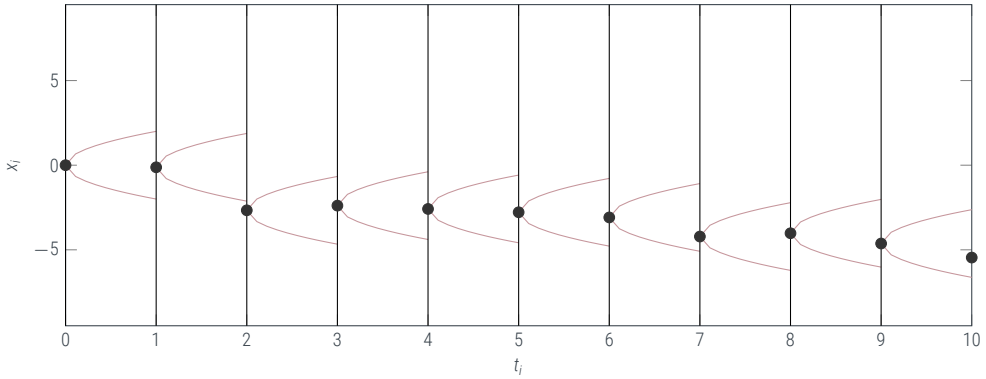
  return $p(x_t \mid Y)\ \forall t = 0, \ldots, n$ // return all marginals

end procedure

Gauss-Markov Models

Local structure for univariate Gaussian models

$$p(x(t_{i+1}) \mid x(t_i)) = \mathcal{N}(x_{i+1}; Ax_i, Q) \qquad \text{(figure: } x_0 = 0, A = Q = 1\text{)}$$



$$p(x(t_i) \mid x(t_{i-1})) = \mathcal{N}(x_i; Ax_{i-1}, Q) \qquad p(y_t \mid x_t) = \mathcal{N}(y_t; Hx_t, R)$$

predict:
$$p(x_t \mid Y_{1:t-1}) = \int p(x_t \mid x_{t-1}) p(x_{t-1} \mid Y_{0:t-1})\, dx_{t-1} = \mathcal{N}(x_t, m_t^-, P_t^-)$$
$$= \mathcal{N}(x_t, Am_t, AP_tA^\intercal + Q)$$

update:
$$p(x_t \mid Y_{1:t}) = \frac{p(y_t \mid x_t) p(x_t \mid Y_{0:t-1})}{p(y_t)} = \mathcal{N}(x_t, m_t, P_t)$$
$$= \mathcal{N}(x_t, m_t^- + Kz, (I - KH)P_t^-)$$
$$K := P_t^- H^\intercal (HP_tH^\intercal + R)^{-1}, \qquad z := y_t - Hm_t^-,$$

smooth:
$$p(x_t \mid Y) = p(x_t \mid Y_{0:t}) \int p(x_{t+1} \mid x_t) \frac{p(x_{t+1} \mid Y)}{p(x_{t+1} \mid Y_{1:t})}\, dx_{t+1} = \mathcal{N}(x_t, m_t^s, P_t^s)$$
$$= \mathcal{N}(x_t, m_t + G_t(m_{t+1}^s - m_{t+1}^-), P_t + G_t(P_{t+1}^s - P_{t+1}^-)G_t^\intercal)$$
$$G_t := P_tA^\intercal (P_t^-)^{-1}$$

Kalman Filtering and Rauch-Tung-Striebel Smoothing

linear-time inference on Gaussian-Markov time series

$$p(x(t_i) \mid x(t_{i-1})) = \mathcal{N}(x_i; Ax_{i-1}, Q) \qquad p(y_t \mid x_t) = \mathcal{N}(y_t; Hx_t, R)$$

(Kalman) Filter:

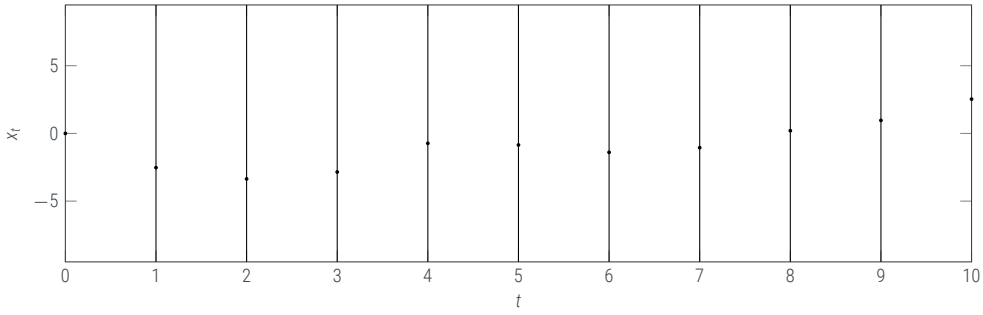
$$m_t^- = Am_{t-1} \quad \text{predictive mean}$$
$$P_t^- = AP_{t-1}A^\intercal + Q \quad \text{predictive covariance}$$
$$z_t = y - Hm_t^- \quad \text{innovation residual}$$
$$S_t = HP_t^- H^\intercal + R \quad \text{innovation covariance}$$
$$K_t = P_t^- H^\intercal S^{-1} \quad \text{Kalman gain}$$
$$m_t = m_t^- + Kz_t \quad \text{estimation mean}$$
$$P_t = (I - KH)P_t^- \quad \text{estimation covariance}$$

(Rauch Tung Striebel) Smoother:

$$G_t = P_t A^\intercal (P_{t+1}^-)^{-1} \quad \text{RTS gain}$$
$$m_t^s = m_t + G_t(m_{t+1}^s - m_{t+1}^-) \quad \text{smoothed mean}$$
$$P_t^s = P_t + G_t(P_{t+1}^s - P_{t+1}^-)G^\intercal \quad \text{smoothed covariance}$$

Continuous Time

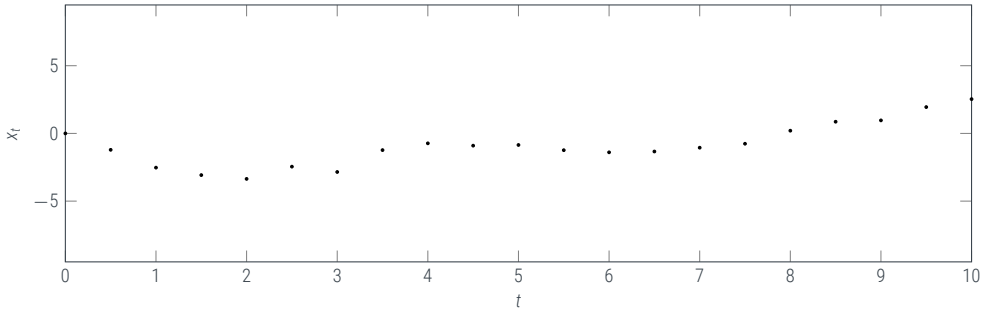
Differential equations defining non-differential curves



$$\delta t = 1 \qquad Q_{\delta t} = 1$$

Continuous Time

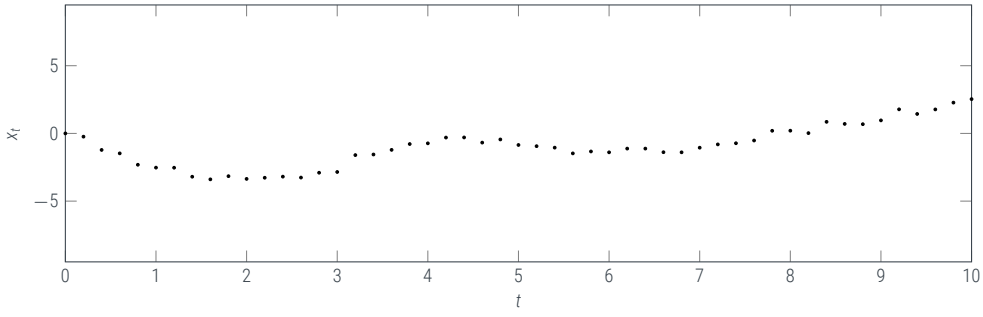
Differential equations defining non-differential curves



$$\delta t = 1/2 \qquad Q_{\delta t} = 1/2$$

Continuous Time

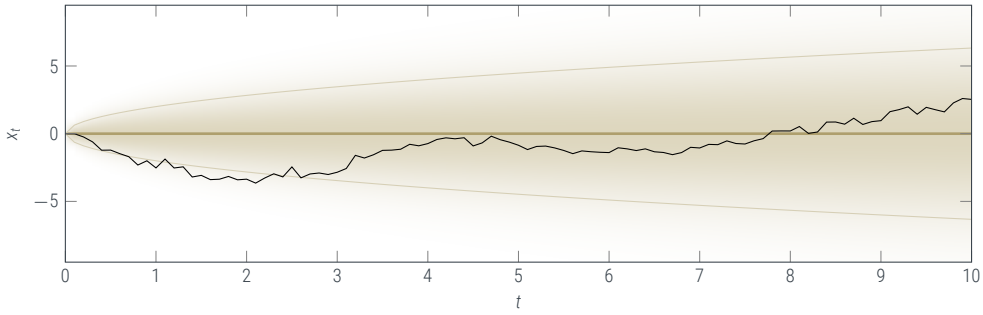
Differential equations defining non-differential curves



$$\delta t = 1/4 \qquad Q_{\delta t} = \delta t$$

Continuous Time

Differential equations defining non-differential curves



$$\delta t \to 0 \qquad Q_{\delta t} = ???$$

For the limit $\delta t \to 0$ we would like to encode that $Q_\delta t / \delta t$ approaches some kind of finite object (like a derivative, but sample paths from this (the Wiener) process) are almost surely not differentiable. So we introduce a new object: $Q_{dt} := d\omega$, known as the *Wiener measure*. (Nb: This is a non-standard construction. $d\omega$ can be defined more elegantly; but this goes beyond the scope of this course.)

Stochastic Differential Equations

a pragmatic definition

For our purposes the (linear, time-invariant) **Stochastic Differential Equation (SDE)**

$$dx(t) = Fx(t)\,dt + L\,d\omega_t,$$

together with $x(t_0) = x_0$, describes the local behaviour of the (unique) Gaussian process with

$$\mathbb{E}(x(t)) =: m(t) = e^{F(t-t_0)}x_0 \qquad \operatorname{cov}(x(t_a), x(t_b)) =: k(t_a, t_b) = \int_{t_0}^{\min t_a, t_b} e^{F(t_a-\tau)} L L^\intercal e^{F^\intercal (t_b-\tau)}\,d\tau$$

This GP is known as the **solution** of the SDE. It gives rise to the discrete-time stochastic recurrence relation $p(x_{t_{i+1}} \mid x_{t_i}) = \mathcal{N}(x_{t_{i+1}}; A_{t_i} x_{t_i}, Q_{t_i})$ with

$$A_{t_i} = e^{F(t_{i+1}-t_i)} \quad \text{and} \quad Q_{t_i} = \int_0^{t_{i+1}-t_i} e^{F\tau} L L^\intercal e^{F\tau}\,d\tau.$$

Matrix exponential: $e^X := \sum_{i=0}^{\infty} \frac{X^i}{i!}$. *Thus :* $e^0 = I$, $(e^X)^{-1} = e^{-X}$, $X = VDV^{-1} \Rightarrow Ve^D V^{-1}$, $e^{\operatorname{diag}_i d_i} = \operatorname{diag}_i e^{d_i}$, $\det e^X = e^{\operatorname{tr} X}$.

The Connection to GPs

Some well-studied examples

$$dx(t) = Fx(t)\,dt + L\,d\omega_t$$

$$\mathbb{E}(x(t)) =: m(t) = e^{F(t-t_0)}x_0 \qquad \operatorname{cov}(x(t_a), x(t_b)) =: k(t_a, t_b) = \int_{t_0}^{\min t_a, t_b} e^{F(t_a-\tau)} L L^\intercal e^{F^\intercal (t_b-\tau)}\,d\tau$$

$$A_{t_i} = e^{F(t_{i+1}-t_i)} \qquad Q_{t_i} = \int_0^{t_{i+1}-t_i} e^{F\tau} L L^\intercal e^{F^\intercal \tau}\,d\tau$$

The Connection to GPs

Some well-studied examples

$$dx(t) = Fx(t)\,dt + L\,d\omega_t$$

$$\mathbb{E}(x(t)) =: m(t) = e^{F(t-t_0)}x_0 \qquad \operatorname{cov}(x(t_a), x(t_b)) =: k(t_a, t_b) = \int_{t_0}^{\min t_a, t_b} e^{F(t_a-\tau)} L L^\intercal e^{F^\intercal (t_b-\tau)}\,d\tau$$

$$A_{t_i} = e^{F(t_{i+1}-t_i)} \qquad Q_{t_i} = \int_0^{t_{i+1}-t_i} e^{F\tau} L L^\intercal e^{F\tau}\,d\tau$$

The scaled Wiener process

$$F = 0, L = \theta \quad \Rightarrow \quad m(t) = x_0 \quad k(t_a, t_b) = \theta^2(\min(t_a, t_b) - t_0)$$
$$A = I \qquad Q_{t_i} = \theta^2(t_{i+1} - t_i)$$

The Connection to GPs

Some well-studied examples

$$dx(t) = Fx(t)\,dt + L\,d\omega_t$$

$$\mathbb{E}(x(t)) =: m(t) = e^{F(t-t_0)}x_0 \qquad \operatorname{cov}(x(t_a), x(t_b)) =: k(t_a, t_b) = \int_{t_0}^{\min t_a, t_b} e^{F(t_a-\tau)} L L^\intercal e^{F^\intercal (t_b-\tau)}\,d\tau$$

$$A_{t_i} = e^{F(t_{i+1}-t_i)} \qquad Q_{t_i} = \int_0^{t_{i+1}-t_i} e^{F\tau} L L^\intercal e^{F\tau}\,d\tau$$

The Ornstein-Uhlenbeck process

$$F = -\frac{1}{\lambda}, L = \frac{2\theta}{\sqrt{\lambda}} \quad \Rightarrow \quad m(t) = x_0 e^{-\frac{t-t_0}{\lambda}} \qquad k(t_a, t_b) = \theta^2 \left(e^{-\frac{|t_a-t_b|}{\lambda}} - e^{\frac{2t_0-t_a-t_b}{\lambda}} \right)$$

(cf. Exercises) $\qquad A = e^{-\delta_t/\lambda} \qquad Q_{t_i} = \theta^2 \left(1 - e^{-2\delta_t/\lambda}\right)$

$$dx(t) = Fx(t)\,dt + L\,d\omega_t$$

- So far, we have seen examples with $x(t) \in \mathbb{R}$.
- But F and L can also be matrices. Consider the example

$$x = \begin{bmatrix} x_{(1)} \\ x_{(2)} \end{bmatrix} \qquad F = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \qquad L = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

That is:

$$\begin{bmatrix} dx_{(1)}(t) \\ dx_{(2)}(t) \end{bmatrix} = \begin{bmatrix} x_{(2)}(t)\,dt + 0\,d\omega \\ 0\,dt + d\omega \end{bmatrix} \quad \Rightarrow \quad x_{(1)}(t) = \int_{t_0}^{t} x_{(2)}(t)\,dt + [x_0]_1$$

- Finding A, Q is an exercise.

- Consider a *linear, time-invariant (LTI)* system (A, Q, H, R)

$$p(x_{i+1} \mid x_i) = \mathcal{N}(x_{i+1}; Ax_i, Q) \qquad p(y_i \mid x_i) = \mathcal{N}(y_i; Hx_i, R)$$

procedure FILTER($m_{t-1}, P_{t-1}, A, Q, H, R, y$)

$$m_t^- = Am_{t-1} \qquad /\!/ \text{ predictive mean}$$
$$P_t^- = AP_{t-1}A^\intercal + Q \qquad /\!/ \text{ predictive covariance}$$
$$z = y - Hm_t^- \qquad /\!/ \text{ residual}$$
$$S = HP_t^- H^\intercal + R \qquad /\!/ \text{ innovation covariance}$$
$$K = P_t^- H^\intercal S^{-1} \qquad /\!/ \text{ gain}$$
$$m_t = m_t^- + Kz \qquad /\!/ \text{ updated mean}$$
$$P_t = (I - KH)P_t^- \qquad /\!/ \text{ updated covariance}$$

return $(m_t, P_t), (m_t^-, P_t^-)$

end procedure

- P^- follows a **Discrete-time algebraic Riccati Equation (DARE)**

$$P_{i+1}^- = AP_i^-A^\intercal - AP_i^-H^\intercal(HP_i^-H^\intercal + R)^{-1}HP_i^-A^\intercal + Q = Q + A((P_i^-)^{-1} + H^\intercal R^{-1}H)^{-1}A^\intercal.$$

$$P^-_{i+1} = AP^-_i A^\intercal - AP^-_i H^\intercal (HP^-_i H^\intercal + R)^{-1} HP^-_i A^\intercal + Q = Q + A((P^-_i)^{-1} + H^\intercal R^{-1} H)^{-1} A^\intercal.$$

Theorem

Consider the (symplectic) matrix $Z = \begin{bmatrix} A^\intercal + H^\intercal R^{-1} H A^{-1} Q & -H^\intercal R^{-1} H A^{-1} \\ -A^{-1} Q & A^{-1} \end{bmatrix}.$

If Z has no eigenvalues on the unit circle, then exactly half its eigenvalues are inside the unit circle, and the DARE above converges to a finite fix-point, known as the ***steady-state*** *predictive covariance. To find it, consider the matrix $U \in \mathbb{C}^{2N \times 2N}$ of the eigenvectors of Z. Assume that they are sorted such that the first N columns of U correspond to the eigenvectors for eigenvalues* inside *the unit circle (and the other N columns to those* outside *the circle). Then separate U into square $N \times N$ sub-matrices as*

$$U = \begin{bmatrix} U_1 & U_3 \\ U_2 & U_4 \end{bmatrix}.$$

The steady-state predictive covariance is given by $P^-_\infty = U_2 U_1^{-1}$.

The most basic kind of estimation

Steady-State filtering yields some old favorites

In steady-state, $P_i^- = P_\infty^-$ and $K = P_\infty^- H^\intercal (H P_\infty^- H^\intercal + R)^{-1}$, Filter turns into

$$m_t \leftarrow A m_{t-1} + K(y - H A m_{t-1})$$

- Wiener process: $F = 0, L = \theta, \delta t = 1, R = \sigma^2, H = I \quad \Rightarrow A = I, Q = \theta^2$

$$Z = \begin{bmatrix} A^\intercal + H^\intercal R^{-1} H A^{-1} Q & -H^\intercal R^{-1} H A^{-1} \\ -A^{-1} Q & A^{-1} \end{bmatrix} = \begin{bmatrix} 1 + \theta^2/\sigma^2 & -1/\sigma^2 \\ -\theta^2 & 1 \end{bmatrix}$$

$$U = \begin{bmatrix} \frac{\sqrt{4\sigma^2/\theta^2 + 1} - 1}{2\sigma^2} & \frac{-\sqrt{4\sigma^2/\theta^2 + 1} - 1}{2\sigma^2} \\ 1 & 1 \end{bmatrix}, \quad P_\infty^- = \frac{2\sigma^2}{\sqrt{4\sigma^2/\theta^2 + 1} - 1}$$

$$m_t \leftarrow (1 - K) m_{t-1} + K y \qquad \text{with} \qquad K = \frac{P_\infty^-}{P_\infty^- + \sigma^2}$$

The most basic kind of estimation

Steady-State filtering yields some old favorites

In steady-state, $P_i^- = P_\infty^-$ and $K = P_\infty^- H^\intercal (H P_\infty^- H^\intercal + R)^{-1}$, Filter turns into

$$m_t \leftarrow A m_{t-1} + K(y - H A m_{t-1})$$

- Wiener process: $F = 0, L = \theta, \delta t = 1, R = \sigma^2, H = I \quad \Rightarrow A = I, Q = \theta^2$

$$Z = \begin{bmatrix} A^\intercal + H^\intercal R^{-1} H A^{-1} Q & -H^\intercal R^{-1} H A^{-1} \\ -A^{-1} Q & A^{-1} \end{bmatrix} = \begin{bmatrix} 1 + \theta^2/\sigma^2 & -1/\sigma^2 \\ -\theta^2 & 1 \end{bmatrix}$$

$$U = \begin{bmatrix} \frac{\sqrt{4\sigma^2/\theta^2 + 1} - 1}{2\sigma^2} & \frac{-\sqrt{4\sigma^2/\theta^2 + 1} - 1}{2\sigma^2} \\ 1 & 1 \end{bmatrix}, \quad P_\infty^- = \frac{2\sigma^2}{\sqrt{4\sigma^2/\theta^2 + 1} - 1}$$

$$m_t \leftarrow (1 - K) m_{t-1} + K y \qquad \text{with} \qquad K = \frac{P_\infty^-}{P_\infty^- + \sigma^2}$$

The steady-state of the Wiener-process filter is the **running average**.

Summary:

Markov Chains capture **finite memory** of a time series through conditional independence

Gauss-Markov models map this state to linear algebra

Kalman filter is the name for the corresponding algorithm

SDEs (Stochastic Differential Equations) are the continuous-time limit of discrete-time stochastic recurrence relations (in particular, linear SDEs are the continuous-time generalization discrete-time linear Gaussian systems)

Complexity of all necessary operations is **linear**, $\mathcal{O}(N)$ in the number of datapoints (as opposed to $\mathcal{O}(N^3)$ for general GPs).
(Although not shown, this includes hyperparameter inference!)

Steady-State update rules amount to various forms of **running averages** (the covariance is constant, the mean update rule is a running average)

HMMs are the generalisation of Gauss-Markov models to non-Gaussian generative relationships

For more on Gaussian and *approximately Gaussian filters* see, e.g.
Simo Särkkä. *Bayesian Filtering and Smoothing* Cambridge University Press, 2013

`https://users.aalto.fi/~ssarkka/pub/cup_book_online_20131111.pdf`

Summary of Gaussian Models

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

Gaussians provide the linear algebra of inference

if all joints are Gaussian and all observations are linear, all posteriors are Gaussian

- products of Gaussians are Gaussians

$$\begin{aligned}&\mathcal{N}(x;a,A)\mathcal{N}(x;b,B)\\&=\mathcal{N}(x;c,C)\mathcal{N}(a;b,A+B)\\C&:=(A^{-1}+B^{-1})^{-1}\quad c:=C(A^{-1}a+B^{-1}b)\end{aligned}$$

- linear projections of Gaussians are Gaussians

$$\begin{aligned}&p(z)=\mathcal{N}(z;\mu,\Sigma)\\\Rightarrow\quad&p(Az)=\mathcal{N}(Az,A\mu,A\Sigma A^\intercal)\end{aligned}$$

- marginals of Gaussians are Gaussians

$$\int\mathcal{N}\left[\begin{pmatrix}x\\y\end{pmatrix};\begin{pmatrix}\mu_x\\\mu_y\end{pmatrix},\begin{pmatrix}\Sigma_{xx}&\Sigma_{xy}\\\Sigma_{yx}&\Sigma_{yy}\end{pmatrix}\right]dy=\mathcal{N}(x;\mu_x,\Sigma_{xx})$$

- (linear) conditionals of Gaussians are Gaussians

$$\begin{aligned}p(x\mid y)&=\frac{p(x,y)}{p(y)}\\&=\mathcal{N}\left(x;\mu_x+\Sigma_{xy}\Sigma_{yy}^{-1}(y-\mu_y),\Sigma_{xx}-\Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx}\right)\end{aligned}$$

Bayesian inference becomes linear algebra

$$\text{If } p(x)=\mathcal{N}(x;\mu,\Sigma)\qquad\text{and}\qquad p(y\mid x)=\mathcal{N}(y;A^\intercal x+b,\Lambda),\ \text{then}$$

$$p(B^\intercal x+c\mid y)=\mathcal{N}[B^\intercal x+c;B^\intercal\mu+c+B^\intercal\Sigma A(A^\intercal\Sigma A+\Lambda)^{-1}(y-A^\intercal\mu-b),B^\intercal\Sigma B-B^\intercal\Sigma A(A^\intercal\Sigma A+\Lambda)^{-1}A^\intercal\Sigma B]$$

$$
\begin{aligned}
\text{prior} \quad p(w) &= \mathcal{N}(w; \mu, \Sigma) \quad \Rightarrow \quad p(f) = \mathcal{N}(f_x; \phi_x^\intercal \mu, \phi_x \Sigma \phi_x) \\
\text{likelihood} \quad p(y \mid w, \phi_X) &= \mathcal{N}(y; \phi_X^\intercal w, \sigma^2 I) = \mathcal{N}(y; f_X, \sigma^2 I) \\
\text{posterior on } \boldsymbol{w} \quad p(w \mid \boldsymbol{y}, \phi_X) &= \mathcal{N}(w; \mu + \Sigma\phi_X(\phi_X^\intercal \Sigma \phi_X + \sigma^2 I)^{-1}(\boldsymbol{y} - \phi_X^\intercal \mu), \\
&= \mathcal{N}\left(w; (\Sigma^{-1} + \sigma^{-2}\phi_X^\intercal \phi_X)^{-1}\left(\Sigma^{-1}\mu + \sigma^{-2}\phi_X \boldsymbol{y}\right), (\Sigma^{-1} + \sigma^{-2}\phi_X^\intercal \phi_X)^{-1}\right) \\
\text{posterior on } f \quad p(f_x \mid y, \phi_X) &= \mathcal{N}(f_x; \phi_x^\intercal \mu + \phi_x^\intercal \Sigma \phi_X (\phi_X^\intercal \Sigma \phi_X + \sigma^2 I)^{-1}(y - \phi_X^\intercal \mu), \\
&\qquad \phi_x^\intercal \Sigma \phi_x - \phi_x^\intercal \Sigma \phi_X (\phi_X^\intercal \Sigma \phi_X + \sigma^2 I)^{-1} \phi_X^\intercal \Sigma \phi_x) \\
&= \mathcal{N}\left(f_x; \phi_x(\Sigma^{-1} + \sigma^{-2}\phi_X^\intercal \phi_X)^{-1}\left(\Sigma^{-1}\mu + \sigma^{-2}\phi_X \boldsymbol{y}\right), \phi_x(\Sigma^{-1} + \sigma^{-2}\phi_X^\intercal \phi_X)^{-1}\phi_x^\intercal\right) \\
\text{model evidence} \quad p(\boldsymbol{y} \mid \phi, \mu, \Sigma) &= \mathcal{N}(\boldsymbol{y}; \phi_X^\intercal \mu, \phi_X^\intercal \Sigma \phi_X + \sigma^2 I)
\end{aligned}
$$

The model (ϕ, μ, Σ) can be learnt by **hierarchical Bayesian inference** (i.e. using the model evidence as a likelihood). Since the likelihood is not linear Gaussian for ϕ, approximate inference has to be used, e.g. MAP inference.

$$
\begin{aligned}
\text{prior} \quad & p(f) = \mathcal{GP}(f; m, k) \\
\text{likelihood} \quad & p(y \mid w, \phi_X) = \mathcal{N}(y; f_X, \sigma^2 I) \\
\text{posterior} \quad & p(f \mid y, \phi_X) = \mathcal{GP}(f_x; m_x + k_{xX}(k_{XX} + \sigma^2 I)^{-1}(y - m_X), \quad k_x x - k_{xX}(k_{XX} + \sigma^2 I)^{-1} k_{Xx})
\end{aligned}
$$

Definition (Gaussian process)

Let $\mu : \mathbb{X} \to \mathbb{R}$ be any function, $k : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ be a Mercer kernel. A **Gaussian process** $p(f) = \mathcal{GP}(f; \mu, k)$ is a probability distribution over the function $f : \mathbb{X} \to \mathbb{R}$, such that every finite restriction to function values $f_X := [f_{x_1}, \ldots, f_{x_N}]$ is a Gaussian distribution $p(f_X) = \mathcal{N}(f_X; \mu_X, k_{XX})$.

Kernels are Beautiful Objects

and rightly popular

- If k_1, k_2 are kernels, ϕ any function, then so are
  - $\alpha \cdot k_1(a,b)$ for $\alpha \in \mathbb{R}_+$
  - $k_1(\phi(c), \phi(d))$ for $c, d \in \mathbb{Y}$
  - $k_1(a,b) + k_2(a,b)$
  - $k_1(a,b) \cdot k_2(a,b)$
- for any kernel k there exists a unique *reproducing kernel Hilbert space (RKHS)* $\mathcal{H}_k$, and vice versa.
- GPs are quite powerful: The posterior mean can approximate ("learn") any function in the RKHS at a rate given by that of the posterior standard deviation.
- GPs are quite limited: If $f \notin \mathcal{H}_k$, they may converge **very** (e.g. logarithmically) slowly to the truth
- Gaussian process regression is closely related to **kernel ridge regression** (nonparametric least-squares):
  - the posterior mean is the kernel ridge / regularized kernel least-squares estimate in the RKHS $\mathcal{H}_k$.

$$m(x) = k_{xX}(k_{XX} + \sigma^2 I)^{-1} \boldsymbol{y} = \operatorname*{arg\,min}_{f \in \mathcal{H}_k} \|\boldsymbol{y} - f_X\|^2 + \|f\|^2_{H_k}$$

  - the posterior variance (**expected square error**) is the **worst-case square error** in the RKHS.

$$v(x) = k_{xx} - k_{xX}(k_{XX} + \sigma^2 I)^{-1} k_{Xx} = \operatorname*{arg\,max}_{f \in \mathcal{H}_k, \|f\|_{H_k} \leq 1} \|f(x) - m(x)\|^2$$

Gauss-Markov Models

Kalman filters allow constant-time estimation in time series models

$$p(x(t_i) \mid x(t_{i-1})) = \mathcal{N}(x_i; Ax_{i-1}, Q) \qquad p(y_t \mid x_t) = \mathcal{N}(y_t; Hx_t, R)$$

predict:
$$p(x_t \mid Y_{1:t-1}) = \int p(x_t \mid x_{t-1}) p(x_{t-1} \mid Y_{0:t-1})\, dx_{t-1} = \mathcal{N}(x_t, m_t^-, P_t^-)$$
$$= \mathcal{N}(x_t, Am_t, AP_tA^\intercal + Q)$$

update:
$$p(x_t \mid Y_{1:t}) = \frac{p(y_t \mid x_t) p(x_t \mid Y_{0:t-1})}{p(y_t)} = \mathcal{N}(x_t, m_t, P_t)$$
$$= \mathcal{N}(x_t, m_t^- + Kz, (I - KH)P_t^-)$$
$$K := P_t^- H^\intercal (HP_tH^\intercal + R)^{-1}, \qquad z := y_t - Hm_t^-,$$

smooth:
$$p(x_t \mid Y) = p(x_t \mid Y_{0:t}) \int p(x_{t+1} \mid x_t) \frac{p(x_{t+1} \mid Y)}{p(x_{t+1} \mid Y_{1:t})}\, dx_{t+1} = \mathcal{N}(x_t, m_t^s, P_t^s)$$
$$= \mathcal{N}(x_t, m_t + G_t(m_{t+1}^s - m_{t+1}^-), P_t + G_t(P_{t+1}^s - P_{t+1}^-)G_t^\intercal)$$
$$G_t := P_tA^\intercal (P_t^-)^{-1}$$

The Laplace Approximation for Generalized Linear Models

Logistic Regression and Beyond

- Assume $p(f) = \mathcal{GP}(f; m, k)$, but non-Gaussian likelihood $p(y \mid f)$ (e.g. $= \sigma(y \cdot f)$)
- Find maximum posterior probability for **latent** f at **training points**

$$\hat{f} = \arg\max \log p(\mathbf{f}_X \mid y)$$

- Assign approximate Gaussian posterior at training points

$$q(\mathbf{f}_X) = \mathcal{N}(\mathbf{f}_X; \hat{\mathbf{f}}, -(\nabla\nabla^\intercal \log p(\mathbf{f}_X \mid y)|_{f_X=\hat{f}})^{-1}) =: \mathcal{N}(\mathbf{f}_X; \hat{\mathbf{f}}, \hat{\Sigma})$$

- approximate posterior **predictions** at f_x for **latent function**

$$\begin{aligned} q(f_x \mid y) &= \int p(f_x \mid \mathbf{f}_X) q(\mathbf{f}_X)\, d\mathbf{f}_X = \int \mathcal{N}(f_x; m_x + k_{xX}K_{XX}^{-1}(\mathbf{f}_X - m_X), k_{xx} - k_{xX}K_{XX}^{-1}k_{Xx}) q(\mathbf{f}_X)\, d\mathbf{f}_X \\ &= \mathcal{N}(f_x;\ m_x + k_{xX}K_{XX}^{-1}(\hat{\mathbf{f}} - m_X),\ k_{xx} - k_{xX}K_{XX}^{-1}k_{Xx} + k_{xX}K_{XX}^{-1}\hat{\Sigma}K_{XX}^{-1}k_{Xx}) \end{aligned}$$

- compute predictions for **label probabilities**:

$$\mathbb{E}_{p(f|y)}[\pi_x] \approx \mathbb{E}_q[\pi_x] = \int \sigma(f_x) q(f_x \mid y)\, df_x$$

Gaussian distributions play a fundamental role in probabilistic reasoning,
similar to how linear functions play a fundamental role in algebra.
Gaussian models provide a flexible and powerful modeling language.
With approximate inference, they even scale to non-Gaussian observations.

But Gaussians do not solve *all* problems. Some open questions for the coming lectures:

- We've seen fully connected models and Markov Chains. Is there something in between? What is the *complexity* of inference?
- We've seen that even discrete observations can be described by latent Gaussians. But what if the *latent* space is discrete? How can we efficiently keep track of fundamentally dissimilar competing hypotheses?
- The numerical approximations we have encountered so far are cheap, but full of flaws.
  - **maximum a-posteriori** has no uncertainty!
  - **Laplace approximation** can be arbitrarily wrong!
  - **numerical integration** does not scale to many dimensions!

Can we do better?