State Space Models

♠

modeling, inferences and applications

Rong Chen
Rutgers University
(1.A) Introduction of State Space Models and many Examples

State Space Models

state equation: \[ x_t = g_t(x_{t-1}, \varepsilon_t) \quad \text{or} \quad x_t \sim q_t(\cdot \mid x_{t-1}) \]
observation equation: \[ y_t = h_t(x_t, e_t) \quad \text{or} \quad y_t \sim f_t(\cdot \mid x_t) \]

\[
\begin{array}{c}
\cdots \quad y_t \quad y_{t+1} \quad \cdots \\
\uparrow \quad \uparrow \\
\cdots \rightarrow x_t \rightarrow x_{t+1} \rightarrow \cdots \\
\end{array}
\]

\[ \pi_t(x_t) = p(x_1, \ldots, x_t \mid y_1, \ldots, y_t) \propto \prod_{s=1}^{t} f_s(y_s \mid x_s) q_s(x_s \mid x_{s-1}) \]

Objective: On-Line in Real Time

(1) Estimation: \[ p(x_t \mid y_1, \ldots, y_t) \]
(2) Prediction: \[ p(x_{t+1} \mid y_1, \ldots, y_t) \]
(3) Smoothing: \[ p(x_1, \ldots, x_{t-1} \mid y_1, \ldots, y_t) \]

(3.1) delayed estimation: \[ p(x_{t-d} \mid y_1, \ldots, y_t) \]
Linear and Gaussian Systems:

\[ x_t = H_t x_{t-1} + W_t w_t \]
\[ y_t = G_t x_t + V_t v_t \]

where \( w_t \sim N(0, I) \) and \( v_t \sim N(0, I) \).

\[ p(x_t \mid y_1, \ldots, y_t) \sim N(\mu_t, \Sigma_t) \]

Kalman Filter:

Recursive updating:

\[ (\mu_t, \Sigma_t) \rightarrow (\mu_{t+1}, \Sigma_{t+1}) \]

Very easy and fast!
Nonlinear and Non-Gaussian:

**Easy:**  $p(x_1, \ldots, x_t \mid y_t)$

**Difficult:**

$$E(x_t \mid y_t) = \int \cdots \int x_t p(x_1, \ldots, x_t \mid y_t) dx_1 \ldots dx_t$$

where $y_t = (y_1, \ldots, y_t)$.

**Our approach:** Monte Carlo method

Generate samples $x_t^{(1)}, \ldots, x_t^{(m)}$ from the target distribution $p(x_t \mid y_t)$, then use approximation

$$E[h(x_t) \mid y_t] \approx \frac{\sum_{i=1}^{m} h(x_t^{(i)})}{m}$$
Example 1: Target Tracking

A single target moving on a straight line with random (Gaussian) acceleration. State: $x_t = (d_t, v_t)$.

$T$ is the time duration between two observations. Constant acceleration within the period: $a_t = w_t/T$.

State Equation (motion model):

\[
\begin{align*}
d_t &= d_{t-1} + v_{t-1}T + w_t T/2, \quad w_t \sim N(0, q^2) \\
v_t &= v_{t-1} + w_t
\end{align*}
\]

Observation equation:

\[
\begin{align*}
z_t &= d_t + e_t, \quad e_t \sim N(0, r^2)
\end{align*}
\]

The system is linear and Gaussian. Objective: $p(x_t | z_1, \ldots, z_t)$. 
Example 2: Tracking a target in clutter

At time $t$, observe $m_t$ signals $y_t = (y_{t1}, \ldots, y_{tm_t})$, where

$$m_t \sim \text{Bernoulli}(p_d) + \text{Poisson}(\lambda\Delta)$$

The true signal $d_t + e_t$ has probability $p_d$ to be observed,

The false signals are uniformly distributed in the detection region $\Delta$.

An additional state: $I_t = i$ if the $i$–th observation is the true signal.

State Equation (motion model):

$$d_t = d_{t-1} + v_{t-1}T + w_tT/2$$
$$v_t = v_{t-1} + w_t$$

$$P(I_t = 0) = 1 - p_d \quad \text{and} \quad P(I_t = i) = p_d/m_t, i = 1, \ldots, m_t.$$ 

Observation Equations:

$$y_{ti} = d_t + e_t, \quad \text{if} \quad I_t = i$$
$$\sim \text{Unif}[\Delta]$$
\[ T = 1, \ p_d = 0.9, \ \lambda = 0.1, \ Var(w_t) = 0.1, \ Var(e_t) = 1 \]
Example 3: Tracking a maneuvering target

A single target moving in a 2-d space with random (Gaussian) acceleration plus maneuvering

\[ x_t = H x_{t-1} + F u_t + W w_t \]
\[ y_t = G x_t + V v_t \]

where \( w_t \sim N(0, I) \) and \( v_t \sim N(0, I) \) are independent.

\( I_t \) maneuvering status:

\( I_t = 0, \) no maneuvering, \( u_t = 0 \)
\( I_t = 1, \) slow maneuvering, \( u_t \sim N(0, s_1^2 I) \)
\( I_t = 2, \) fast maneuvering, \( u_t \sim N(0, s_2^2 I) \)

With a known transition matrix \( P = P(I_{t+1} \mid I_t). \)
Example 4: Mobile network for nuclear material surveillance

- Detection of nuclear material in large cities
  - radiation dispersion devices (dirty bombs)
  - enriched uranium and weapon grade plutonium (nuclear weapons)

- Mobile sensor network
  - inexpensive sensor with GPS mounted on taxi cabs and police vehicles
  - command center receives signals and does the analysis in real time
  - lower cost and difficult to be tempered than fixed sensor network.
Source and sensor specification

- Detection distance $r$ (unknown – source energy)
- True signal $S = 1$ if $d_i < r$ when $d_i$: distance of $i$-th sensor to source location.
- Sensor error: $D = 1$ positive signal
  - Sensor False Positive rate $\gamma = P(D = 1 \mid S = 0)$
  - Sensor False Negative rate $\delta = P(D = 0 \mid S = 1)$
State Space Model

State Equation: (Moving source)

- Source location $s(t)$
- Motion model for the source location (on a city map)

Observation Equation:

- observations: $(y_i(t), D_i(t))$ (for every sensor)
- distance $d_i(t) = ||y_i(t) - s(t)||$

$$P(D_i = 1) = (1 - \delta)^{d_i(t)}\delta^{1-d_i(t)}\gamma^{1-d_i(t)}(1 - \gamma)^{d_i(t)}$$
Example 5: All Source Positioning and Navigation System

- enable[s] low cost, robust, and seamless navigation solutions
- for military users on any operational platform and in any environment,
- with or without GPS.

Objectives:

- rapid integration and reconfiguration of any combination of sensors.
- using Images, Maps, Signal databases, Location lookup tables (with landmarks, ranging signal sources, etc.)
- with platforms including Dismounts, UAVs (all sizes), Submersibles, Wheeled vehicles, Tracked Vehicles, Aircraft, Small robots
- under environments: Underwater, Underground, Jungle, Forest canopy, Suburban, Urban canyon, Building interior, Open field
Computing resources

• Portable:
  – small, light and limited battery power
  – single target

• Vehicle mounted
  – medium size, good power source.
  – a small group of targets, close to each other

• Central command
  – unlimited computing source
  – many many targets
  – new source generation
Navigational sensors:

- **GPS** – signals from multiple satellites
  - can be blocked by overcast, weak in the city
  - not available indoor, in tunnels etc
  - can be jammed

- **Wi-Fi/RF receivers**
  - measures distance to known locations
  - using signal power

- **Inertial measurement unit (IMU)**
  - a combination of accelerometers and gyroscopes
  - measures acceleration related to its own frame
  - measures rotational acceleration of the unit (frame)
  - often in combination with gravity sensors, barometer, and magnetic compass
• Range Finder
  – Measure distance to a known landmark
  – with the assistance of a map

• Millimeter-Wave Radar (based on radio-frequency technology)
  – distance and relative velocity
  – weather independent

• Star tracker, acoustic sensors, GyroCompass, inclinometer, ...
Example 2: Quickly integrate different combinations of sensors

Jamming begins

Image-based orientation to known landmark

Position Error (m)

Time (minutes)

2

Seal Delivery Vehicle (Submerged)

Vehicle (In the Clear)

Urban (Structures)

Approved for Public Release, Distribution Unlimited
State Space Models

- **State:**
  - 3-D position, velocity and acceleration
  - other sensor related states (IMU facing angles)
  - Motion states: stationary, walking, on vehicle ...
  - Environmental states: open field, highway, jungle ...

- **Observations:** sensor readings

Special features:

- **Plug-and-Play:** Sensor in-and-out, hence changing system configurations
- **Limited computational power** – approximation, sensor selection and adaptation
- **Sensor network** – a group of devices moving together.
Example 6: Digital Signal Extraction in Fading Channels

\[
\begin{align*}
& s_t \\ & \downarrow \\
& H x_t \quad \odot \quad G x_t \quad \alpha_t \quad v_t \\ & w_t \quad \downarrow \quad \odot \quad y_t
\end{align*}
\]

- **State Equations:**
  \[
  \begin{cases}
  x_t = H x_{t-1} + w_t \\
  \alpha_t = G x_t \\
  s_t \sim p(\cdot | s_{t-1})
  \end{cases}
  \]

- **Observation equation:**
  \[ y_t = \alpha_t s_t + v_t \]

- \( \alpha_t = G x_t \): Butterworth filter of order \( r = 3 \) i.e. ARMA(3,3)
  
  Cutoff frequency 0.1

- **Noise:**
  1. \( v_t \sim N(0, \sigma^2) \)
  2. \( v_t \sim (1 - \alpha)N(0, \sigma_1^2) + \alpha N(0, \sigma_2^2) \)
Phase Ambiguity: \[ p(\alpha_t, s_t \mid y_t) = p(-\alpha_t, -s_t \mid y_t) \]

Differential coding:
Information sequence: \( s_1, \ldots, s_t \).
Transmitted sequence: \( s_1^*, \ldots, s_t^* \), s.t \( s_{t-1}^* s_t^* = s_t, s_1^* = s_1 \).

Differential detector:
\[
\hat{s}_t = \text{sign}(y_t y_{t-1}) = \text{sign}(\alpha_t \alpha_{t-1} s_t + \alpha_t s_t^* e_{t-1} + \alpha_{t-1} s_{t-1}^* e_t + e_{t-1} e_t)
\]

Assumption: \( \alpha_t \) changing slowly.

Error floor: the frequency that \( \alpha_t \) changes the sign.
Example 7: Stochastic Volatility Models:

$Y_t$ stock returns (zero mean). $\alpha_t$ volatility

State Equation: $\alpha_t = c + \phi \alpha_{t-1} + \eta_t$

Observation Equation: $Y_t \sim N(0, \exp(\alpha_t))$

where $\eta_t \sim N(0, \sigma^2)$, and $c > 0$.

Or

State Equation: $\alpha_t = c + \phi \alpha_{t-1} + \eta_t$

Observation Equation: $\log(Y_t^2) = \alpha_t + v_t$

where $v_t = \log(e_t^2)$, $e_t \sim N(0, 1)$. 
Example 8: Dynamic Factor Models:

- $y_{1t}, \ldots, y_{nt}$ time series ($n$ large)

- Factor models ($k$ small): [Observation Equations]

$$y_{it} = \beta_{0i} + \beta_{1i}f_{1t} + \ldots + \beta_{ki}f_{kt} + \varepsilon_t \quad i = 1, \ldots, n$$

- The factors $f_{1t}, \ldots, f_{kt}$ are unobserved states with a dynamic structure.

- State Equations: $(f_{1t}, \ldots, f_{kt})$ follows a Vector ARMA model
Example 9: Non-Gaussian TS – Generalized ARMA

- Binomial Counts
  - Observation equation: \( y_t \sim Binomial(N_t, p_t) \)
  - State equation: \( x_t = \log(p_t) - \log(1 - p_t) \sim \) an ARMA model

- Poisson Counts
  - Observation equation: \( y_t \sim Poisson(\lambda_t) \)
  - State equation: \( x_t = \log(\lambda_t) \) follows an ARMA model

- Gamma time series
  - Observation equation: \( y_t \sim Gamma(c\mu_t^d, c\mu_t^{d-1}) \)
  - State equation: \( x_t = \log(\mu_t) \) follows an ARMA model

- Generalized ARMA-Stochastic Volatility model
  - Observation equation: \( y_t \sim p(\mu_t, \sigma_t, \psi) \)
  - \((\mu_t, \sigma_t)\) follows a stochastic volatility model
Example 10: Market sentiment with text data

- $y_{t1}, \ldots, y_{tdt}$: text data or a summary of text data (e.g., Binomial counts of certain key words)
- $x_t$ market sentiment (whatever it is defined)
- $y_{ti} \sim p(x_t, \theta)$ e.g. $(y_{t1}, \ldots, y_{tdt}) 	ext{Multinomial}(N_t, \eta_t(x_t))$
- $x_t \sim \text{ARMA model}$

In a way, it is a dynamic factor model, with text data.
Example 11: Yield curve (Interest rate) over time
Curve time series $X_t(s)$ driven by dynamic processes

- Any fixed $t$, $X_t(s) = f_t(s; \theta_t) + \varepsilon_t(s)$, and $s \in \Omega$
- The function $f_t(\cdot)$ is known, except (parameter) $\theta_t$.
- $\varepsilon_t(s)$: a white noise process defined on $\Omega$ with $E(\varepsilon_t(s)) = 0$.
- $\theta_t$: a random (driving) process over $t$.
- The dependency between $X_t(s)$ is completely characterized by the parameter process $\theta_t$ and the noise process $\varepsilon_t$.
- We call $\{\theta_t\}$ the driving process.
- In most applications, $X_t(\cdot)$ is only observed at a finite number of locations $\{X_t(s_{ti}), i = 1, \ldots, m_t\}$. 
Finite dimensional driving processes

\( \theta_t \) follows a parametric ARMA process:

\[
X_t(s_{ti}) = f(s_{ti}, \theta_t) + \varepsilon_t(s_{ti}), \quad i = 1, \ldots, m_t,
\]

\[
\theta_t = g(\theta_{t-1}, \ldots, \theta_{t-p}, e_t, \ldots, e_{t-q}, \gamma),
\]

\[
\begin{array}{cccc}
\cdots & \varepsilon_t(s) & \varepsilon_{t+1}(s) & \cdots \\
\downarrow & \downarrow \\
\cdots & X_t(s) & X_{t+1}(s) & \cdots \\
\uparrow & \uparrow \\
\cdots & \rightarrow & \theta_t & \rightarrow & \theta_{t+1} & \rightarrow & \cdots \\
\end{array}
\]

- A generalized state space model

- \( g(\cdot) \) is a known function with unknown parameters \( \gamma \) and \( e_t \) is a sequence of scalar or vector white noises.
Probabilistic Dynamic Systems

Definition: A probabilistic dynamic system is abstracted as a sequence of evolving probability distributions $\pi_t(x_t)$.

$x_t$: state variable:

(i) increasing dimension: $x_{t+1} = (x_t, x_{t+1})$

(ii) discharging: $x_t = (x_{t+1}, d_t)$

(iii) no change: $x_{t+1} = x_t$

State space model is a special case of probabilistic dynamic system.

$$\pi_t(x_t) = p(x_1, \ldots, x_t \mid y_1, \ldots, y_t)$$
Example 12: Sequential Bayesian Inference:

\[ x_t = \theta \]

and

\[ \pi_t(x_t) = p(\theta \mid y_1, \ldots, y_t) \]
Example 13: Bayesian Missing Data

- $z_1, \ldots, z_n$ iid from $p(z, \theta)$.
- $z_i = (y_i, x_i)$: $y_i$ observed $x_i$ missing

Let $x_t = (x_0, x_1, \ldots, x_t)$. $x_0 = \theta$. $y_t = (y_1, \ldots, y_t)$.

Then the dynamic system is $\pi_t(x_t) = p(x_t \mid y_t)$

Note: for fixed $\theta$, the likelihood can be evaluated
The Growth Principle

*Decompose a complex problem into a sequence of simpler problems,*
— forming a dynamic system from a fixed dimensional problem

- Target distribution $\pi(\mathbf{x})$ where $\mathbf{x} = (x_1, \ldots, x_N)$
- Let $\mathbf{x}_t = (x_1, \ldots, x_t) = (\mathbf{x}_{t-1}, x_t)$
- Define a sequence of intermediate distributions $\pi_t(\mathbf{x}_t)$.
- Moving from $\pi_{t-1}(\mathbf{x}_{t-1})$ to $\pi_t(\mathbf{x}_t)$ is simple.
- Moving from $\pi_{t-1}(\mathbf{x}_{t-1})$ to $\pi_t(\mathbf{x}_t)$ is smooth.

$$\int \pi_t(\mathbf{x}_{t-1}, x_t) dx_t \approx \pi_{t-1}(\mathbf{x}_{t-1})$$

- $\pi_N(\mathbf{x}_N) = \pi(\mathbf{x})$
Example 14: Self-avoiding walks (SAW) and Self-avoiding loops (SAL)

- How many are there?
- What is the average size of enclosed void of SAL(n)?
- What is the average number of contacts?
Protein Structures
RNA local structure

\[ \frac{\Delta S}{k_B} \]

\[ \text{Loop length} \]
Enumeration:

<table>
<thead>
<tr>
<th>$n$</th>
<th>SAWs</th>
<th>SALs</th>
<th>void size</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>110,188</td>
<td>4,116</td>
<td>1.40</td>
</tr>
<tr>
<td>16</td>
<td>802,075</td>
<td>23,504</td>
<td>2.06</td>
</tr>
<tr>
<td>18</td>
<td>5,808,335</td>
<td>137,412</td>
<td>2.82</td>
</tr>
<tr>
<td>20</td>
<td>41,889,578</td>
<td>818,210</td>
<td>3.68</td>
</tr>
<tr>
<td>22</td>
<td>301,100,754</td>
<td>4,945,292</td>
<td>4.63</td>
</tr>
<tr>
<td>24</td>
<td>2,158,326,727</td>
<td>30,255,240</td>
<td>5.68</td>
</tr>
</tbody>
</table>
SALs:

- Starting at \((0, 0)\) and \((1, 0)\) and ending at \((0, 0)\)
- Target distribution: uniform of all SALs of length \(n\).
- Partial chain \((x_1, \ldots, x_t)\) is a SAW.
- Strong constraint at the end – shrinking support

Hence

- Intermediate distributions
  \(\pi_t(x_t)\): uniform of all SAWs of length \(t\) such that

\[
d(x_t) < n - t \quad \text{(support)}
\]

where \(d(x_t) = |x_{t,1}| + |x_{t,2}|\)
Example 15: Diffusion Bridges

(Multivariate) Diffusion process $v_t$ satisfying SDE:

$$dv_t = b(v_t; \theta)dt + \sigma(v_t; \theta)dB_t$$

where $B_t$: Brownian Motion. $\theta$: unknown parameters

Observations: $v_i = v_{t_i}$, $i = 0, \ldots, n$, observed at discrete time points $t_0, \ldots, t_n$.

Aims:

- Estimate the log-likelihood function (conditioned on $v_0$)

$$L(\theta) = \log p(v_1, \ldots, v_n \mid v_0, \theta) = \sum_{i=1}^{n} \log p(v_i \mid v_{i-1}, \theta)$$

- Obtain the MLE of $\theta$

- Estimate $E(h(v[t_0, t_n]))$ (e.g. quantile of the path $v([t_0, t_n])$)
Consider the transition density \( p(v_1 \mid v_0, \theta) \).

- Except for a few cases, no analytical form for \( p(v_1 \mid v_0, \theta) \)
- If \( v[t_0, t_1] \) is known or observed completely, then \( p(v[t_0, t_1] \mid \theta) \) can be found since

\[
v(t_1) - v(t_0) = \int_{t_0}^{t_1} b(v(t), \theta) dt + \int_{t_0}^{t_1} \sigma(v(t), \theta) dB_t.
\]

The integration can be obtained analytically or numerically.

- But \( v(t_0, t_1) \) is missing.

- Monte Carlo: simulate the diffusion bridge \( v(t_0, t_1) \) following \( p(v(t_0, t_1) \mid v_0, v_1, \theta) \)
- Consider one segment $[V_0, V_1]$ (due to Markovian property).
- **Euler-Maruyama approximation** (Gaussian system)
  
  $V_t \approx V_{t-\Delta t} + b(V_{t-\Delta t}; \theta)\Delta t + \sigma(V_{t-\Delta t}; \theta)(W_t - W_{t-\Delta t})$

- Divide $[0, 1]$ into small intervals $0 < s_1 < \ldots < s_m < 1$.
- $V_{s_1}, \ldots, V_{s_m}$ missing.
- Target distribution $\pi(V_0, V_{s_1}, \ldots, V_{s_m}, V_1 | \theta)$ can be easily evaluated.

- The intermediate distributions: $\pi_t(V_0, V_{s_1}, \ldots, V_{s_t}, V_1 | \theta)$ defined by the same linearized system, with special treatment of the last *large* step from $V_{s_t}$ to $V_1$. 
**Fixed dimension with augmentation**


- Target distribution \( \pi(x) \), for \( x \in \Omega \)
- Let \( x_t = (x_1, \ldots, x_t) \in \Omega^t \) where \( x_i \in \Omega \).
- Construct a sequence of intermediate distributions \( \pi_t(x_t) \).
- The marginal distribution of \( \pi_n(x_n) = \pi(x_n) \).

Note:

- This is very similar to MCMC.
- Samples are *moved* within the same space with the trial dis-
tribution \( g_t(x_t | x_{t-1}) \).
- Eventually the final marginal dist. is the target dist.
- Finite steps of movements, but with weights
For example:

- Design a sequence of *marginal intermediate distributions* $\pi_t(x_t)$, with $\pi_n(x_n) = \pi(x)$.

For example:

- $\pi_t(x) \propto \pi(x)^{\phi_t} \mu_1(x)^{1-\phi_t}$ with $0 \leq \phi_1 < \ldots < \phi_n = 1$.
- $\pi_t(x) \propto \pi(x \mid y_1, \ldots, y_t)$: sequential new observations
- $\pi_t(x) \propto \pi(x)^{\phi_t}$ with $\phi_t \to \infty$ (simulated annealing)

- Let

$$\pi_t(x_1, \ldots, x_t) = \pi_t(x_t) \prod_{k=1}^{t-1} L_k(x_{k+1}, x_k)$$

where $L_k(x, y)$ is a Markov kernel with $\pi_k$ as the invariant distribution.
Other Applications:

- Target recognition (Srivastava et al 2001).
- Blind equalization (Liu and Chen 1995)
- Speech recognition (Rabiner 1989)
- Mobile robot localization (Dellaert et al 1999, Fox et al 1999, 2001)
- Freeway traffic vision (for vehicle control) (Huang et al 1994)
- DNA sequence analysis (Churchill 1989)
- Switching (auto)regression models (Kaufmann, 2002)
- Dynamic Bayesian networks (Koller and Lerner, 2001, Murphy and Russell, 2001)
- On-line control of industrial production (Marrs, 2001)
- Combinatorial optimizations (Wong and Liang 1997)
- Wireless communications (Chen et al 2000, Wang et al 2000)
• Signal processing (Djuric, 2001, Wang et al 2002)
• Audio signal enhancing (Fong et al, 2002)
• Data network analysis (Coates and Nowak, 2002)
• Counting 0-1 tables (Liu 2001)
• Neural networks (Andriew et al 1999, de Freitas et al. 2000)