<table>
<thead>
<tr>
<th>#</th>
<th>date</th>
<th>content</th>
<th>Ex</th>
<th>#</th>
<th>date</th>
<th>content</th>
<th>Ex</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20.04.</td>
<td>Introduction</td>
<td></td>
<td>14</td>
<td>09.06.</td>
<td>Logistic Regression</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>21.04.</td>
<td>Reasoning under Uncertainty</td>
<td></td>
<td>15</td>
<td>15.06.</td>
<td>Exponential Families</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>27.04.</td>
<td>Continuous Variables</td>
<td></td>
<td>16</td>
<td>16.06.</td>
<td>Graphical Models</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>28.04.</td>
<td>Monte Carlo</td>
<td></td>
<td>17</td>
<td>22.06.</td>
<td>Factor Graphs</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>04.05.</td>
<td>Markov Chain Monte Carlo</td>
<td></td>
<td>18</td>
<td>23.06.</td>
<td>The Sum-Product Algorithm</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>05.05.</td>
<td>Gaussian Distributions</td>
<td></td>
<td>19</td>
<td>29.06.</td>
<td>Example: Topic Models</td>
<td>11</td>
</tr>
<tr>
<td>7</td>
<td>11.05.</td>
<td>Parametric Regression</td>
<td></td>
<td>20</td>
<td>30.06.</td>
<td>Mixture Models</td>
<td>11</td>
</tr>
<tr>
<td>8</td>
<td>12.05.</td>
<td>Learning Representations</td>
<td></td>
<td>21</td>
<td>06.07.</td>
<td>EM</td>
<td>12</td>
</tr>
<tr>
<td>9</td>
<td>18.05.</td>
<td>Gaussian Processes</td>
<td></td>
<td>22</td>
<td>07.07.</td>
<td>Variational Inference</td>
<td>12</td>
</tr>
<tr>
<td>11</td>
<td>25.05.</td>
<td>An Example for GP Regression</td>
<td></td>
<td>24</td>
<td>14.07.</td>
<td>Example: Inferring Topics</td>
<td>13</td>
</tr>
<tr>
<td>13</td>
<td>08.06.</td>
<td>GP Classification</td>
<td></td>
<td>26</td>
<td>21.07.</td>
<td>Revision</td>
<td>13</td>
</tr>
</tbody>
</table>

Probabilistic ML — P. Hennig, SS 2021 — Lecture 12: Gauss-Markov Models — © Philipp Hennig, 2021 CC BY-NC-SA 3.0
Recap: Atomic Independence Structures
From Lecture 2

For uni- and bi-variate graphs, conditional independence is trivial.
For tri-variate sub-graphs, there are three possible structures:

<table>
<thead>
<tr>
<th>graph</th>
<th>factorization</th>
<th>implications</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i)</td>
<td>$P(A, B, C) = P(C \mid B) \cdot P(B \mid A) \cdot P(A)$</td>
<td>$A \perp!!!\perp C \mid B$ but not, i.g., $A \not!!!\perp C$</td>
</tr>
<tr>
<td>(ii)</td>
<td>$P(A, B, C) = P(A \mid B) \cdot P(C \mid B) \cdot P(B)$</td>
<td>$A \perp!!!\perp C \mid B$ but not, i.g., $A \not!!!\perp C$</td>
</tr>
<tr>
<td>(iii)</td>
<td>$P(A, B, C) = P(B \mid A, C) \cdot P(C) \cdot P(A)$</td>
<td>$A \perp!!!\perp C$ but not, i.g., $A \not!!!\perp C \mid B$</td>
</tr>
</tbody>
</table>
Graphical View: Parametric Model

Conditional independence of data given model weights

\[
p(f) = \mathcal{GP}(f; 0, \Phi_X^\top \Sigma \Phi_X) \quad p \left( \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \bigg| w \right) = \prod_i \delta(f_i - \phi_i^\top w) \quad p(y \mid f) = \prod_i \mathcal{N}(y_i; f_i, \sigma^2)
\]
Graphical View: Nonparametric Model

Fully connected graph

\[ p(f) = \mathcal{GP}(f; 0, k) \]

\[
p \left( \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \right) = \mathcal{N} \left( \begin{bmatrix} K_{11}^{-1} & K_{12}^{-1} & K_{13}^{-1} & K_{14}^{-1} \\ K_{12}^{-1} & K_{22}^{-1} & K_{23}^{-1} & K_{24}^{-1} \\ K_{13}^{-1} & K_{23}^{-1} & K_{33}^{-1} & K_{34}^{-1} \\ K_{14}^{-1} & K_{24}^{-1} & K_{34}^{-1} & K_{44}^{-1} \end{bmatrix} \right)^{-1} \]

\[ p(y | f) = \prod_i \mathcal{N}(y_i; f_i, \sigma^2) \]
Definition

A time series is a sequence \([y(t_i)]_{i \in \mathbb{N}}\) of observations \(y_i := x(t_i) \in \mathbb{Y}\), indexed by a scalar variable \(t \in \mathbb{R}\). In many applications, the time points \(t_i\) are equally spaced: \(t_i = t_0 + i \cdot \delta t\). Models that account for all values \(t \in \mathbb{R}\) are called continuous time, while models that only consider \([t_i]_{i \in \mathbb{N}}\) are called discrete time.

Examples:

- climate & weather observations … in Climate Science
- sensor readings in cars, … in Engineering
- EEG, ECG, patch clamp signals, … in Medicine and Neuroscience
- just about any sensing of a dynamical process in Physics
- stock prices, supply & demand data, polling numbers, … in Economics and Social Science
- body weight measurements in the previous lecture

Inference in time series often has to happen in real-time, and scale to an unbounded set of data, typically on small-scale or embedded systems. So it has to be of (low) constant time and memory complexity.
Markov Chains
Processes with a “local memory”

\[ p(f) = \mathcal{GP}(f; 0, k) \]

\[
p \left( \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \right) = \mathcal{N} \left( 0, \begin{bmatrix} K_{11}^{-1} & K_{12}^{-1} & 0 & 0 \\ K_{12}^{-1} & K_{22}^{-1} & 0 & 0 \\ 0 & 0 & K_{33}^{-1} & K_{34}^{-1} \\ 0 & 0 & K_{34}^{-1} & K_{44}^{-1} \end{bmatrix} \right)^{-1} \]

\[ p(y | f) = \prod_i \mathcal{N}(y_i; f_i, \sigma^2) \]
It’s all about (Conditional) Independence

This point is way to easily missed

A generalization of the law of large numbers to variables that depend on each other.
Proceedings of the Society for Physics and Mathematics at Kazan University, 1906

Andrej Andreevič Markov
(1856 – 1922)
Man kommt also dazu, im Begriffe der Unabhängigkeit wenigstens den ersten Keim der eigenartigen Problematik der Wahrscheinlichkeitsrechnung zu erblicken — ein Umstand, welcher in diesem Buche nur wenig hervortreten wird, da wir hier hauptsächlich nur mit den logischen Vorbereitungen zu den eigentlichen wahrscheinlichkeitstheoretischen Untersuchungen zu tun haben werden.

Es ist dementsprechend eine der wichtigsten Aufgaben der Philosophie der Naturwissenschaften, nachdem sie die vielumstrittene Frage über das Wesen des Wahrscheinlichkeitsbegriffes selbst erklärt hat, die Voraussetzungen zu präzisieren, bei denen man irgendwelche gegebene reelle Erscheinungen für gegenseitig unabhängig halten kann. Diese Frage fällt allerdings aus dem Rahmen unseres Buches.
Cave: Change of Notation

- Previously: Observe $y \in \mathbb{R}^D$ at $N$ locations $x \in \mathbb{X}$, assume latent function $f \in \mathbb{R}^M$, and $y \approx Hf(x)$.
- The notion of a local finite memory only works in an ordered space of inputs. Thus, $\mathbb{X} \subset \mathbb{R}$.
- Now: Observe $y_1, \ldots, y_N$ with $y_i \in \mathbb{R}^D$ at times $[t_1, \ldots, t_N]$ with $t_i \in \mathbb{R}$.
  Assume latent state $x_i \in \mathbb{R}^M$, and $y_i \approx Hx(t_i)$. (The state will constitute the local memory)
- Such models are known as state-space models. (They are related to Finite-State Machines)

**Definition:** A joint distribution $p(X)$ over a sequence of random variables $X := [x_0, \ldots, x_N]$ is said to have the Markov property if

$$p(x_i \mid x_0, x_1, \ldots, x_{i-1}) = p(x_i \mid x_{i-1}).$$

The sequence is then called a Markov chain.
Markov Chains
Finite Memory through Conditional Independence

Assume:

\[ p(x_t \mid X_{0:t-1}) = p(x_t \mid x_{t-1}) \]

and

\[ p(y_t \mid X) = p(y_t \mid x_t) \]

\[
p(x_t \mid Y_{0:t-1}) = \frac{\int_{j \neq t} p(x)p(Y_{0:t-1} \mid x) \, dx}{\int p(x)p(Y_{0:t-1} \mid x) \, dx} = \frac{\int_{j \neq t} p(Y_{0:t-1} \mid X_{0:t-1}) p(x_0) \left( \prod_{0 < j < t} p(x_j \mid x_{j-1}) \, dx_j \right) p(x_t \mid x_{t-1}) \left( \prod_{j > t} p(x_j \mid x_{j-1}) \, dx_j \right)}{\int p(Y_{0:t-1} \mid X_{0:t-1}) p(x_0) \left( \prod_{0 < j < t} p(x_j \mid x_{j-1}) \, dx_j \right) \, dx}
\]

\[
= \frac{\int_{j < t} p(x_t \mid x_{t-1}) p(Y_{0:t-1} \mid X_{0:t-1}) p(x_0) \left( \prod_{0 < j < t} p(x_j \mid x_{j-1}) \, dx_j \right)}{\int_{j \leq t} p(Y_{0:t-1} \mid X_{0:t-1}) p(x_0) \left( \prod_{0 < j < t} p(x_j \mid x_{j-1}) \, dx_j \right)} = \int p(x_t \mid x_{t-1}) p(x_{t-1} \mid Y_{0:t-1}) \, dx_{t-1}
\]
Assume:

\[ p(x_t \mid X_{0:t-1}) = p(x_t \mid x_{t-1}) \]
and

\[ p(y_t \mid X) = p(y_t \mid x_t) \]

If you believe the graph, though, just note that the joint is

\[ p(x_t, x_{t-1} \mid y_{1:t-1}) = p(x_t \mid x_{t-1}, y_{1:t-1})p(x_{t-1} \mid y_{1:t-1}) = p(x_t \mid x_{t-1})p(x_{t-1} \mid y_{1:t-1}) \]

which we can integrate over \( x_{t-1} \) to obtain the Chapman-Kolmogorov equation

\[ p(x_t \mid y_{1:t-1}) = \int p(x_t \mid x_{t-1})p(x_{t-1} \mid y_{1:t-1})dx_{t-1}. \]
Assume:

\[ p(x_t \mid X_{0:t-1}) = p(x_t \mid x_{t-1}) \]

and

\[ p(y_t \mid X) = p(y_t \mid x_t) \]

\[
p(x_t \mid Y_{0:t}) = \frac{p(y_t \mid x_t)p(x_t \mid Y_{0:t-1})}{\int p(y_t \mid x_t)p(x_t \mid Y_{0:t-1}) \, dx_t}
\]
Markov Chains
Finite Memory through Conditional Independence

Assume:

\[ p(x_t \mid X_{0:t-1}) = p(x_t \mid x_{t-1}) \]
and

\[ p(y_t \mid X) = p(y_t \mid x_t) \]

\[
\begin{align*}
p(x_t \mid Y) &= \int p(x_t, x_{t+1} \mid Y) \, dx_{t+1} = \int p(x_t \mid x_{t+1}, Y) p(x_{t+1} \mid Y) \, dx_{t+1} \\
p(x_t \mid x_{t+1}, Y) &= \frac{p(Y_{t+1:n} \mid x_{t+1}, x_t, Y_{0:t}) p(x_t \mid x_{t+1}, Y_{0:t})}{\int p(Y_{t+1:n} \mid x_{t+1}, x_t, Y_{0:t}) p(x_t \mid x_{t+1}, Y_{0:t}) \, dx_t} = \frac{p(Y_{t+1:n} \mid x_{t+1}, Y_{0:t}) \cdot p(x_t \mid x_{t+1}, Y_{0:t})}{p(Y_{t+1:n} \mid x_{t+1}, Y_{0:t}) \cdot \int p(x_t \mid x_{t+1}, Y_{0:t}) \, dx_t} = p(x_t \mid x_{t+1}, Y_{0:t}) \\
p(x_t \mid x_{t+1}, Y_{0:t}) &= \frac{p(x_{t+1} \mid Y_{0:t})}{p(x_{t+1} \mid Y_{0:t})} = \frac{p(x_t \mid x_{t+1}, Y_{0:t}) p(x_t \mid Y_{0:t})}{p(x_{t+1} \mid Y_{0:t})} = \frac{p(x_{t+1} \mid x_t) p(x_t \mid Y_{0:t})}{p(x_{t+1} \mid Y_{0:t})} \\
p(x_t \mid Y) &= p(x_t \mid Y_{0:t}) \int p(x_{t+1} \mid x_t) \frac{p(x_{t+1} \mid Y)}{p(x_{t+1} \mid Y_{0:t})} \, dx_{t+1}
\end{align*}
\]
Assume:

\[ p(x_t \mid X_{0:t-1}) = p(x_t \mid x_{t-1}) \]

and

\[ p(y_t \mid X) = p(y_t \mid x_t) \]

Filtering: \( O(T) \)

**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: \( O(T) \)

**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: $O(T)$

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

  **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: $O(T)$

  **smooth:**
  \[
  p(x_t \mid Y) = p(x_t \mid Y_{0:t}) \int p(x_{t+1} \mid x_t) \frac{p(x_{t+1} \mid Y)}{p(x_{t+1} \mid Y_{0:t})} \, dx_{t+1}
  \]
procedure INFERENCE$(Y, p(x_0), p(x_t | x_{t-1}) \, \forall t, p(y_t | x_t) \, \forall t)$

for $i=1,...,n$ do
  
  $p(x_t | y_{1:t-1}) = \int p(x_t | x_{t-1})p(x_{t-1} | Y_{0:t-1}) \, dx_{t-1}$  
  
  $p(x_t | y_{1:t}) = p(y_t | x_t)p(x_t | Y_{0:t-1})/p(y_t)$

end for

for $i=n-1,...,0$ do
  
  $p(x_t | Y) = p(x_t | Y_{0:t}) \int p(x_{t+1} | x_t)p(x_{t+1} | Y)p(x_{t+1} | Y_{1:t}) \, dx_{t+1}$

end for

return $p(x_t | Y) \, \forall t = 0, \ldots, n$

end procedure
Gauss-Markov Models

Local structure for univariate Gaussian models

\[ p(x(t_{i+1}) \mid X_{1:i}) = \mathcal{N}(x_{i+1}; Ax_i, Q) \quad \text{and} \quad p(x_0) = \mathcal{N}(x_0; m_0, P_0) \quad \text{and} \quad p(y_i \mid X) = \mathcal{N}(y_i; Hx_i, R) \]

**predict:**  
\[ p(x_t \mid Y_{1:t-1}) = \int p(x_t \mid x_{t-1}) p(x_{t-1} \mid Y_{1:t-1}) \, dx_{t-1} \]

\[ = \int \mathcal{N}(x_t; Ax_{t-1}, Q) \cdot \mathcal{N}(x_{t-1}; m_{t-1}, P_{t-1}) \, dx_{t-1} \]

\[ = \mathcal{N}(x_t, Am_{t-1}, AP_{t-1}A^\top + Q) \]

\[ = \mathcal{N}(x_t, m_t^-, P_t^-) \]
Gauss-Markov Models

Local structure for univariate Gaussian models

\[ p(x(t_{i+1}) \mid X_{1:i}) = \mathcal{N}(x_{i+1}; Ax_i, Q) \quad \text{and} \quad p(x_0) = \mathcal{N}(x_0; m_0, P_0) \quad \text{and} \quad p(y_i \mid X) = \mathcal{N}(y_i; Hx_i, R) \]

update:

\[
p(x_t \mid Y_{1:t}) = \frac{p(y_t \mid x_t)p(x_t \mid Y_{1:t-1})}{p(y_t)} = \frac{\mathcal{N}(y_t; Hx_t, R)\mathcal{N}(x_t; m_t^-, P_t^-)}{\mathcal{N}(y_t; Hm_t^-, HP_t^- H^\top)} = \mathcal{N}(x_t, m_t^+ + Kz, (I - KH)P_t^-) = \mathcal{N}(x_t, m_t, P_t) \quad \text{where}
\]

\[
K := P_t^- H^\top (HP_t^\top H + R)^{-1}, \quad (\text{gain})
\]

\[
z := y_t - Hm_t^- \quad (\text{residual})
\]
Gauss-Markov Models

Local structure for univariate Gaussian models

\[ p(x(t_{i+1}) \mid X_{1:i}) = \mathcal{N}(x_{i+1}; Ax_i, Q) \quad \text{and} \quad p(x_0) = \mathcal{N}(x_0; m_0, P_0) \quad \text{and} \quad p(y_i \mid X) = \mathcal{N}(y_i; Hx_i, R) \]

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

\[ = \mathcal{N}(x_t; m_t, P_t) \int \mathcal{N}(x_{t+1}, Ax_t, Q) \frac{\mathcal{N}(x_{t+1}; m_{t+1}^s, P_{t+1}^s)}{\mathcal{N}(x_{t+1}; m_{t+1}, P_{t+1})} \, dx_{t+1} \]

\[ = \mathcal{N}(x_t, m_t + G_t(m_{t+1}^s - m_{t+1}^-), P_t + G_t(P_{t+1}^s - P_{t+1}^-)G_t^T) \]

\[ = \mathcal{N}(x_t, m_t^s, P_t^s) \quad \text{where} \]

\[ G_t := P_tA^T(P_t^-)^{-1} \quad \text{(smoother gain)} \]
Gauss-Markov Models

Local structure for univariate Gaussian models

\[ p(x(t_{i+1}) \mid X_{1:i}) = \mathcal{N}(x_{i+1}; Ax_i, Q) \quad \text{and} \quad p(x_0) = \mathcal{N}(x_0; m_0, P_0) \quad \text{and} \quad p(y_i \mid X) = \mathcal{N}(y_i; Hx_i, R) \]

(Kalman) Filter:

\[
\begin{align*}
p(x_t) &= \mathcal{N}(x_t; m_t^-, P_t^-) \\
m_t^- &= Am_{t-1} \\
P_t^- &= AP_{t-1}A^T + Q \\
p(x_t \mid y_t) &= \mathcal{N}(x_t; m_t, P_t) \\
z_t &= y_t - Hm_t^- \\
S_t &= HP_t^-H^T + R \\
K_t &= P_t^-H^T S^{-1} \\
m_t &= m_t^- + Kz_t \\
P_t &= (I - KH)P_t^-
\end{align*}
\]

with predictive mean and predictive covariance

(Rauch Tung Striebel) Smoother:

\[
\begin{align*}
p(x_t \mid Y) &= \mathcal{N}(x_t; m_t^s, P_t^s) \\
m_t^s &= m_t + G_t(m_t^s + m_t^- - m_{t+1}^-) \\
P_t^s &= P_t + G_t(P_t^s - P_t^-)G^T
\end{align*}
\]

with RTS gain and smoothed mean and smoothed covariance

Probabilistic ML — P. Hennig, SS 2021 — Lecture 12: Gauss-Markov Models — © Philipp Hennig, 2021 CC BY-NC-SA 3.0
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. If all relationships are *linear* and *Gaussian*,

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

then inference is analytic and given by the Kalman Filter and the Rauch-Tung-Striebel Smoother:

**(Kalman) Filter:**

\[
p(x_t) = \mathcal{N}(x_t; m_t^-, P_t^-)
\]

with

\[
m_t^- = Am_{t-1}
\]

predictive mean

\[
P_t^- = AP_t^{-1}A^T + Q
\]

predictive covariance

\[
p(x_t \mid y_t) = \mathcal{N}(x_t; m_t, P_t)
\]

with

\[
z_t = y_t - Hm_t^-
\]

innovation residual

\[
S_t = HP_t^-H^T + R
\]

innovation covariance

\[
K_t = P_t^-H^T S^{-1}
\]

Kalman gain

\[
m_t = m_t^- + Kz_t
\]

estimation mean

\[
P_t = (I - KH)P_t^-
\]

estimation covariance

**Rauch Tung Striebel Smoother:**

\[
p(x_t \mid Y) = \mathcal{N}(x_t; m_s^t, P_s^t)
\]

with

\[
G_t = P_t A^T (P_{t+1}^-)^{-1}
\]

RTS gain

\[
m_s^t = m_t + G_t(m_s^{t+1} - m_t^{t+1})
\]

smoothed mean

\[
P_s^t = P_t + G_t(P_{t+1}^{s} - P_{t+1}^-)G_t^T
\]

smoothed covariance
Continuous Time

Differential equations defining non-differential curves

\[ \delta t = 1 \quad Q_{\delta t} = 1 \]
Continuous Time

Differential equations defining non-differential curves

\[ \delta t = \frac{1}{2} \quad Q_{\delta t} = \frac{1}{2} \]
Continuous Time

Differential equations defining non-differential curves

\[ \delta t = \frac{1}{4} \quad Q_\delta t = \delta t \]
Continuous Time

Differential equations defining non-differential curves

For the limit $\delta t \to 0$ we would like to encode that $Q_{\delta t}$ 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.)
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

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

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

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

Some well-studied examples

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

\[ \mathbb{E}(x(t)) =: m(t) = e^{F(t-t_0)}x_0 \]

\[ \text{cov}(x(t_a), x(t_b)) =: k(t_a, t_b) = \int_{t_0}^{\min(t_a, t_b)} e^{F(t_a - \tau)}LL^T e^{F(t_b - \tau)} \, d\tau \]

\[ A_{t_i} = e^{F(t_{i+1} - t_i)} \]

\[ Q_{t_i} = \int_{0}^{t_{i+1} - t_i} e^{F\tau}LL^T e^{F\tau} \, d\tau \]
The Connection to GPs

Some well-studied examples

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

\[ \mathbb{E}(x(t)) =: m(t) = e^{F(t-t_0)}x_0 \]

\[ \text{cov}(x(t_a), x(t_b)) =: k(t_a, t_b) = \int_{t_0}^{\min{t_a,t_b}} e^{F(t_\tau-t_0)}LL^T e^{F(t_b-t_\tau)} \, d\tau \]

\[ A_{ti} = e^{F(t_{i+1}-t_i)} \]

\[ Q_{ti} = \int_0^{t_{i+1}-t_i} e^{F\tau}LL^T e^{F\tau} \, d\tau \]

The scaled Wiener process

\[ F = 0, \, L = \theta \quad \Rightarrow \quad m(t) = x_0 \quad k(t_a, t_b) = \theta^2 (\min(t_a, t_b) - t_0) \]

\[ A = I \quad Q_{ti} = \theta^2 (t_{i+1} - t_i) \]
The Connection to GPs

Some well-studied examples

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

\[ \mathbb{E}(x(t)) =: m(t) = e^{F(t-t_0)} x_0 \]

\[ \text{cov}(x(t_a), x(t_b)) =: k(t_a, t_b) = \int_{t_0}^{\min t_a, t_b} e^{F(t_a - \tau)} L L^\top e^{F^\top (t_b - \tau)} \, d\tau \]

\[ A_t = e^{F(t+1-t)} \]

\[ Q_t = \int_0^{t+1-t} e^{F \tau} L L^\top e^{F \tau} \, d\tau \]

The Ornstein-Uhlenbeck process

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

\[ A = e^{-\delta_t/\lambda} \]

\[ Q_t = \theta^2 \left( 1 - e^{-2\delta_t/\lambda} \right) \]
Non-Scalar State-Space Models

Integrators and Polynomial Splines

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

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

\[
\begin{bmatrix}
  x(1) \\
  x(2)
\end{bmatrix}
\begin{bmatrix}
  0 & 1 \\
  0 & 0
\end{bmatrix}
\begin{bmatrix}
  0 \\
  1
\end{bmatrix}
\]

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}
\Rightarrow
x(1)(t) = \int_{t_0}^{t} x(2)(t) \, dt + [x_0]_1
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, $O(N)$ in the number of datapoints (as opposed to $O(N^3)$ for general GPs).

(Although not shown, this includes hyperparameter inference!)

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