

Linear Dynamical Systems for Neuroscience

Joaquín Rapela and Bruno Cruz

Gatsby Computational Neuroscience Unit & Sainsbury Wellcome Centre
University College London

April 5, 2023

Contents

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 Introduction
- 2 Design of a linear dynamical system for tracking
- 3 Inference
 - Kalman filter
 - Kalman smoother
- 4 Derivation of Kalman filter equations
- 5 Remarks
- 6 Bruno Cruz: applications of the Kalman filter with Bonsai

Contents

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 Introduction
- 2 Design of a linear dynamical system for tracking
- 3 Inference
- 4 Derivation of Kalman filter equations
- 5 Remarks
- 6 Bruno Cruz: applications of the Kalman filter with Bonsai

Linear dynamical systems

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

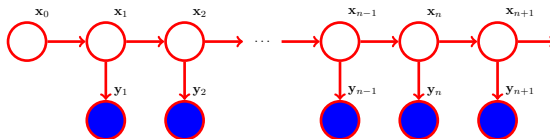
Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References



Linear dynamical systems

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

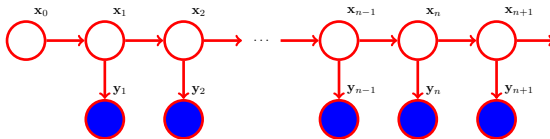
Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References



Linear dynamical systems

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

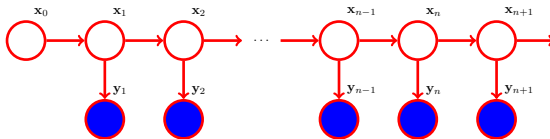
Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

true planes positions, velocities and accelerations



The linear dynamical system model

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

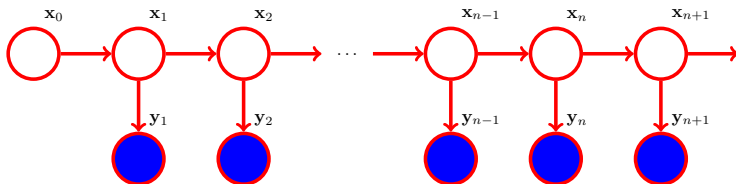
References

$$\mathbf{x}_n = A\mathbf{x}_{n-1} + \mathbf{w}_n \quad \mathbf{w}_n \sim N(\mathbf{w}_n | \mathbf{0}, Q) \quad \mathbf{x}_n \in \mathbb{R}^M$$

$$\mathbf{y}_n = C\mathbf{x}_n + \mathbf{v}_n \quad \mathbf{v}_n \sim N(\mathbf{v}_n | \mathbf{0}, R) \quad \mathbf{y}_n \in \mathbb{R}^P \quad n = 1 \dots N$$

$$\mathbf{x}_0 \sim N(\mathbf{x}_0 | \mathbf{m}_0, V_0)$$

$\{\mathbf{x}_0, \mathbf{v}_n, \mathbf{w}_n\}_{n=1}^N$ uncorrelated



$\{\mathbf{x}_0, \mathbf{x}_n, \mathbf{y}_n\}_{n=1}^N$ jointly normal

Some applications of linear dynamical systems in engineering

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

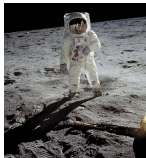
Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- Apollo mission ([McGee and Schmidt, 1985](#)),



Edwin Aldrin, July 20, 1969

- aircraft guidance ([Schmidt et al., 1970](#)),
- GPS navigation ([Hofmann-Wellenhof and Lichtenegger, 1997](#)),
- weather forecasting ([Buehner et al., 2017](#)).

Some applications of linear dynamical systems in engineering

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

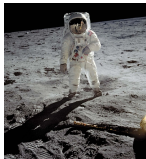
Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- Apollo mission ([McGee and Schmidt, 1985](#)),



Edwin Aldrin, July 20, 1969

- aircraft guidance ([Schmidt et al., 1970](#)),
- GPS navigation ([Hofmann-Wellenhof and Lichtenegger, 1997](#)),
- weather forecasting ([Buehner et al., 2017](#)).

Rudolf Emil Kalman received the 2008 Draper Prize from the National Academy of Engineering “*for the development and dissemination of the optimal digital technique (known as the Kalman filter) that is pervasively used to control a vast array of consumer, health, commercial, and defence products.*”

Example application of linear dynamical systems in neuroscience

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

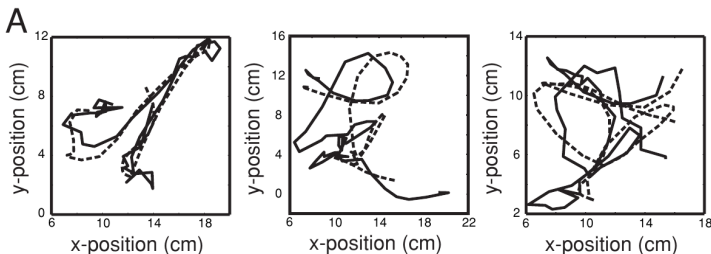
Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Decoding hand kinematics from motor cortex neural activity (Wu et al., 2006). **Observation** (y_n): neural activity recorded from motor cortex of monkeys. **State** (x_n): hand position, velocity and acceleration.



Wu et al. (2006)

SWC: inferring mouse kinematics from video recordings.

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 **Observation** (y_n): noisy position estimates from DeepLabCut.
- 2 **State** (x_n): mouse position, velocity and acceleration.

joint work with Foraging Working Group

SWC: inferring low-dimensional representations of neural population spiking activity.

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

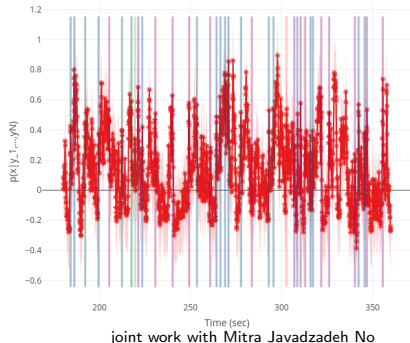
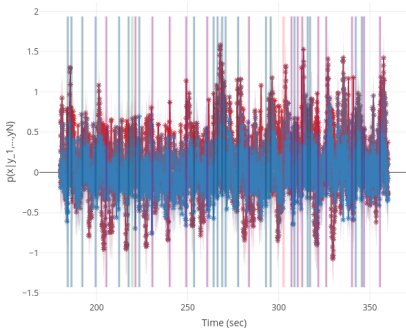
Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 **Observation** (y_n): spike counts in 100 ms bins from neural population.
- 2 **State** (x_n): low-dimensional representation of neural population spikes.



Contents

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 Introduction
- 2 Design of a linear dynamical system for tracking
- 3 Inference
- 4 Derivation of Kalman filter equations
- 5 Remarks
- 6 Bruno Cruz: applications of the Kalman filter with Bonsai

Discrete Wiener Process Acceleration (DWPA) model for tracking an object moving in 1D

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

$$\begin{aligned}\mathbf{x}_n &= A\mathbf{x}_{n-1} + \mathbf{w}_n \quad \text{with} \quad \mathbf{w}_n \sim N(\mathbf{0}, Q) \quad \text{and} \quad \mathbf{x}_0 \sim N(\mathbf{m}_0, V_0) \\ y_n &= C\mathbf{x}_n + v_n \quad \text{with} \quad v_n \sim N(0, \sigma^2)\end{aligned}$$

$$\mathbf{x}_n = [\xi[n], \dot{\xi}[n], \ddot{\xi}[n]]^\top$$

$$A = \begin{bmatrix} 1 & T & T^2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}$$

$$Q = \gamma^2 \begin{bmatrix} \frac{1}{4}T^4 & \frac{1}{2}T^3 & \frac{1}{2}T^2 \\ \frac{1}{2}T^3 & T^2 & T \\ \frac{1}{2}T^2 & T & 1 \end{bmatrix}$$

$$C = [1 \quad 0 \quad 0]$$

Derivation of DWPA model for tracking

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Consider the Taylor series expansion of the position as a function of time, $\xi(t)$, up to second order.

$$\xi(t + T) = \xi(t) + \dot{\xi}(t)T + \frac{\ddot{\xi}(t)}{2}T^2 \quad (1)$$

Approximations of the velocity and acceleration are derived from Eq. 1 by successive differentiation (with respect to T).

$$\dot{\xi}(t + T) = \dot{\xi}(t) + \ddot{\xi}(t)T \quad (2)$$

$$\ddot{\xi}(t + T) = \ddot{\xi}(t) \quad (3)$$

Derivation of DWPA model for tracking

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

According to Eq. 3 the approximation of the acceleration, $\ddot{\xi}(t)$ is constant across all times. The DWPA model generalises this by replacing it with Eq. 4.

$$\ddot{\xi}_a(t) = \ddot{\xi}(kT) + v(k) \quad t \in [kT, (k+1)T) \quad (4)$$

Replacing $\ddot{\xi}_a(t)$ by $\ddot{\xi}(t)$ in Eqs. 1, 2 and 3 and discretizing we obtain the motion equations for the DWPA model.

$$\xi(k+1) = \xi(k) + \dot{\xi}(k)T + \frac{\ddot{\xi}(k)}{2}T^2 + \frac{v(k)}{2}T^2 \quad (5)$$

$$\dot{\xi}(k+1) = \dot{\xi}(k) + \ddot{\xi}(k)T + v(k)T \quad (6)$$

$$\ddot{\xi}(k+1) = \ddot{\xi}(k) + v(k) \quad (7)$$

Derivation of DWPA model for tracking

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Calling $x(k) = [\xi(k), \dot{\xi}(k), \ddot{\xi}(k)]^\top$, Eq. 8 rewrites the previous equations in matrix form.

$$\begin{aligned} x(k) &= \begin{bmatrix} 1 & T & T^2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} x(k-1) + \begin{bmatrix} \frac{1}{2} T^2 \\ T \\ 1 \end{bmatrix} v(k) \\ &= Ax(k-1) + \Gamma v(k) \\ &= Ax(k-1) + w(k) \end{aligned} \tag{8}$$

We still need to find the covariance of $w(k)$.

Derivation of DWPA model for tracking

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Because $v(k) \sim \mathcal{N}(0, \gamma^2)$ then $w(k)$ is also Gaussian with mean zero and covariance Q (Eq. 9).

$$\begin{aligned} E\{w(k)\} &= \Gamma E\{v(k)\} = 0 \\ E\{w(k)w(k)^\top\} &= \Gamma E\{v(k)^2\} \Gamma^\top = \Gamma \gamma^2 \Gamma^\top = \gamma^2 \Gamma \Gamma^\top \\ &= \gamma^2 \begin{bmatrix} \frac{1}{2} T^2 \\ T \\ 1 \end{bmatrix} \begin{bmatrix} \frac{1}{2} T^2, T, 1 \end{bmatrix} \\ &= \gamma^2 \begin{bmatrix} \frac{1}{4} T^4 & \frac{1}{2} T^3 & \frac{1}{2} T^2 \\ \frac{1}{2} T^3 & T^2 & T \\ \frac{1}{2} T^2 & T & 1 \end{bmatrix} = Q \end{aligned} \quad (9)$$

Practical: sample from a DWPA model for tracking

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter

Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

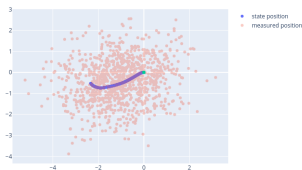
References

$$\mathbf{x}_n = [\xi_x[n], \dot{\xi}_x[n], \ddot{\xi}_x[n], \xi_y[n], \dot{\xi}_y[n], \ddot{\xi}_y[n]]$$

$$\mathbf{x}_n = \mathbf{A}\mathbf{x}_{n-1} + \mathbf{w}_n, \quad \mathbf{w}_n \sim \mathcal{N}(\mathbf{0}, Q)$$

$$\mathbf{y}_n = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \mathbf{x}_n + \mathbf{v}_n, \quad \mathbf{v}_n \sim \mathcal{N}(\mathbf{0}, R)$$

$$\mathbf{x}_0 \sim \mathcal{N}(\mathbf{m}_0, V_0)$$



code

Contents

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 Introduction
- 2 Design of a linear dynamical system for tracking
- 3 **Inference**
- 4 Derivation of Kalman filter equations
- 5 Remarks
- 6 Bruno Cruz: applications of the Kalman filter with Bonsai

Inference in LDSs

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Prediction

$$P(\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) = N(\mathbf{x}_n | \mathbf{x}_{n|n-1}, P_{n|n-1})$$

Filtering

$$P(\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_n) = N(\mathbf{x}_n | \mathbf{x}_{n|n}, P_{n|n})$$

Smoothing

$$P(\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_N) = N(\mathbf{x}_n | \mathbf{x}_{n|N}, P_{n|N})$$

Inference in LDSs

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Prediction

$$P(\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) = N(\mathbf{x}_n | \mathbf{x}_{n|n-1}, P_{n|n-1})$$

Filtering

$$P(\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_n) = N(\mathbf{x}_n | \mathbf{x}_{n|n}, P_{n|n})$$

Smoothing

$$P(\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_N) = N(\mathbf{x}_n | \mathbf{x}_{n|N}, P_{n|N})$$

These conditional probabilities are Normal because
 $\{\mathbf{x}_0, \mathbf{x}_n, \mathbf{y}_n\}_{n=1}^N$ are jointly normal.

Kalman Filter

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

The Kalman filtering algorithm estimates the prediction and filtering means and covariances: $\mathbf{x}_{n|n-1}$, $P_{n|n-1}$, $\mathbf{x}_{n|n}$, $P_{n|n}$,

$$\mathbf{x}_{0|0} = \mathbf{m}_0$$

init filtered mean

$$P_{0|0} = V_0$$

init filtered covariance

$$\mathbf{x}_{n+1|n} = A\mathbf{x}_{n|n}$$

prediction mean

$$P_{n+1|n} = AP_{n|n}A^T + Q$$

prediction covariance

$$\mathbf{y}_{n|n-1} = C\mathbf{x}_{n|n-1}$$

predicted observation

$$\tilde{\mathbf{y}}_n = \mathbf{y}_n - \mathbf{y}_{n|n-1}$$

residual

$$S_n = CP_{n|n-1}C^T + R$$

residual covariance

$$\mathbf{x}_{n|n} = \mathbf{x}_{n|n-1} + K_n\tilde{\mathbf{y}}_n$$

filtering mean

$$K_n = P_{n|n-1}C^TS_n^{-1}$$

Kalman gain

$$P_{n|n} = (I_M - K_nC)P_{n|n-1}$$

filtering covariance

Steps in the Kalman Filter

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter

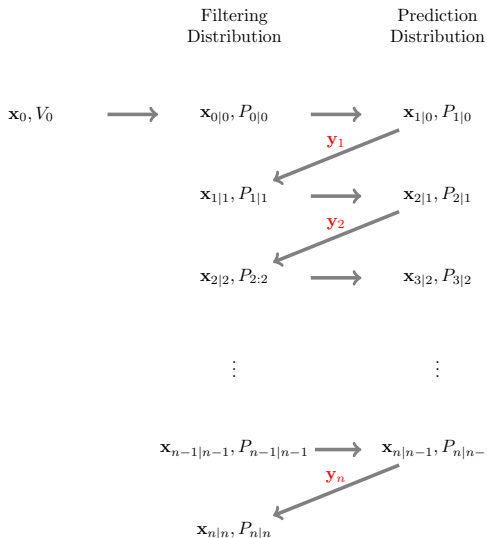
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References



Steps in the Kalman Filter

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter

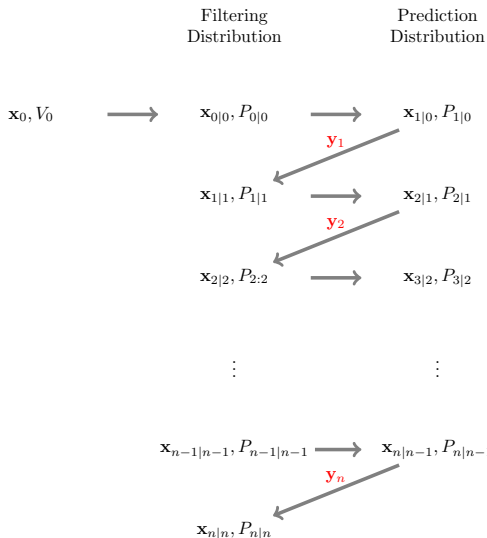
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References



forward, online

Practical: infer filtered positions

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

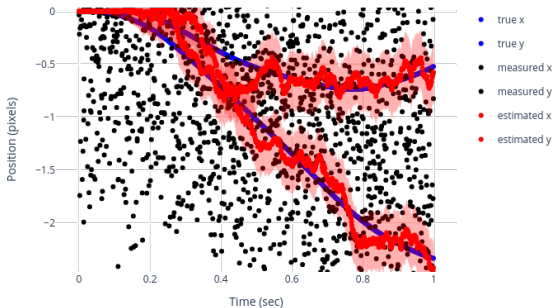
Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References



code

Kalman Smoother

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter

Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

The Kalman smoother algorithm estimates the smoothed means and covariances: $x_{n|N}, P_{n|N}$.

$$x_{n|N} = x_{n|n} + C_n(x_{n+1|N} - x_{n+1|n}) \quad \text{smoothed mean}$$

$$P_{n|N} = P_{n|n} + C_n(P_{n+1|N} - P_{n+1|n})C_n^T \quad \text{smoothed covariance}$$

$$C_n = P_{n|n}A^T P_{n+1|n}^{-1}$$

- The initial mean and covariances (i.e., $x_{N|N}, P_{N|N}$) are initialised from the last step of the Kalman filter.
- Smoothing proceeds in a **backward** fashion:
$$x_{N|N}, P_{N|N} \rightarrow x_{N-1|N}, P_{N-1|N} \rightarrow x_{N-2|N}, P_{N-2|N} \rightarrow \dots \rightarrow x_{1|N}, P_{1|N}$$
- Smoothing can only be calculated **offline**.

Practical: infer the smoothed positions

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter

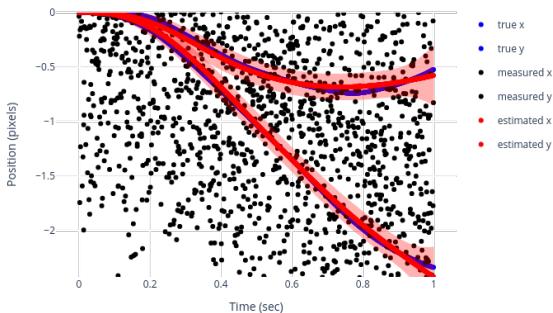
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References



code

Check that the state x position lies inside the 95% confidence ellipse 95% of the time

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Contents

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 Introduction
- 2 Design of a linear dynamical system for tracking
- 3 Inference
- 4 Derivation of Kalman filter equations
- 5 Remarks
- 6 Bruno Cruz: applications of the Kalman filter with Bonsai

Prediction mean

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Claim

$$\mathbf{x}_{n|n-1} = A\mathbf{x}_{n-1|n-1}$$

Proof.

$$\begin{aligned}\mathbf{x}_{n|n-1} &= E\{\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \\ &= E\{A\mathbf{x}_{n-1} + \mathbf{w}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \quad (10)\end{aligned}$$

$$= AE\{\mathbf{x}_{n-1} | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} + E\{\mathbf{w}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \quad (11)$$

$$= A\mathbf{x}_{n-1|n-1} + E\{\mathbf{w}_n\} \quad (12)$$

$$= A\mathbf{x}_{n-1|n-1} \quad (13)$$



Notes:

- Eq. 10 arises from the state equation of the LDS model,
- Eq. 11 holds because the expectation distributes over sums
- Eq. 12 uses the definition of $\mathbf{x}_{n-1|n-1}$ and the fact that the state noise, $\mathbf{w}_n$, is independent of previous observations.
- Eq. 13 follows due to the zero mean of $\mathbf{w}_n$.

Prediction covariance

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Claim

$$P_{n|n-1} = AP_{n-1|n-1}A^T + Q$$

Proof.

$$\begin{aligned} P_{n|n-1} &= \text{Cov}\{\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \\ &= \text{Cov}\{\mathbf{A}\mathbf{x}_{n-1} + \mathbf{w}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \end{aligned} \quad (14)$$

$$= \text{Cov}\{\mathbf{A}\mathbf{x}_{n-1} | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} + \text{Cov}\{\mathbf{w}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \quad (15)$$

$$= \mathbf{A} \text{Cov}\{\mathbf{x}_{n-1} | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \mathbf{A}^T + \text{Cov}\{\mathbf{w}_n\} \quad (16)$$

$$= \mathbf{A} P_{n-1|n-1} \mathbf{A}^T + Q \quad (17)$$



Notes:

- Eq. 14 used the state equation of the LDS model,
- Eq. 15 is true because $\mathbf{w}_n$ is independent from $\mathbf{x}_{n-1}$,
- Eq. 16 holds by the property $\text{Cov}\{\mathbf{A}\mathbf{x}\} = \mathbf{A} \text{Cov}\{\mathbf{x}\}' \mathbf{A}^T$ and because $\mathbf{w}_n$ is independent of previous observations.
- Eq. 17 applied the definitions of $P_{n-1|n-1}$ and Q .

Filtered mean and covariance

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Claim

$\mathbf{x}_{n|n} = \mathbf{x}_{n|n-1} + K_n \tilde{\mathbf{y}}_n$ and $P_{n|n} = (I - K_n C) P_{n|n-1}$.

Proof.

Define the random variables $\mathbf{x} = \mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}$ and $\mathbf{y} = \mathbf{y}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}$. Then $\mathbf{x} | \mathbf{y} = \mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_n$ and the mean and covariance that we want to find, $\mathbf{x}_{n|n}$ and $P_{n|n}$, are those of $\mathbf{x} | \mathbf{y}$. Thus, we want to compute the mean, $\mu_{\mathbf{x}|\mathbf{y}} = \mathbf{x}_{n|n}$, and covariance, $\Sigma_{\mathbf{x}|\mathbf{y}} = P_{n|n}$, of $\mathbf{x} | \mathbf{y}$.

Because $\mathbf{x}_n$ and $\mathbf{y}_n$ are jointly Gaussian, then $\mathbf{x}$ and $\mathbf{y}$ are also jointly Gaussian. Then, $\mu_{\mathbf{x}|\mathbf{y}}$ and $\Sigma_{\mathbf{x}|\mathbf{y}}$ are ([Bishop, 2016](#), Chapter 2)

$$\mu_{\mathbf{x}|\mathbf{y}} = \mu_{\mathbf{x}} + \Sigma_{\mathbf{xy}} \Sigma_{\mathbf{yy}}^{-1} (\mathbf{y}_n - \mu_{\mathbf{y}}) \quad (18)$$

$$\Sigma_{\mathbf{x}|\mathbf{y}} = \Sigma_{\mathbf{xx}} - \Sigma_{\mathbf{xy}} \Sigma_{\mathbf{yy}}^{-1} \Sigma_{\mathbf{yx}} \quad (19)$$

Thus, to compute $\mu_{\mathbf{x}|\mathbf{y}}$ and $\Sigma_{\mathbf{x}|\mathbf{y}}$ we need to calculate $\mu_{\mathbf{x}}$, $\mu_{\mathbf{y}}$, $\Sigma_{\mathbf{xx}}$, $\Sigma_{\mathbf{xy}}$ and $\Sigma_{\mathbf{yy}}$.

Filtered mean and covariance (cont)

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Proof.

$$\mu_{\mathbf{x}} = E\{\mathbf{x}\} = E\{\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} = \mathbf{x}_{n|n-1} \quad (20)$$

$$\begin{aligned} \mu_{\mathbf{y}} &= E\{\mathbf{y}\} = E\{\mathbf{y}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} = E\{C\mathbf{x}_n + \mathbf{v}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} = \\ &= CE\{\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} + E\{\mathbf{v}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} = C\mathbf{x}_{n|n-1} + E\{\mathbf{v}_n\} \\ &= C\mathbf{x}_{n|n-1} = \mathbf{y}_{n|n-1} \end{aligned} \quad (21)$$

Notes:

- The fifth equality in Eq. 21 uses the definition of $\mathbf{x}_{n|n-1}$ and the fact that $\mathbf{v}_n$ is independent of previous observations.
- The last equality in Eq. 21 holds because the mean of $\mathbf{v}_n$ is zero.

Filtered mean and covariance (cont)

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Proof.

$$\begin{aligned}\Sigma_{yy} &= \text{Cov}(\mathbf{y}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) \\ &= \text{Cov}(C\mathbf{x}_n + \mathbf{v}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) \\ &= \text{Cov}(C\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) + \text{Cov}(\mathbf{v}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) \\ &= C \text{Cov}(\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) C^T + \text{Cov}(\mathbf{v}_n)\end{aligned}\tag{22}$$

$$= CP_{n|n-1} C^T + R\tag{23}$$

Notes:

- As in Eq. 15, Eq. 22 holds by the property $\text{Cov}\{A\mathbf{x}\} = A \text{Cov}\{\mathbf{x}\} A^T$ and because $\mathbf{v}_n$ is independent of previous observations.

Filtered mean and covariance (cont)

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Proof.

$$\begin{aligned}\Sigma_{xy} &= \text{cCov}(\mathbf{x}_n, \mathbf{y}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) \\ &= \text{cCov}(\mathbf{x}_n, C\mathbf{x}_n + \mathbf{v}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) \\ &= \text{cCov}(\mathbf{x}_n, C\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) + \text{cCov}(\mathbf{x}_n, \mathbf{v}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1})\end{aligned}\quad (24)$$

$$= \text{Cov}(\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) C^T + 0 \quad (25)$$

$$= P_{n|n-1} C^T \quad (26)$$

Notes:

- the first term in Eq. 28 arises from the first term of Eq. 27 since

$$\begin{aligned}\text{cCov}(\mathbf{x}_n, C\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) &= E\{(\mathbf{x}_n - \mu_x)(C\mathbf{x}_n - C\mu_x)^T | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \\ &= E\{(\mathbf{x}_n - \mu_x)(\mathbf{x}_n - \mu_x)^T C^T | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \\ &= E\{(\mathbf{x}_n - \mu_x)(\mathbf{x}_n - \mu_x)^T | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} C^T \\ &= \text{Cov}(\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) C^T\end{aligned}$$

Filtered mean and covariance (cont)

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Proof.

$$\begin{aligned}\Sigma_{xy} &= \text{cCov}(\mathbf{x}_n, \mathbf{y}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) \\ &= \text{cCov}(\mathbf{x}_n, \mathbf{C}\mathbf{x}_n + \mathbf{v}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) \\ &= \text{cCov}(\mathbf{x}_n, \mathbf{C}\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) + \text{cCov}(\mathbf{x}_n, \mathbf{v}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1})\end{aligned}\quad (27)$$

$$= \text{Cov}(\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) \mathbf{C}^\top + 0 \quad (28)$$

$$= \mathbf{P}_{n|n-1} \mathbf{C}^\top \quad (29)$$

$$\Sigma_{xx} = \text{Cov}(\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) = \mathbf{P}_{n|n-1} \quad (30)$$

Notes:

- the first term in Eq. 28 arises from the first term of Eq. 27 since

$$\begin{aligned}\text{cCov}(\mathbf{x}_n, \mathbf{C}\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) &= E\{(\mathbf{x}_n - \mu_x)(\mathbf{C}\mathbf{x}_n - \mathbf{C}\mu_x)^\top | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \\ &= E\{(\mathbf{x}_n - \mu_x)(\mathbf{x}_n - \mu_x)^\top \mathbf{C}^\top | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \\ &= E\{(\mathbf{x}_n - \mu_x)(\mathbf{x}_n - \mu_x)^\top | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \mathbf{C}^\top \\ &= \text{Cov}(\mathbf{x}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) \mathbf{C}^\top\end{aligned}$$

Filtered mean and covariance (cont)

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- the second term in Eq. 28 arises from the second term of Eq. 27 since

$$\begin{aligned}\text{cCov}(\mathbf{x}_n, \mathbf{v}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}) &= E\{(\mathbf{x}_n - \mathbf{x}_{n|n-1})\mathbf{v}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \\ &= E\{(\mathbf{x}_n - \mathbf{x}_{n|n-1}) | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} E\{\mathbf{v}_n | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \\ &= E\{(\mathbf{x}_n - \mathbf{x}_{n|n-1}) | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} E\{\mathbf{v}_n\} \\ &= E\{(\mathbf{x}_n - \mathbf{x}_{n|n-1}) | \mathbf{y}_1, \dots, \mathbf{y}_{n-1}\} \mathbf{0} \\ &= \mathbf{0}\end{aligned}$$

the second line follows from the first one because $\mathbf{v}_n$ is independent of $\mathbf{x}_n$, and the third line follows from the second one because $\mathbf{v}_n$ is independent of previous observations.

Filtered mean and covariance (cont)

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Proof.

Having calculated μ_x , μ_y , Σ_{xx} , Σ_{xy} and Σ_{yy} we now use Eqs. 18, 19, 20, 21, 23, 29, and 30 to obtain $\mathbf{x}_{n|n}$ and $P_{n|n}$.

$$\begin{aligned}\mathbf{x}_{n|n} &= \mu_{x|y} = \mu_x + \Sigma_{xy} \Sigma_{yy}^{-1} (\mathbf{y}_n - \mu_y) \\ &= \mathbf{x}_{n|n-1} + P_{n|n-1} C^\top S_n^{-1} (\mathbf{y}_n - \mathbf{y}_{n|n-1}) \\ &= \mathbf{x}_{n|n-1} + K_n \tilde{\mathbf{y}}_n \\ P_{n|n} &= \Sigma_{x|y} = \Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx} \\ &= P_{n|n-1} - P_{n|n-1} C^\top S_n^{-1} C P_{n|n-1} \\ &= (I - P_{n|n-1} C^\top S_n^{-1} C) P_{n|n-1} \\ &= (I - K_n C) P_{n|n-1}\end{aligned}$$



Contents

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 Introduction
- 2 Design of a linear dynamical system for tracking
- 3 Inference
- 4 Derivation of Kalman filter equations
- 5 Remarks
- 6 Bruno Cruz: applications of the Kalman filter with Bonsai

Summary

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Today we learned

- 1 What a linear dynamical system (LDS) is,
- 2 How to design a LDS for tracking,
- 3 How to infer distributions of hidden states from measurements (i.e., Kalman filtering and smoothing),
- 4 How to implement LDSs in Python.

Summary

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Today we learned

- 1 What a linear dynamical system (LDS) is,
- 2 How to design a LDS for tracking,
- 3 How to infer distributions of hidden states from measurements (i.e., Kalman filtering and smoothing),
- 4 How to implement LDSs in Python.

Today we did NOT learn

- 1 how to learn parameters of LDSs,
- 2 non-linear and non-Gaussian extensions of LDSs (e.g., extended KF, particle filter, assumed density filter, Poisson LDS).

A few new concepts in this lecture wrt previous neuroinformatics ones

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

① behavioral data (vs ephys),

A few new concepts in this lecture wrt previous neuroinformatics ones

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 behavioral data (vs ephys),
- 2 time series models (vs iid assumption),

A few new concepts in this lecture wrt previous neuroinformatics ones

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 behavioral data (vs ephys),
- 2 time series models (vs iid assumption),
- 3 probabilistic (vs deterministic) modeling,

A few new concepts in this lecture wrt previous neuroinformatics ones

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 behavioral data (vs ephys),
- 2 time series models (vs iid assumption),
- 3 probabilistic (vs deterministic) modeling,
- 4 Bayesian (vs frequentist) statistics,

A few new concepts in this lecture wrt previous neuroinformatics ones

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 behavioral data (vs ephys),
- 2 time series models (vs iid assumption),
- 3 probabilistic (vs deterministic) modeling,
- 4 Bayesian (vs frequentist) statistics,
- 5 online (vs batch) algorithms.

Want to learn more?

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References



Durbin and Koopman (2012)

Contents

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

- 1 Introduction
- 2 Design of a linear dynamical system for tracking
- 3 Inference
- 4 Derivation of Kalman filter equations
- 5 Remarks
- 6 Bruno Cruz: applications of the Kalman filter with Bonsai

References

LDS for Neuro

Joaquín
Rapela and
Bruno Cruz

Introduction

Design of a
linear
dynamical
system for
tracking

Inference

Kalman filter
Kalman smoother

Derivation of
Kalman filter
equations

Remarks

Bruno Cruz:
applications of
the Kalman
filter with
Bonsai

References

Bishop, C. M. (2016). *Pattern recognition and machine learning*. Springer-Verlag New York.

Buehner, M., McTaggart-Cowan, R., and Heilliette, S. (2017). An ensemble kalman filter for numerical weather prediction based on variational data assimilation: Varenkf. *Monthly Weather Review*, 145(2):617–635.

Durbin, J. and Koopman, S. J. (2012). *Time series analysis by state space methods*, volume 38. OUP Oxford.

Hofmann-Wellenhof, B. and Lichtenegger, H. (1997). *J. collins gps theory and practice*.

McGee, L. A. and Schmidt, S. F. (1985). Discovery of the kalman filter as a practical tool for aerospace and industry. Technical report, NASA Technical Memorandum 86847.

Schmidt, S. F., Weinberg, J., and Lukesh, J. (1970). Application of kalman filtering to the c-5 guidance and control system. In *Theory and Applications of Kalman Filtering*, volume 139, pages 312–313. AGARD.

Wu, W., Gao, Y., Bienenstock, E., Donoghue, J. P., and Black, M. J. (2006). Bayesian population decoding of motor cortical activity using a kalman filter. *Neural computation*, 18(1):80–118.