2022 Fall
IE 313 Time Series Analysis

A2. State Space Models

Yongjae Lee
Department of Industrial Engineering
ULSAN NATIONAL INSTITUTE OF SCIENCE AND TECHNOLOGY

State space models

- State-space model or dynamic linear model
  - First introduced in Kalman (1960) and Kalman and Bucy (1961)
  - The model arose in the spacecraft tracking setting, where the state equation defines the motion equations for
    - Position or state of a spacecraft with location x_t
    - Data y_t reflect information that can be observed from a tracking device such as velocity and azimuth
  - The model has been applied to modeling data from various fields including economics, medicine, and soil sciences.

State space models

- In general, the state space model is characterized by two principles
  - **State process** x_t
    - A hidden or latent process
    - It is assumed to be a Markov process
      - This means that the future $\{x_s \mid s > t\}$ and past $\{x_s \mid s < t\}$ are independent conditional on the present x_t
  - **Observations** y_t
    - These are independent given the states x_t
      - This means that the dependence among the observations is generated by states

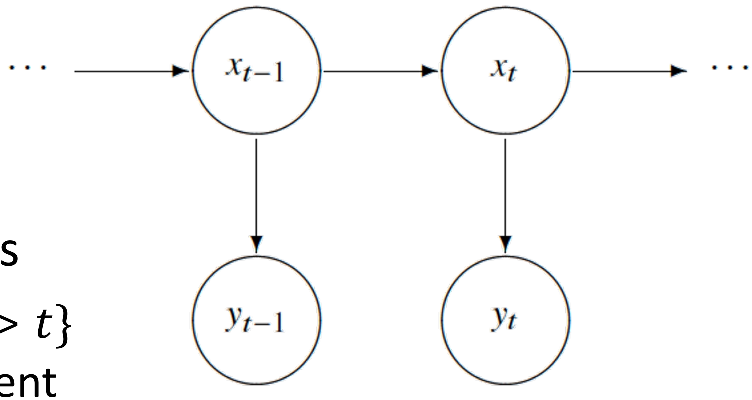
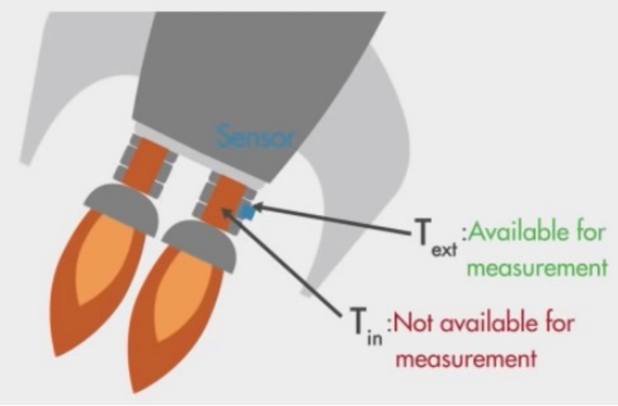
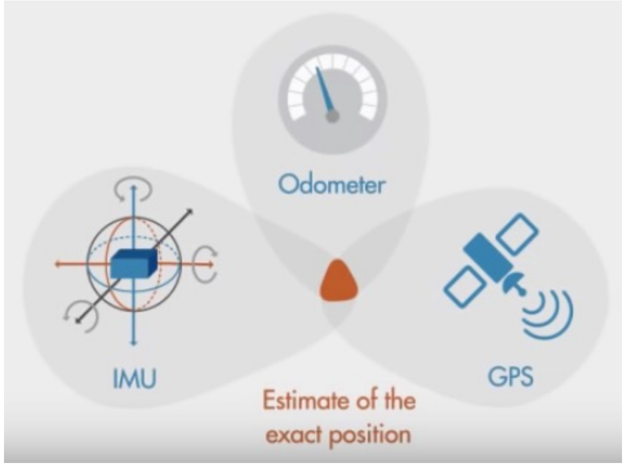


Fig. 6.1. *Diagram of a state space model.*

State space models

- State space models are largely used when
  - The variables of interest can only be measured indirectly
    - e.g., financial market state (bull or bear), temperature of combustion chamber in spacecrafts
  - Measurements are available from various sensors but might be subject to noise
    - e.g. estimate the location of a car based on various sensors





Section 6.1

Linear Gaussian Model

Dynamic linear model (DLM)

- The **linear Gaussian state space model** or **dynamic linear model (DLM)**, in its basic form, employs an order one, p-dimensional vector autoregression as the **state equation**,

$$\overset{\text{state vector}}{x_t} = \Phi x_{t-1} + \overset{\text{noise}}{w_t}$$

  – w_t are $p \times 1$ iid zero-mean normal vectors with covariance matrix Q, i.e., $w_t \sim \text{iid } \mathrm{N}_p(0, Q)$

- In DLM, we assume the process starts with a normal vector x_0, such that $x_t \sim \mathrm{N}_p(\mu_0, \Sigma_0)$

Dynamic linear model (DLM)

- In DLM, we do not observe the state vector x_t directly, but only **a linear transformed version of it with noise added**, say

observation data vector — state vector — observation noise

$$y_t = A_t x_t + v_t$$

  - where A_t is a $q \times p$ measurement or observation matrix
  - $v_t \sim \text{iid } N_q(0, R)$ is additive observation noise
  - The above equation is called the **observation equation**
    - Observed data vector y_t is q-dimensional, which can be larger than or smaller than p, the state dimension
  - In addition, we initially assume, for simplicity, x_0, $\{w_t\}$, and $\{v_t\}$ are uncorrelated

Dynamic linear model (DLM)

- We can also incorporate **exogenous variables**, or **fixed inputs**, into the states or into the observations
  - Suppose that we have an $r \times 1$ vector of inputs, and write the model as

$$x_t = \Phi x_{t-1} + \Upsilon u_t + w_t$$
$$y_t = A_t x_t + \Gamma u_t + v_t$$

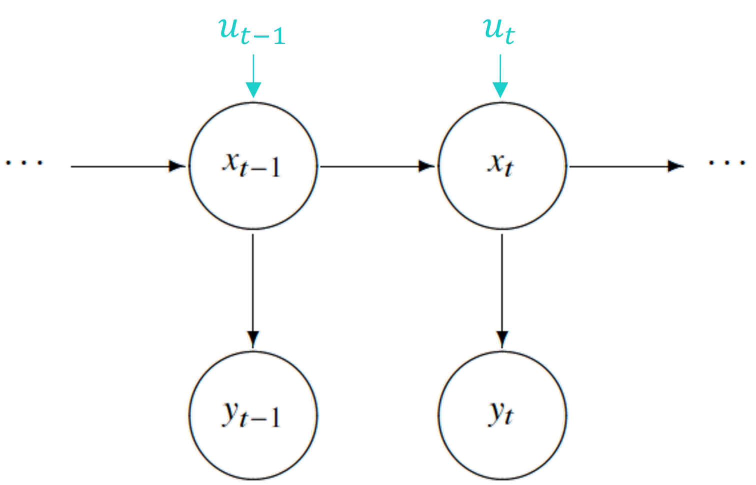


Fig. 6.1. Diagram of a state space model.

Dynamic linear model (DLM)

Bone marrow: 척수
White blood count: 백혈구 수치
Platelet: 혈소판
Hematocrit: 혈구용적율

- **Example 6.1**: A biomedical example
  - Suppose we consider the problem of monitoring the level of several biomedical markers after a cancer patient undergoes a bone marrow transplant

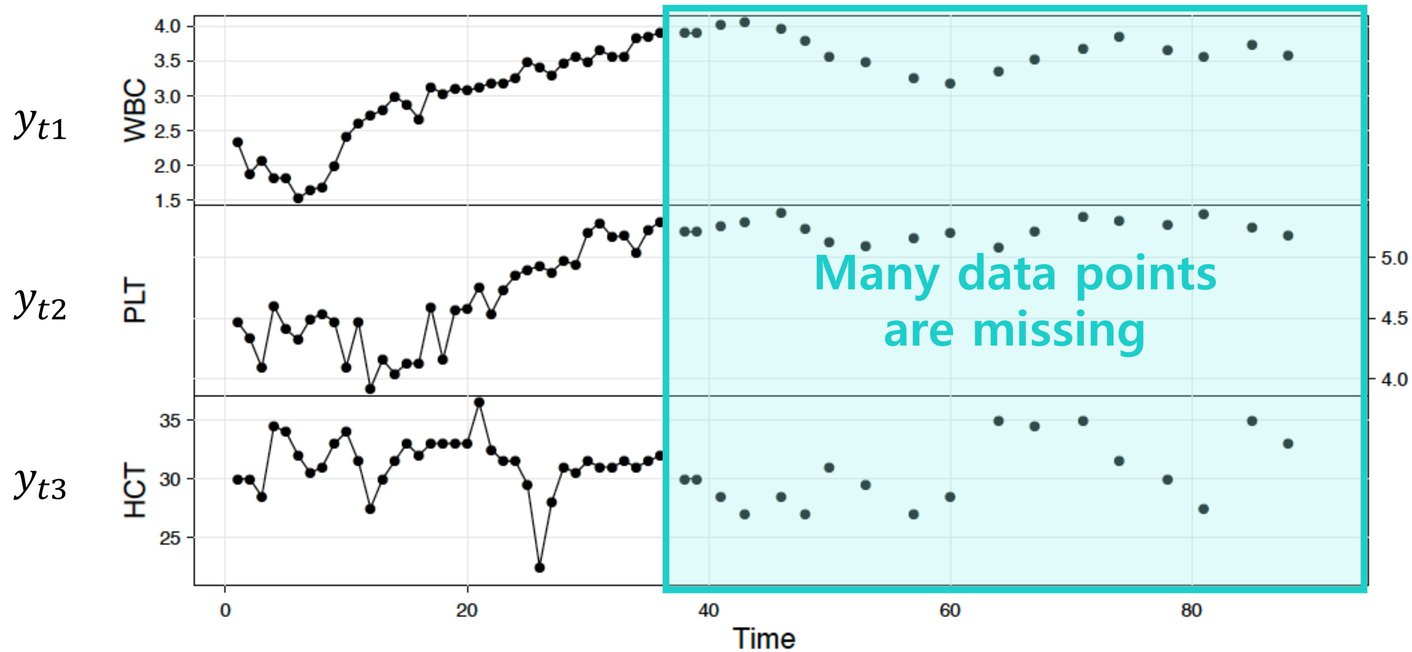


Fig. 6.2. Longitudinal series of monitored blood parameters, log (white blood count) [WBC], log (platelet) [PLT], and hematocrit [HCT], after a bone marrow transplant (n = 91 days).

Dynamic linear model (DLM)

- **Example 6.1**: A biomedical example (cont'd)
  - The main objectives are to model the three variables using the state-space approach, and to estimate the missing values
  - We model the three variables in terms of the state equation

$$\begin{pmatrix} x_{t1} \\ x_{t2} \\ x_{t3} \end{pmatrix} = \begin{pmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ \phi_{21} & \phi_{22} & \phi_{23} \\ \phi_{31} & \phi_{32} & \phi_{33} \end{pmatrix} \begin{pmatrix} x_{t-1,1} \\ x_{t-1,2} \\ x_{t-1,3} \end{pmatrix} + \begin{pmatrix} w_{t1} \\ w_{t2} \\ w_{t3} \end{pmatrix}.$$

  - The observation equations would be

$$y_t = A_t x_t + v_t$$

    - $A_t = \begin{cases} I & \text{when a blood sample was taken on that day} \\ 0 & \text{otherwise} \end{cases}$

Dynamic linear model (DLM)

- **Example 6.2**: Global warming

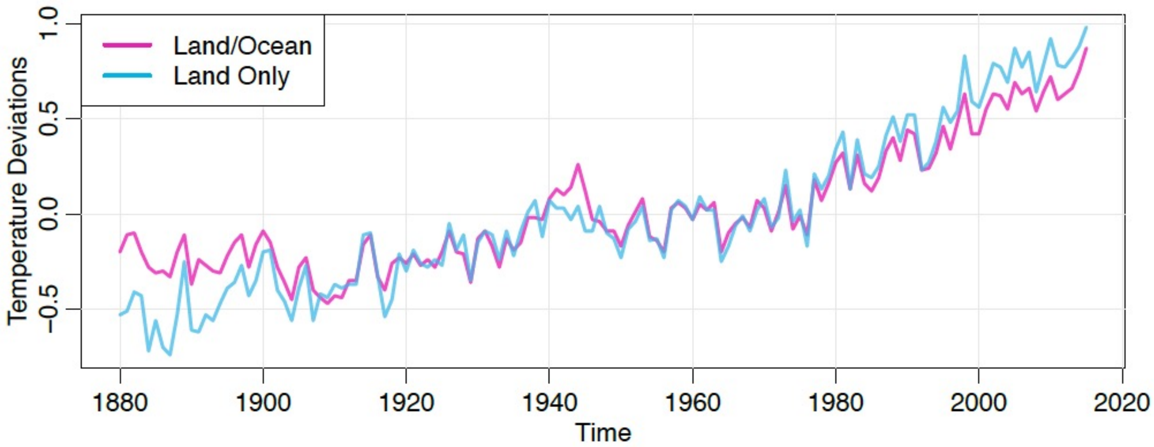


Fig. 6.3. *Annual global temperature deviation series, measured in degrees centigrade, 1880–2015. The series differ by whether or not ocean data is included.*

– The figure shows two different estimator for the global temperature series from 1880 to 2015
  - First series are the global mean land-ocean temperature index data
  - Second series are the surface air temperature index data

Dynamic linear model (DLM)

- **Example 6.2**: Global warming (cont'd)
  - Conceptually, both series should be measuring **the same underlying climatic signal**
  - We consider the problem of extracting this underlying signal
    - Suppose both series are observing the same signal with different noise; that is

$$y_{t1} = x_t + v_{t1} \quad \text{and} \quad y_{t2} = x_t + v_{t2}$$

or more compactly as

$$\begin{pmatrix} y_{t1} \\ y_{t2} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} x_t + \begin{pmatrix} v_{t1} \\ v_{t2} \end{pmatrix}$$

› where $R = \text{var}\begin{pmatrix} v_{t1} \\ v_{t2} \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{pmatrix}$

Dynamic linear model (DLM)

- **Example 6.2**: Global warming (cont'd)

  – It is reasonable to suppose that the unknown common signal x_t can be modeled as a random walk with drift of the form

$$x_t = \delta + x_{t-1} + w_t$$

  - with $Q = \mathrm{var}(w_t)$

Dynamic linear model (DLM)

- **Example 6.2**: Global warming (cont'd)

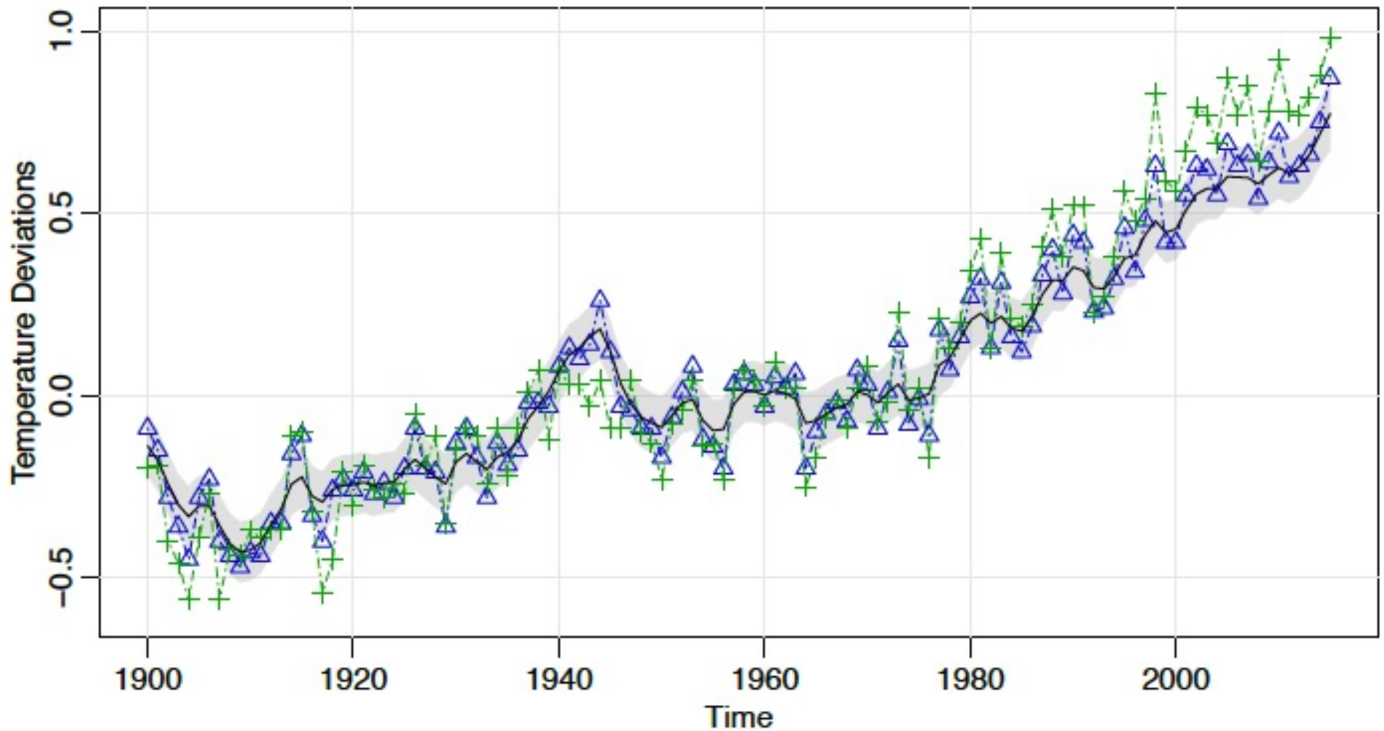


Fig. 6.5. Plot for Example 6.7. The dashed lines with points (+ and △) are the two average global temperature deviations shown in Figure 6.3. The solid line is the estimated smoother $\hat{x}_t^n$, and the corresponding two root mean square error bound is the gray swatch. Only the values later than 1900 are shown.

Dynamic linear model (DLM)

- In the previous examples, we had well-defined differential equations describing the state transition

- In general, we need to estimate many parameters to
  - Define the particular model
  - Estimate or forecast values of the underlying unobserved process x_t

- Still, the advantages of the state-space formulation are
  - The ease with which we can treat various missing data configurations
  - The incredible array of models that can be generated from (6.3) and (6.4)
    - Note that the observation matrix A_t is analogous to the design matrix in the usual regression and ANOVA setting

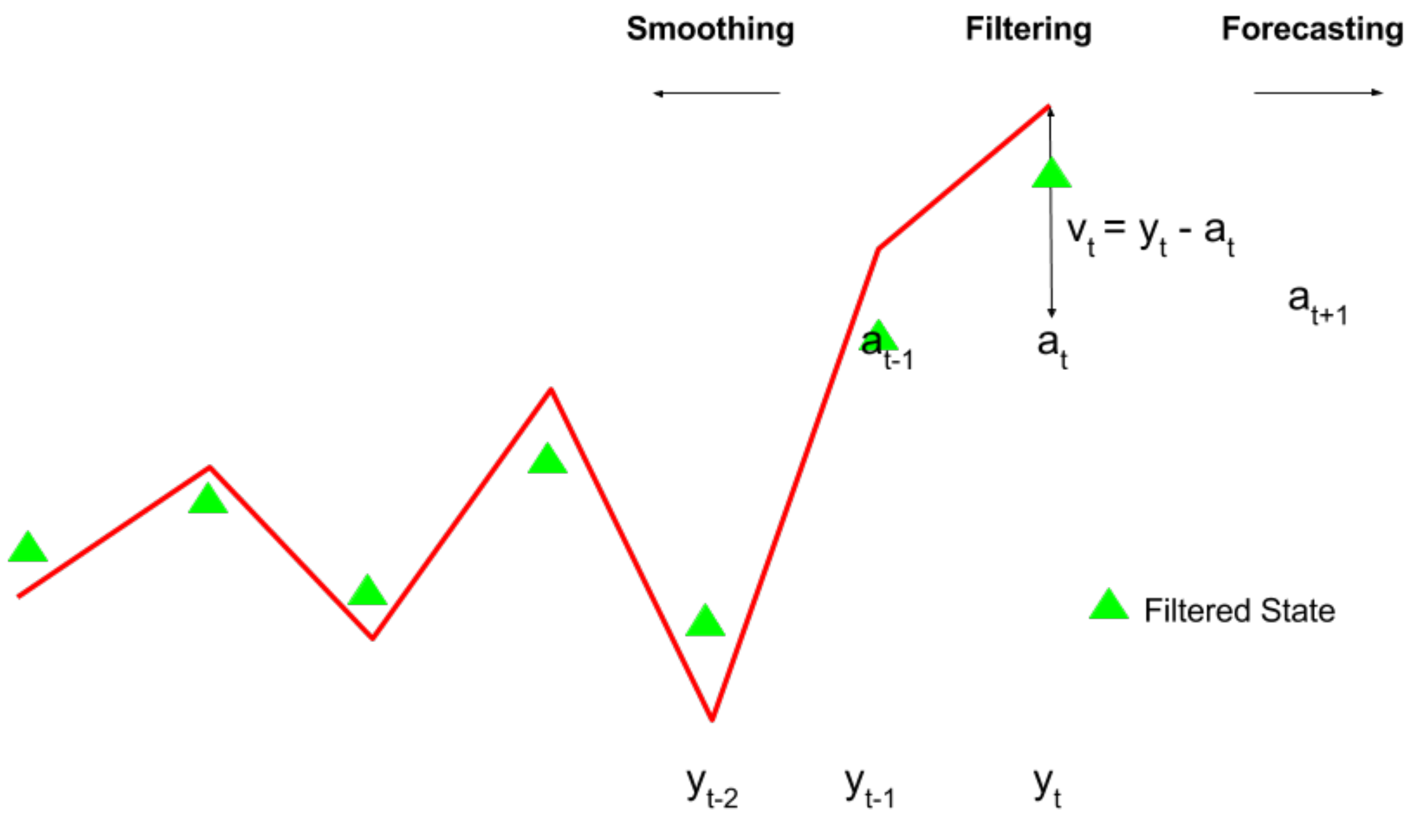
Section 6.3

Filtering, Smoothing, and Forecasting

Filtering, smoothing, and forecasting

- From a practical view, a primary aim of any analysis involving the state space model would be to produce **estimators for the underlying unobserved signal x_t** given the data $y_{1:s}$ to time s
  - When $s < t$, the problem is called **forecasting** or **prediction**
  - When $s = t$, the problem is called **filtering**
  - When $s > t$, the problem is called **smoothing**
- In addition to these estimates, we would also want to measure their **precision**
  - The solution to these problems is accomplished via the **Kalman filter and smoother**

Filtering, smoothing, and forecasting



Filtering, smoothing, and forecasting

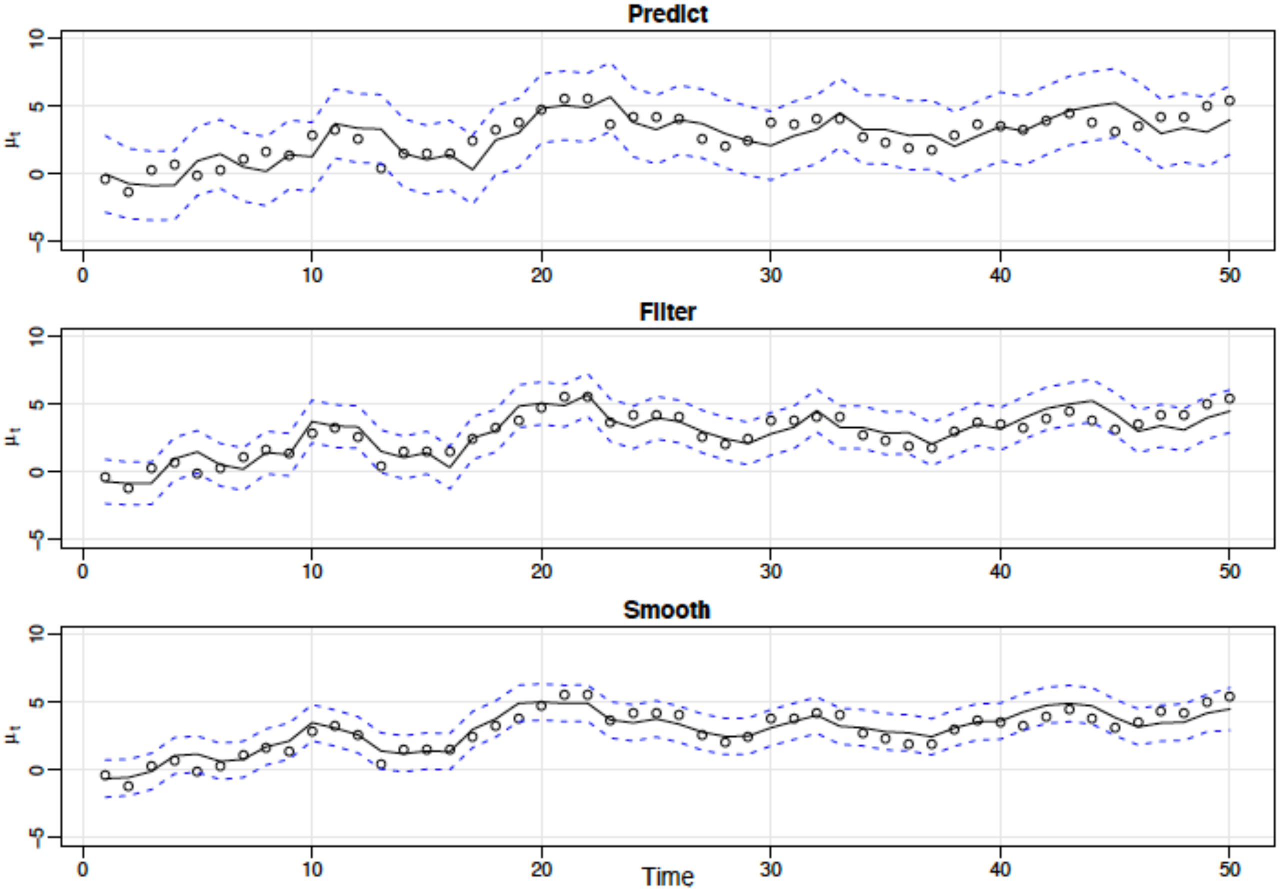


Fig. 6.4. Displays for Example 6.5. The simulated values of μ_t, for $t = 1, \dots, 50$, given by (6.51) are shown as points. The top shows the predictions μ_t^{t-1} as a line with $\pm 2\sqrt{P_t^{t-1}}$ error bounds as dashed lines. The middle is similar, showing $\mu_t^t \pm 2\sqrt{P_t^t}$. The bottom shows $\mu_t^n \pm 2\sqrt{P_t^n}$.

Kalman filter

- Notations

  - $x_t^s = \mathrm{E}(x_t \mid y_{1:s})$

  - $P_{t1,t2}^s = \mathrm{E}[(x_{t1} - x_{t1}^s)(x_{t2} - x_{t2}^s)']$
    - When $t1 = t2$ ($= t$ say), write P_t^s for convenience

Kalman filter

- **Property 6.1**: The Kalman filter
  - For the state-space model specified as

$$x_t = \Phi x_{t-1} + \Upsilon u_t + w_t$$
$$y_t = A_t x_t + \Gamma u_t + v_t$$

$w_t \sim \text{iid } \mathrm{N}_p(0, Q)$
$v_t \sim \text{iid } N_q(0, R)$

with initial conditions $x_0^0 = \mu_0$ and $P_0^0 = \Sigma_0$,

- for $t = 1, \dots, n,$

prediction　prior knowledge

$$x_t^{t-1} = \Phi x_{t-1}^{t-1} + \Upsilon u_t,$$
$$P_t^{t-1} = \Phi P_{t-1}^{t-1} \Phi' + Q,$$

- with

posteriori estimate　prediction　update　new observation (or measurement)

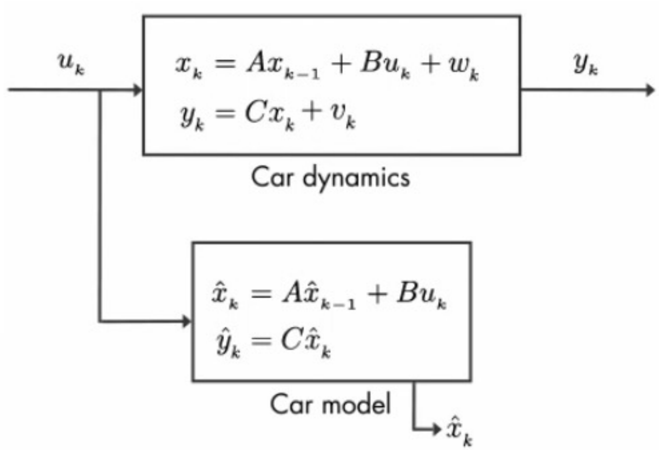
$$x_t^t = x_t^{t-1} + K_t(y_t - A_t x_t^{t-1} - \Gamma u_t),$$
$$P_t^t = [I - K_t A_t] P_t^{t-1}$$

- where $K_t = P_t^{t-1} A_t' [A_t P_t^{t-1} A_t' + R]^{-1}$ is called the Kalman **gain**

- Prediction for $t > n$ is accomplished by the above procedure with initial conditions x_n^n and P_n^n

Kalman filter

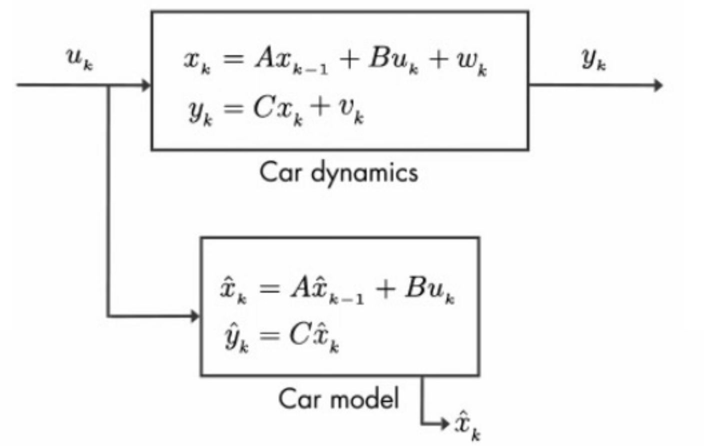
- Example: estimating a car's position

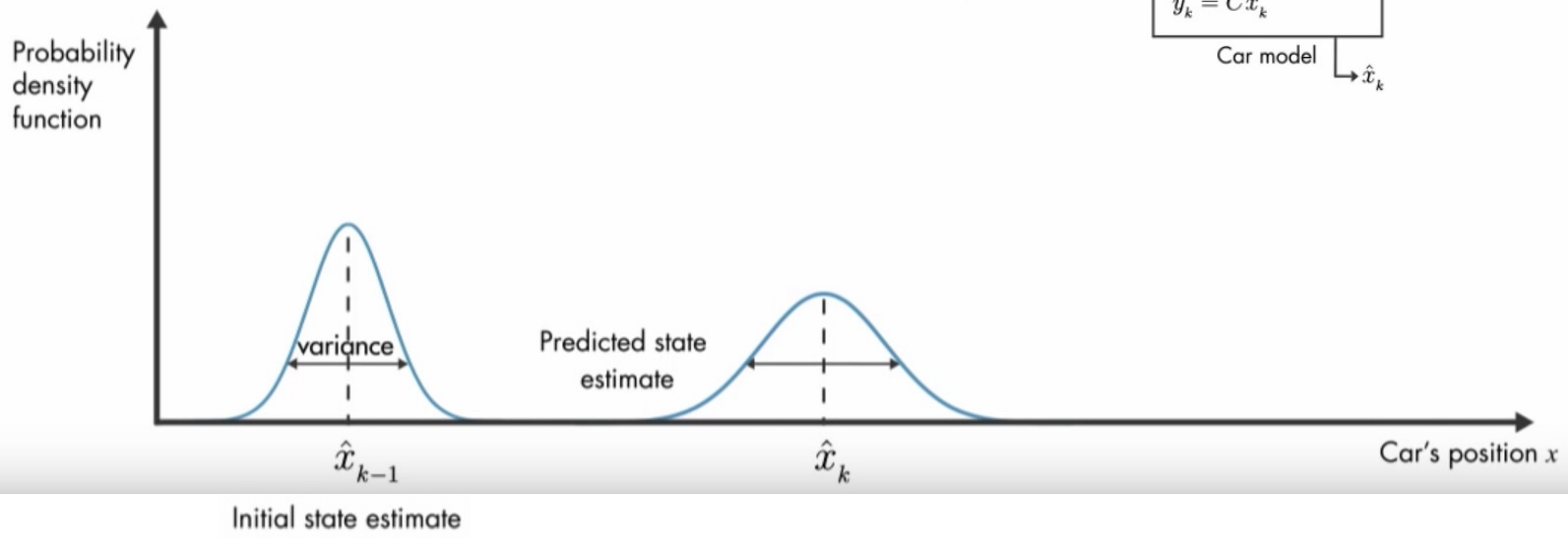


Initial state estimate

Kalman filter

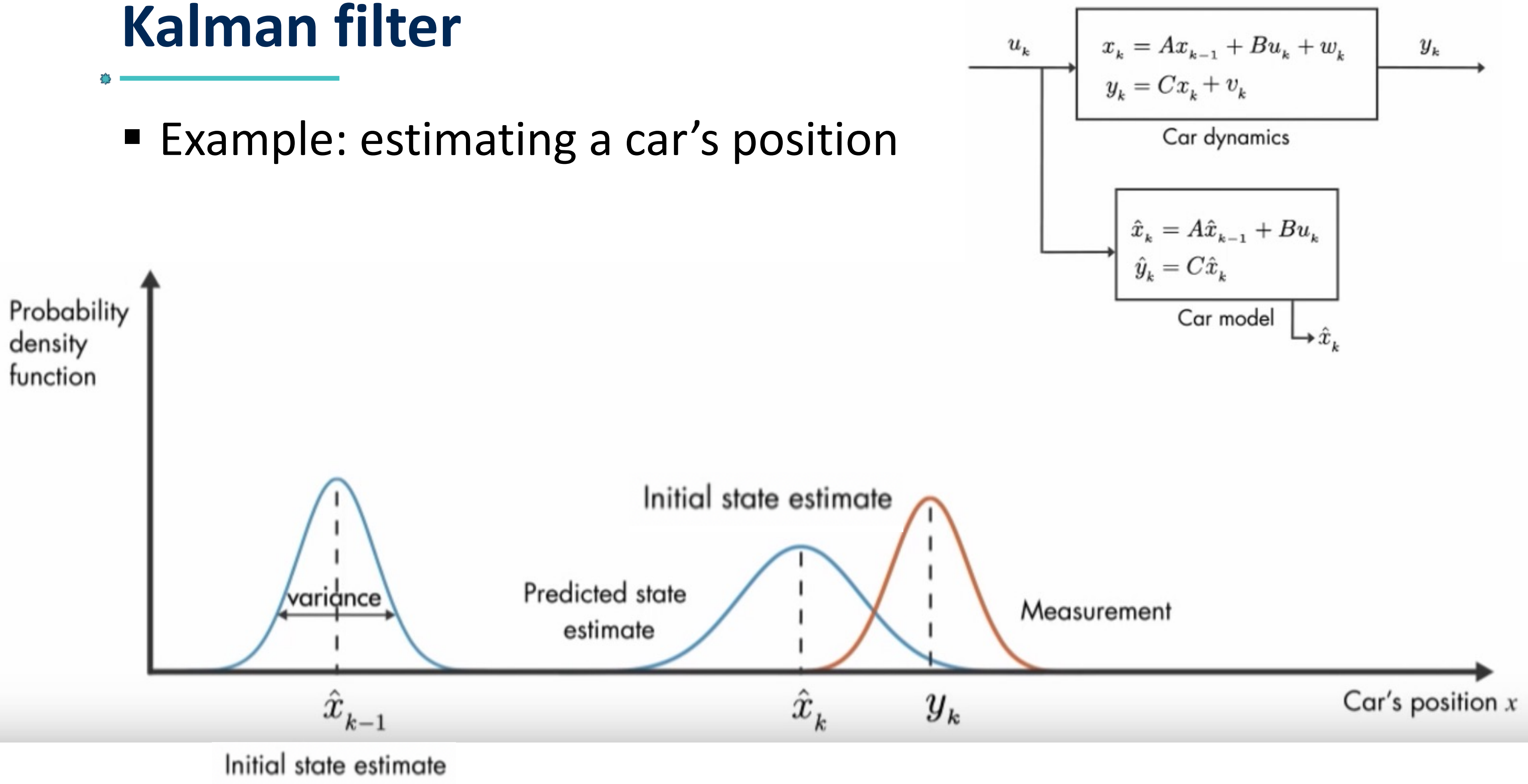
- Example: estimating a car's position





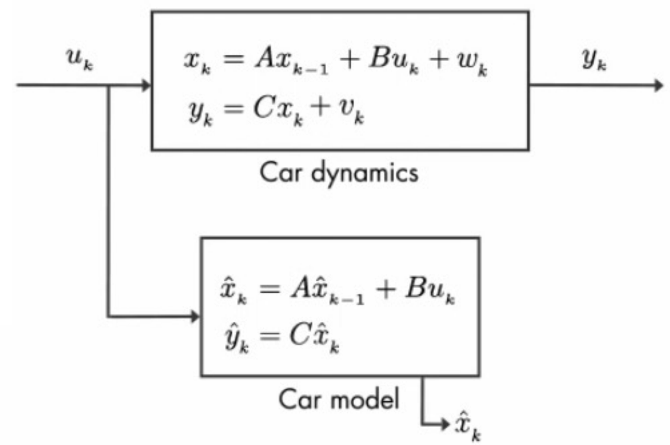
Kalman filter

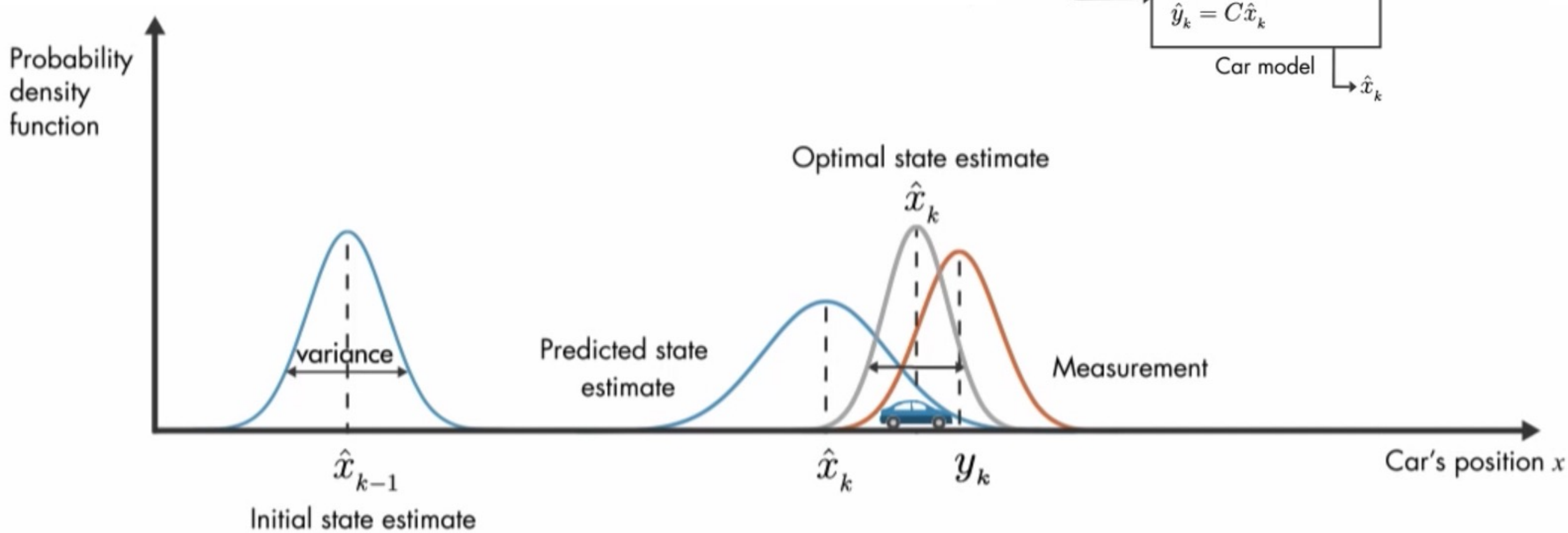
- Example: estimating a car's position



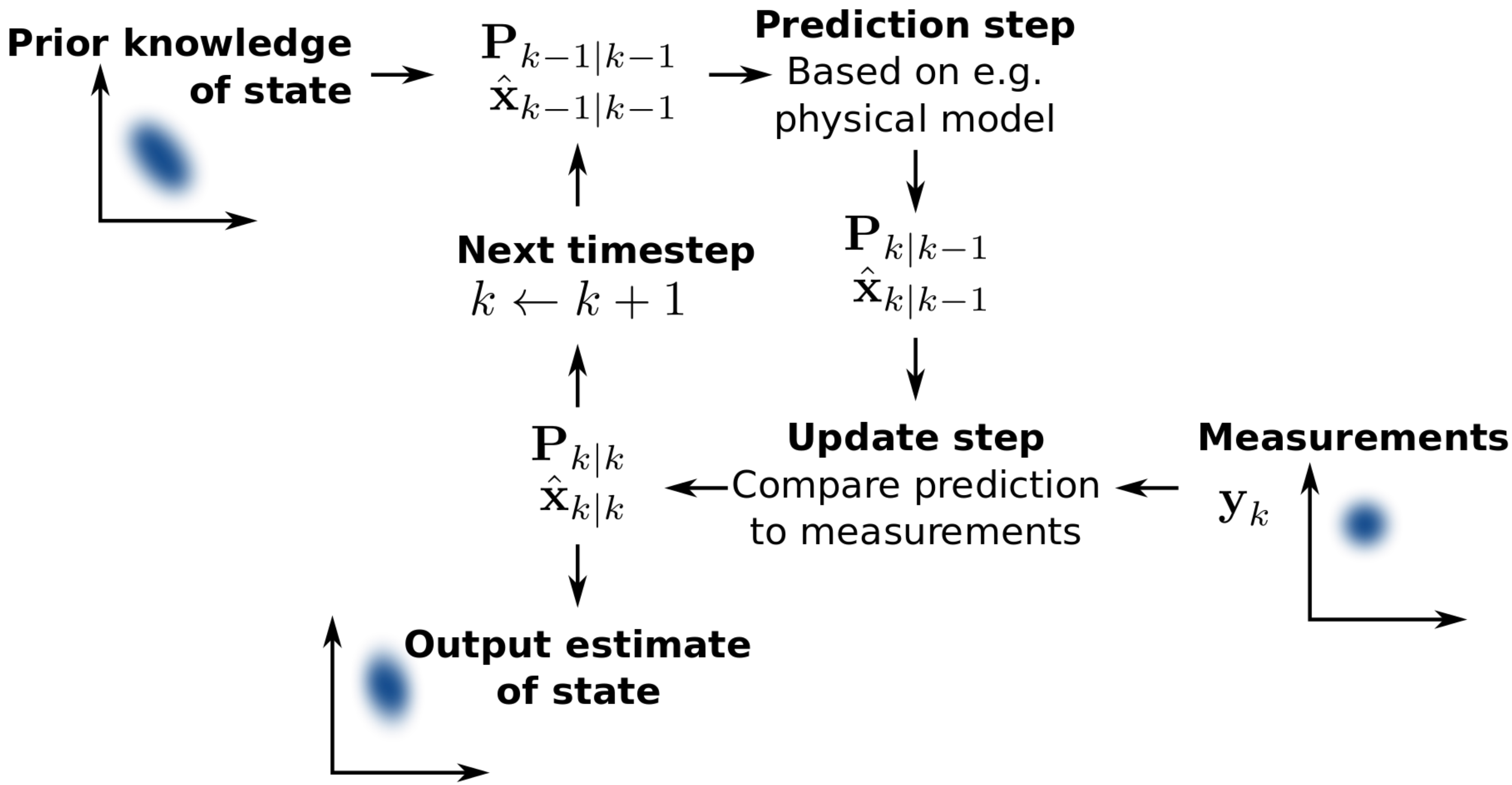
Kalman filter

- Example: estimating a car's position





Kalman filter



Kalman filter

- Kalman gain has the form

$$K_t = \frac{P_t^{t-1} A_t'}{A_t P_t^{t-1} A_t' + R}$$

  - It determines the balance between the prior prediction and measurement (observation)

posteriori estimate | prediction | new observation (or measurement)

$$x_t^t = x_t^{t-1} + K_t(y_t - A_t x_t^{t-1} - \Gamma u_t)$$

    - If R (covariance of measurement y_t) is close to 0,
      - K_t becomes close to $1/A_t$ and then x_t^{t-1} will be almost cancelled out
      - And thus, x_t^t will be much closer to y_t
    - If P_t^{t-1} (error covariance of prior x_t^{t-1}) is close to 0,
      - K_t becomes close to 0
      - And thus, x_t^t will be much closer to x_t^{t-1}

Kalman filter (time-varying case)

- **Corollary 6.1**: Kalman filter (time-varying case)
  - If any or all of the parameters of the state-space model are time dependent,
    - $\Phi = \Phi_t, \Upsilon = \Upsilon_t, Q = Q_t$ in the state equation
    - $\Gamma = \Gamma_t, R = R_t$ in the observation equation,
  - Or the dimension of the observational equation is time dependent,
  - Property 6.1 holds with the appropriate substitutions

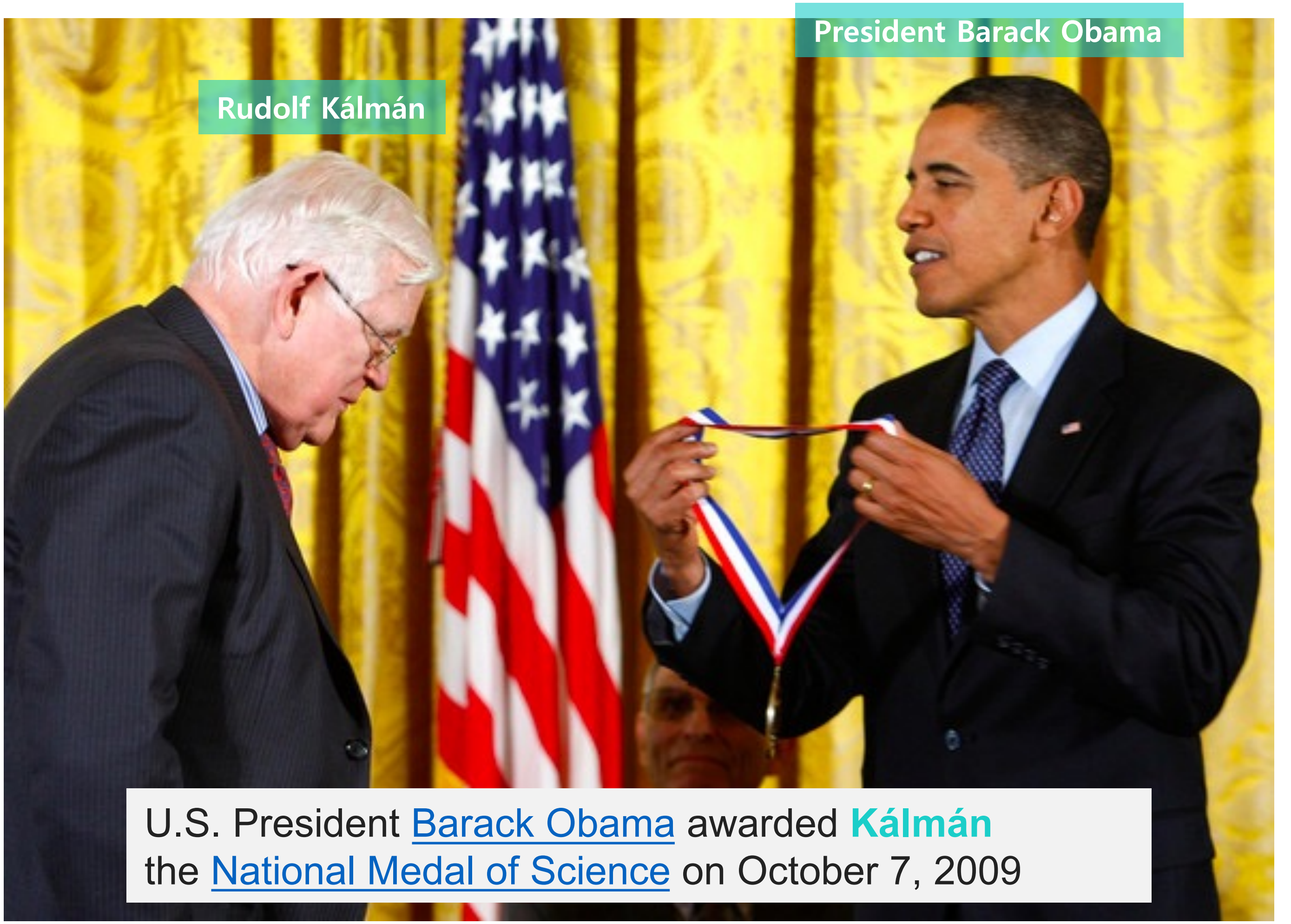
Kalman filter

- The Kalman filter is a recursive algorithm to estimate the parameters of dynamic linear models

- For non-linear systems,
  – Extended Kalman filter or unscented Kalman filter may be used

- For non-linear and non-Gaussian systems,
  – Particle filter may be useful

Kalman filter



U.S. President Barack Obama awarded **Kálmán** the National Medal of Science on October 7, 2009

Section 6.9

Hidden Markov Models and Switching Autoregression

State space models: Markov chain approach

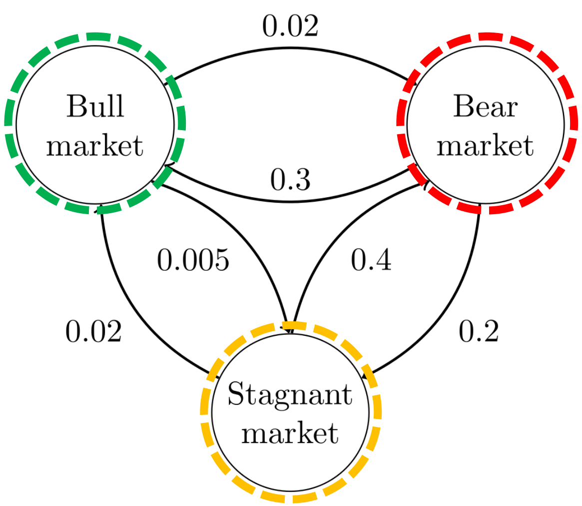
- We have been focusing on linear Gaussian models
  - State process x_t follows linear state equations and errors are Gaussian
- In this section, we will see the case when the state process x_t is a discrete-valued Markov chain
  - Value of the state at time t specifies the distribution of the observation at time t
  - The model was developed in Goldfeld and Quandt (1973) and Lindgren (1978)
    - May see many applications including Quandt (1978), Juang and Rabiner (1985), Bar-Shalom (1978), Hamilton (1989), McCulloch and Tsay (1993)

State space models: Markov chain approach

- **Markov chain**: It is a collection of random variables $\{x_t\}$ having the property that, given the present, the future is conditionally independent of the past
  - i.e.,
    $P(x_t = j \mid x_0 = i_0, x_1 = i_1, \ldots, x_{t-1} = i_{t-1}) = P(x_t = j \mid x_{t-1} = i_{t-1})$
  - Hence, a Markov chain is **memoryless** and this is called the **Markov property**
  - For the finite state space (i.e., x_t can take values in, say, {1,…,m}), we have
    - Stationary distribution $\pi_j = P(x_t = j)$
    - Stationary transition probabilities $\pi_{ij} = P(x_{t+1} = j \mid x_t = i)$
    - Also, we can specify the distributions $p_j(y_t) = P(y_t \mid x_t = j)$

State space models: Markov chain approach

- **Markov chain**: It is a collection of random variables $\{x_t\}$ having the property that, given the present, the future is conditionally independent of the past
  - **Example**: Suppose there are three states {Bull, Bear, Stagnant} in a financial market

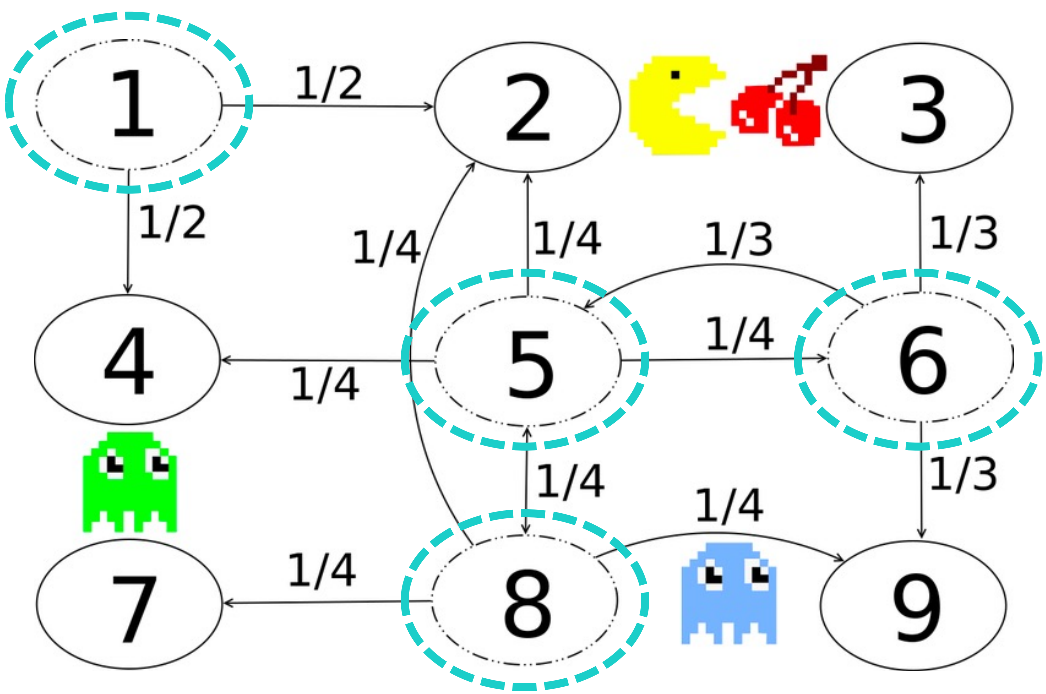


$$P = \left[\pi_{ij}\right]_{i,j=1}^{3}$$

$$= \begin{bmatrix} 0.93 & 0.02 & 0.05 \\ 0.3 & 0.5 & 0.2 \\ 0.02 & 0.4 & 0.58 \end{bmatrix}$$

State space models: Markov chain approach

- **Markov chain**: It is a collection of random variables $\{x_t\}$ having the property that, given the present, the future is conditionally independent of the past
  - **Example**: Movement of ghosts in Pac-Man



Hidden Markov models

- In the Markov chain approach, the dynamics of the system at time t are generated by one of m possible regimes evolving according to a Markov chain over time

- The case in which the particular regime is **unknown** to the observer is called the **Hidden Markov model (HMM)**
  - Although HMM satisfies the conditions to be classified as a state space model, it was developed in parallel

Hidden Markov models

- **Hidden Markov model (HMM)**
  - Can be seen as a dynamic version of Gaussian mixture models
    - Widely used unsupervised machine learning technique
      - e.g., speech recognition, signal treatment, etc.
  - Again, the transition of regimes (states) follows a Markov chain, and the regimes are hidden (unobservable)
    - All we can observe is the outcome
  - Need to estimate the most likely regime parameters (distributions) and their dynamics (transition matrix) based on the observations and specified number of regimes
    - Usually, the Baum-Welch algorithm is used (it is a special case of the EM algorithm)

A Gamble example: Hidden Markov Game

Available data:
Sequence of outcomes

:

HMM gives us

“Forward-Looking” Regime Identification

Deck 1

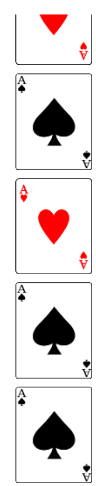
α

1- α

Deck 2

β

1-β



From Deck *i* with prob P

From Deck *i* with prob P

From Deck *i* with prob P

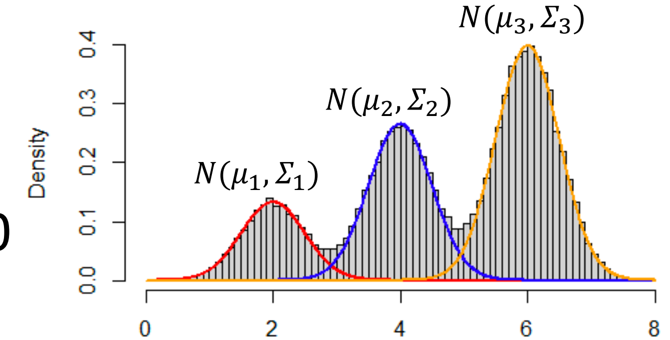
From Deck *i* with prob P

From Deck *i* with prob P

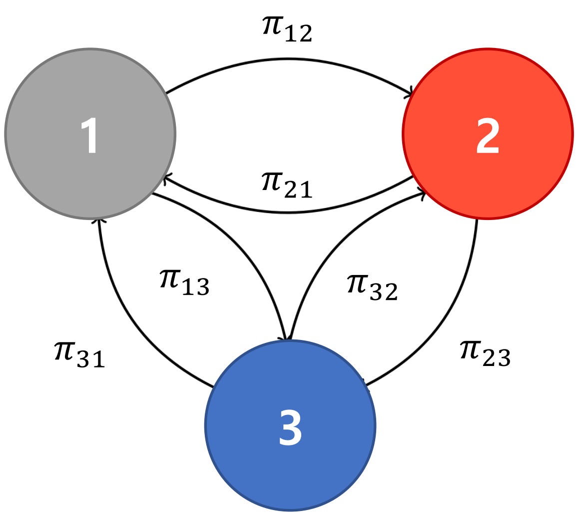
If Deck 1 was chosen at T-1
- Deck 1 is chosen with p chance at T
- Deck 2 is chosen with 1-p chance at T

If Deck 2 was chosen at T-1
- Deck 1 is chosen with q chance at T
- Deck 2 is chosen with 1-q chance at T

Hidden Markov model



- **Example**: S&P500 weekly returns
  - **Observations**: weekly returns of S&P500
  - **Assumption**: 3 states
    - Returns are generated from 3 different Gaussian distributions
      - i.e., $y_t \mid x_t = j \sim N(\mu_j, \Sigma_j)$ for $j = 1,2,3$
  - **Objective**: find the most likely μ_i, Σ_i and π_{ij} given observations



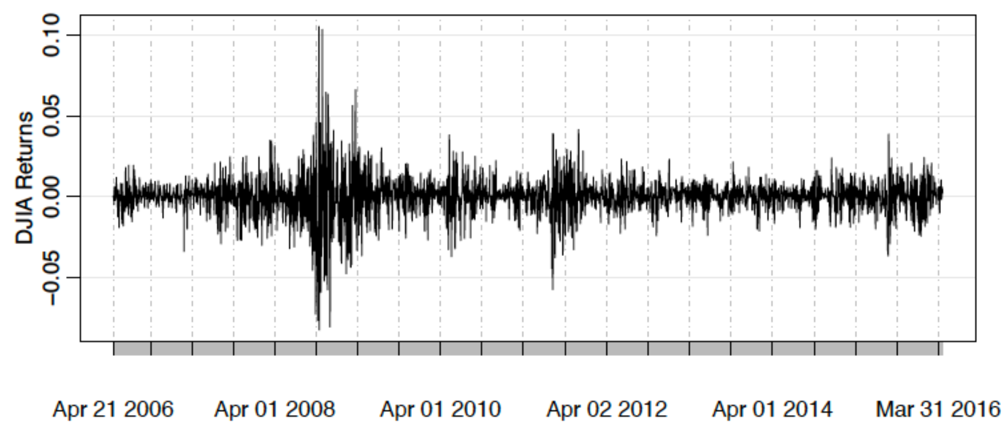
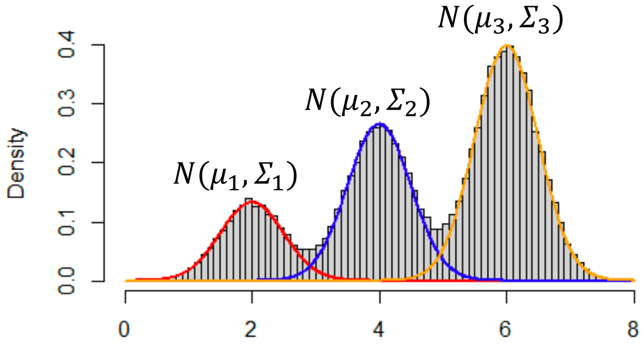
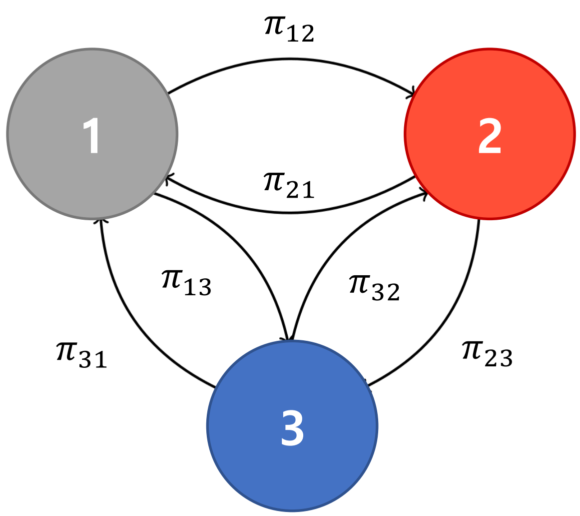


Fig. 1.4. The daily returns of the Dow Jones Industrial Average (DJIA) from April 20, 2006 to April 20, 2016.

Hidden Markov model

- **Example**: S&P500 weekly returns
  - **Observations**: weekly returns of S&P500
  - **Assumption**: 3 states
  - **Objective**: find the most likely μ_i, Σ_i and π_{ij} given observations





Estimated parameters

$$N(\hat{\mu}_1 = .004_{(.173)}, \hat{\sigma}_1 = .014_{(.968)})$$

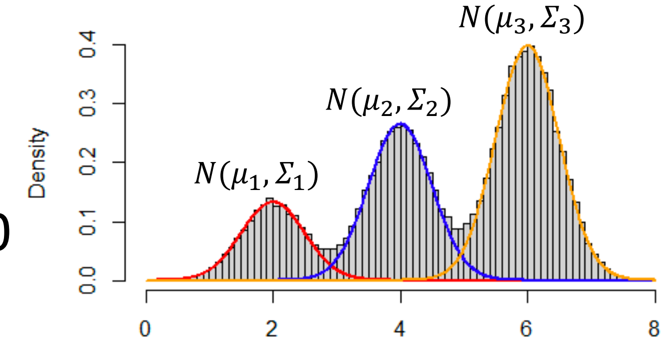
$$N(\hat{\mu}_2 = -.034_{(.909)}, \hat{\sigma}_2 = .009_{(.777)}).$$

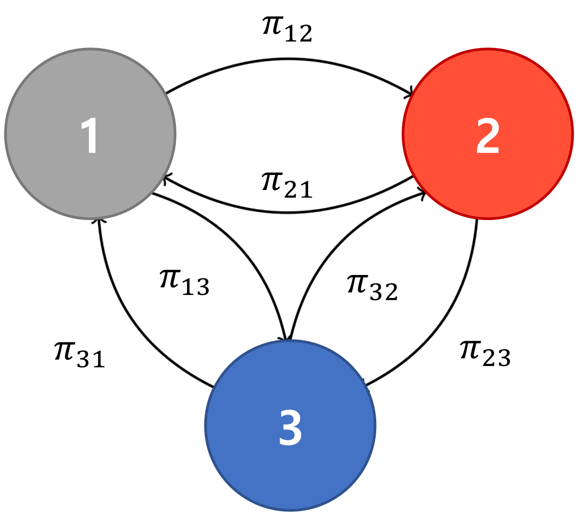
$$N(\hat{\mu}_3 = -.003_{(.317)}, \hat{\sigma}_3 = .044_{(.910)}).$$

$$\widehat{P} = \begin{bmatrix} .945_{(.074)} & .055_{(.074)} & .000_{(.000)} \\ .739_{(.275)} & .000_{(.000)} & .261_{(.275)} \\ .032_{(.122)} & .027_{(.057)} & .942_{(.147)} \end{bmatrix}$$

Hidden Markov model

- **Example**: S&P500 weekly returns
  - **Observations**: weekly returns of S&P500
  - **Assumption**: 3 states
  - **Objective**: find the most likely μ_i, Σ_i and π_{ij} given observations





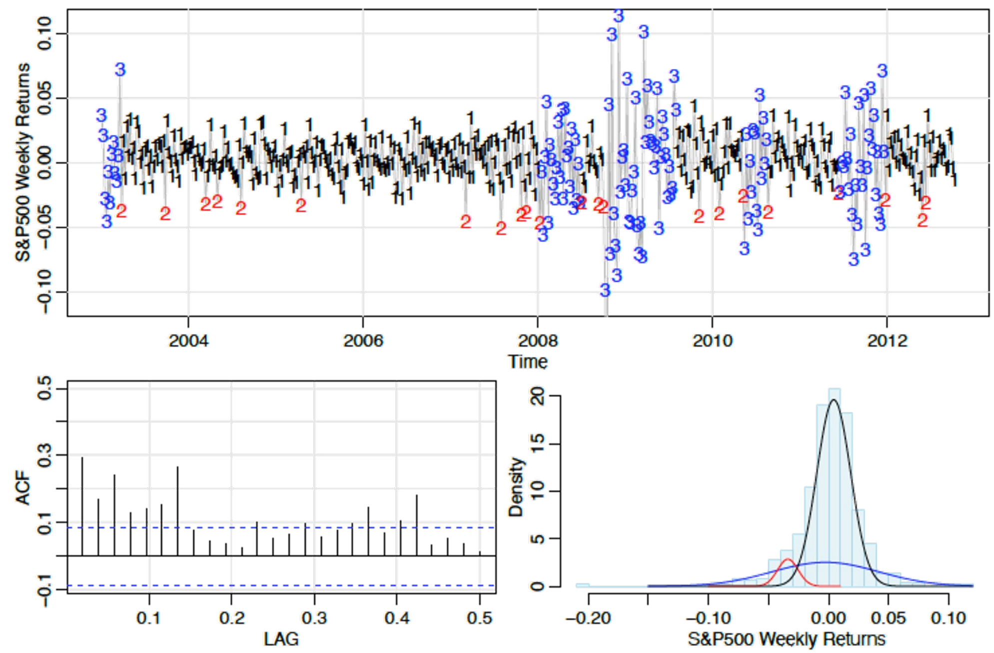
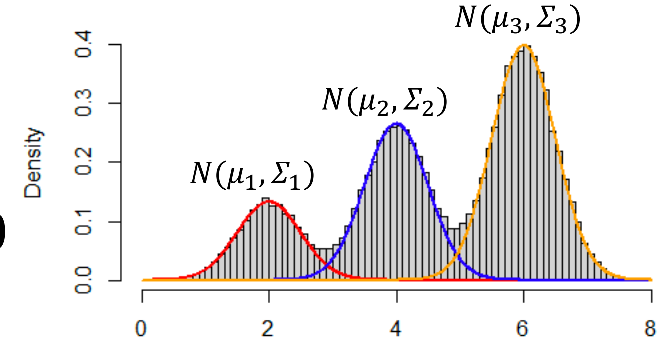
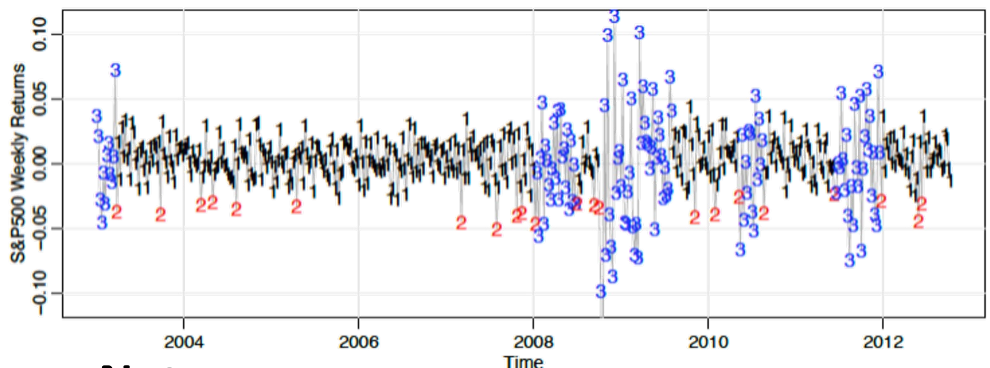


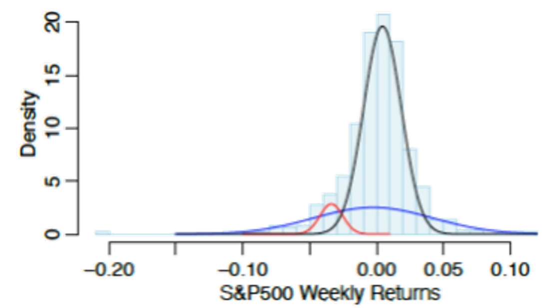
Fig. 6.14. Top: S&P 500 weekly returns with estimated regimes labeled as a number, 1, 2, or 3 The minimum value of −20% during the financial crisis has been truncated to improve the graphics. Bottom left: Sample ACF of the squared ruturns. Bottom right: Histogram of the data with the three estimated normal densities superimposed.

Hidden Markov model

- **Example**: S&P500 weekly returns
  - **Observations**: weekly returns of S&P500
  - **Assumption**: 3 states
  - **Objective**: find the most likely μ_i, Σ_i and π_{ij} given observations



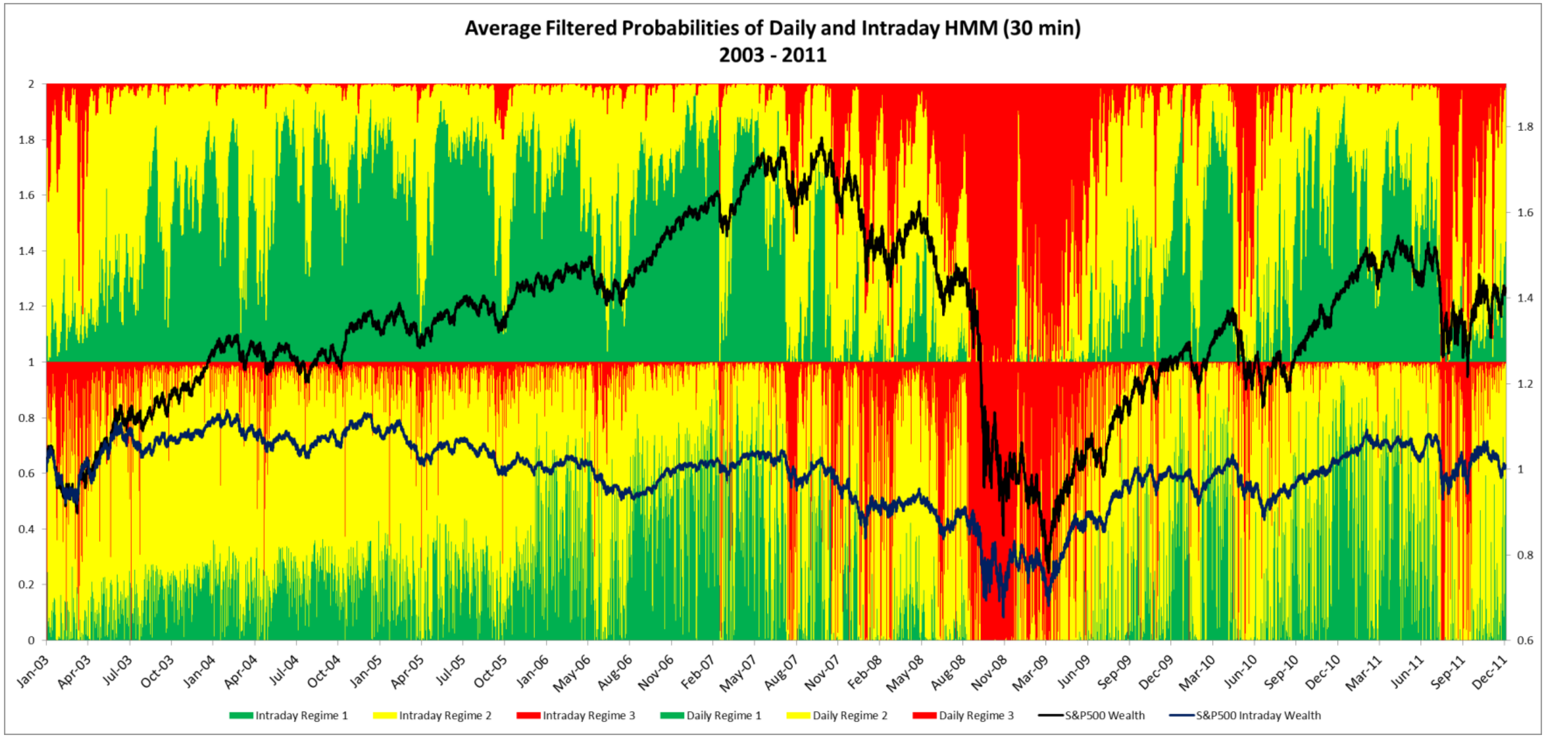




  - **Note**
    - It can be seen as
      - Regime 1: moderate market, Regime 2: bear market, Regime 3: volatile market
    - Note that we specified nothing about the characteristics of each regime other than normality and the number of regimes (interpretations should be made by humans)

Hidden Markov model

- **Example**: developing trading strategies



Hidden Markov model

- **Example**: developing trading strategies

Performance Statistics				
	S&P500	Daily	Intraday	WA
Return	2.56%	12.71%	2.95%	13.18%
Volatility	17.21%	9.59%	13.69%	9.32%
Sharpe Ratio	-0.0254	1.0124	-0.0035	1.0927
Max DD	57.29%	8.95%	36.80%	8.50%
Return / Max DD	0.0447	1.4192	0.0802	1.5511
Position Change	0	330	7346	724