Machine Learning for Signal Processing
Predicting and Estimation from Time Series

Bhiksha Raj
15 Nov 2016
Preliminaries : \( P(y|x) \) for Gaussian

- If \( P(x,y) \) is Gaussian:
  \[
P(x, y) = N\left( \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} C_{xx} & C_{xy} \\ C_{yx} & C_{yy} \end{bmatrix} \right)
  \]

- The conditional probability of \( y \) given \( x \) is also Gaussian
  - The slice in the figure is Gaussian

\[
P(y|x) = N\left( \mu_y + C_{yx}C_{xx}^{-1}(x - \mu_x), C_{yy} - C_{yx}C_{xx}^{-1}C_{xy} \right)
\]

- The mean of this Gaussian is a function of \( x \)
- The variance of \( y \) reduces if \( x \) is known
  - Uncertainty is reduced
Preliminaries: $P(y|\mathbf{x})$ for Gaussian

Best guess for $y$ when $\mathbf{x}$ is not known

$$P(y|\mathbf{x}) = \mathcal{N}(\mu_y + C_{yx} C^{-1}_{xx} (\mathbf{x} - \mu_x), C_{yy} - C_{yx} C^{-1}_{xx} C_{xy})$$
Preliminaries: $P(y|x)$ for Gaussian

Correction of $Y$ using information in $X$

Best guess for $Y$ when $X$ is not known

Correction is 0 if $X$ and $Y$ are uncorrelated, i.e $C_{yx} = 0$

Mean of $Y$ given $X$

Given $X$ value

$$P(y|x) = N(\mu_y + C_{yx} C_{xx}^{-1} (x - \mu_x), C_{yy} - C_{yx} C_{xx}^{-1} C_{xy})$$
Preliminaries : $P(y \mid x)$ for Gaussian

Correction to $Y = \text{slope} \times \text{(offset of } X \text{ from mean)}$

Best guess for $Y$ when $X$ is not known

Correction of $Y$ using information in $X$

Mean of $Y$ given $X$

Given $X$ value

$P(y \mid x) = N(\mu_y + C_{yx} C_{xx}^{-1}(x - \mu_x), C_{yy} - C_{yx} C_{xx}^{-1} C_{xy})$
Preliminaries: $P(y|x)$ for Gaussian

Correction of $Y$ using information in $X$

Best guess for $Y$ when $X$ is not known

Uncertainty in $Y$ when $X$ is not known

$$P(y|x) = N(\mu_y + C_{yx}C_{xx}^{-1}(x - \mu_x), C_{yy} - C_{yx}C_{xx}^{-1}C_{xy})$$
Preliminaries: $P(y \mid x)$ for Gaussian

Shrinkage of variance is 0 if $X$ and $Y$ are uncorrelated, i.e $C_{yx} = 0$

Correction of $Y$ using information in $X$

Best guess for $Y$ when $X$ is not known

Reduced uncertainty from knowing $X$

Uncertainty in $Y$ when $X$ is not known

Shrinkage of uncertainty from knowing $X$

\[
P(y \mid x) = N(\mu_y + C_{yx} C_{xx}^{-1} (x - \mu_x), C_{yy} - C_{yx} C_{xx}^{-1} C_{xy})
\]
Preliminaries: $P(y|x)$ for Gaussian

Knowing $X$ modifies the mean of $Y$ and shrinks its variance

Overall variance of $Y$ when $X$ is unknown

Variance of $Y$ when $X$ is known

Mean of $Y$ given $X$ (MAP estimate of $Y$)

Given $X$ value

$$P(y|x) = N(\mu_y + C_{yx}C_{xx}^{-1}(x - \mu_x), C_{yy} - C_{yx}C_{xx}^{-1}C_{xy})$$
The little parable

You’ve been kidnapped

And blindfolded

You can only hear the car
You must find your way back home from wherever they drop you off
Kidnapped!

- Determine by only *listening* to a running automobile, if it is:
  - Idling; or
  - Travelling at constant velocity; or
  - Accelerating; or
  - Decelerating

- You only record energy level (SPL) in the sound
  - The SPL is measured once per second
What we know

• An automobile that is at rest can accelerate, or continue to stay at rest

• An accelerating automobile can hit a steady-state velocity, continue to accelerate, or decelerate

• A decelerating automobile can continue to decelerate, come to rest, cruise, or accelerate

• A automobile at a steady-state velocity can stay in steady state, accelerate or decelerate
What else we know

- The probability distribution of the SPL of the sound is different in the various conditions
  - As shown in figure
    - In reality, depends on the car
- The distributions for the different conditions overlap
  - Simply knowing the current sound level is not enough to know the state of the car
• The state-space model
  – Assuming all transitions from a state are equally probable
  – This is a Hidden Markov Model!
Estimating the state at $T = 0$:

- A $T=0$, before the first observation, we know nothing of the state
  - Assume all states are equally likely

<table>
<thead>
<tr>
<th>State</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idling</td>
<td>0.25</td>
</tr>
<tr>
<td>Declerating</td>
<td>0.25</td>
</tr>
<tr>
<td>Cruising</td>
<td>0.25</td>
</tr>
<tr>
<td>Accelerating</td>
<td>0.25</td>
</tr>
</tbody>
</table>
The first observation: \( T=0 \)

- At \( T=0 \) you observe the sound level \( x_0 = 68 \text{dB} \) SPL
  - The observation modifies our belief in the state of the system
The first observation: $T=0$

| $P(x|\text{idle})$ | $P(x|\text{deceleration})$ | $P(x|\text{cruising})$ | $P(x|\text{acceleration})$ |
|---------------------|----------------------------|------------------------|-----------------------------|
| 0                   | 0.0001                     | 0.5                    | 0.7                         |

These don’t have to sum to 1

Can even be greater than 1!
The first observation: \( T=0 \)

\[
\begin{align*}
P(x|\text{idle}) & \quad 45 \\
P(x|\text{decel}) & \quad 60 \\
P(x|\text{cruise}) & \quad 65 \\
P(x|\text{accel}) & \quad 70
\end{align*}
\]

\( P(x_0|\text{state}) \)

\[
\begin{align*}
\text{Idling} & \quad 0 \\
\text{Declerating} & \quad 0.0001 \\
\text{Cruising} & \quad 0.5 \\
\text{Accelerating} & \quad 0.7
\end{align*}
\]

Remember the prior

\[
\begin{align*}
\text{Prior: } P(\text{state}) & \\
0.25 & \quad 0.25 \\
\text{Idling} & \quad \text{Declerating} \\
0.25 & \quad 0.25 \\
\text{Cruising} & \quad \text{Accelerating}
\end{align*}
\]
Estimating state after at observing $x_0$

- Combine prior information about state and evidence from observation
- We want $P(state|x_0)$
- We can compute it using Bayes rule as

$$P(state|x_0) = \frac{P(state)P(x_0|state)}{\sum_{state'} P(state')P(x_0|state')}$$
• Multiply the two, term by term, and normalize them so that they sum to 1.0
Estimating the state at $T = 0^+$

- At $T=0$, after the first observation $x_0$, we update our belief about the states
  - The first observation provided some evidence about the state of the system
  - It modifies our belief in the state of the system
Predicting the state at $T=1$

- Predicting the probability of idling at $T=1$
  - $P(idling \mid idling) = 0.5$;
  - $P(idling \mid deceleration) = 0.25$
  - $P(idling \text{ at } T=1 \mid x_0) = P(I_{T=0} \mid x_0) \cdot P(I \mid I) + P(D_{T=0} \mid x_0) \cdot P(I \mid D) = 2.1 \times 10^{-5}$

- In general, for any state $S$
  - $P(S_{T=1} \mid x_0) = \sum_{S_{T=0}} P(S_{T=0} \mid x_0) \cdot P(S_{T=1} \mid S_{T=0})$
Predicting the state at $T = 1$

$$P(S_{T=0}|x_0)$$

<table>
<thead>
<tr>
<th>State</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idling</td>
<td>0.0</td>
</tr>
<tr>
<td>Decelerating</td>
<td>8.3 x 10^{-5}</td>
</tr>
<tr>
<td>Cruising</td>
<td>0.42</td>
</tr>
<tr>
<td>Accelerating</td>
<td>0.57</td>
</tr>
</tbody>
</table>

$$P(S_{T=1}|x_0) = \sum_{S_{T=0}} P(S_{T=0}|x_0)P(S_{T=1}|S_{T=0})$$

$$P(S_{T=1}|x_0)$$

<table>
<thead>
<tr>
<th>State</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idling</td>
<td>0.33</td>
</tr>
<tr>
<td>Decelerating</td>
<td>0.33</td>
</tr>
<tr>
<td>Cruising</td>
<td>0.33</td>
</tr>
<tr>
<td>Accelerating</td>
<td>2.1 x 10^{-5}</td>
</tr>
</tbody>
</table>

Rounded. In reality, they sum to 1.0
Updating after the observation at $T=1$

- At $T=1$ we observe $x_1 = 63$ dB SPL
Updating after the observation at $T=1$

$$P(x_{1}|\text{state})$$

| $P(x|\text{idle})$ | $P(x|\text{deceleration})$ | $P(x|\text{cruising})$ | $P(x|\text{acceleration})$ |
|---------------------|-----------------------------|------------------------|-----------------------------|
| 0                   | 0.2                         | 0.5                    | 0.01                        |

$63\text{dB}$
The first observation: T=0

\[ P(x_{1}|state) \]

\[
\begin{array}{c|c|c|c}
\text{State} & \text{Idling} & \text{Declerating} & \text{Cruising} & \text{Accelerating} \\
\hline
P(x_{1}|state) & 0.2 & 0.5 & 0.02 \\
\end{array}
\]

\[ Prior: \quad P(state|x_0) \]

\[
\begin{array}{c|c|c|c}
\text{State} & \text{Idling} & \text{Declerating} & \text{Cruising} & \text{Accelerating} \\
\hline
P(state|x_0) & 0.33 & 0.33 & 0.33 \\
\end{array}
\]

Remember the prior

\[ 2.1 \times 10^{-5} \]
Estimating state after at observing $x_1$

• Combine prior information from the observation at time $T=0$, AND evidence from observation at $T=1$ to estimate state at $T=1$

• We want $P(state|x_0, x_1)$

• We can compute it using Bayes rule as

$$P(state|x_0, x_1) = \frac{P(state|x_0)P(x_1|state)}{\sum_{state'} P(state'|x_0)P(x_1|state')}$$
The Posterior at $T = 1$

- Multiply the two, term by term, and normalize them so that they sum to 1.0
Estimating the state at $T = 1^+$

- The updated probability at $T=1$ incorporates information from both $x_0$ and $x_1$
  - It is NOT a local decision based on $x_1$ alone
  - Because of the Markov nature of the process, the state at $T=0$ affects the state at $T=1$
    - $x_0$ provides evidence for the state at $T=1$
# Overall Process

<table>
<thead>
<tr>
<th>Time</th>
<th>Computation</th>
</tr>
</thead>
<tbody>
<tr>
<td>T=0- : A priori probability</td>
<td>$P(S_0) = P(S)$</td>
</tr>
<tr>
<td>T = 0+: Update after $X_0$</td>
<td>$P(S_0</td>
</tr>
<tr>
<td>T=1- (Prediction before $X_1$)</td>
<td>$P(S_1</td>
</tr>
<tr>
<td>T = 1+: Update after $X_1$</td>
<td>$P(S_1</td>
</tr>
<tr>
<td>T=2- (Prediction before $X_2$)</td>
<td>$P(S_2</td>
</tr>
<tr>
<td>T = 2+: Update after $X_2$</td>
<td>$P(S_2</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>T= t- (Prediction before $X_t$)</td>
<td>$P(S_t</td>
</tr>
<tr>
<td>T = t+: Update after $X_t$</td>
<td>$P(S_t</td>
</tr>
</tbody>
</table>
Overall procedure

- At T=0 the predicted state distribution is the initial state probability
- At each time T, the current estimate of the distribution over states considers all observations $x_0 \ldots x_T$
  - A natural outcome of the Markov nature of the model
- The prediction+update is identical to the forward computation for HMMs to within a normalizing constant

$$P(S_T \mid x_{0:T-1}) = \sum_{S_{T-1}} P(S_{T-1} \mid x_{0:T-1}) P(S_T \mid S_{T-1})$$

$$P(S_T \mid x_{0:T}) = C. P(S_T \mid x_{0:T-1}) P(x_T \mid S_T)$$

Predict the distribution of the state at T

Update the distribution of the state at T after observing $x_T$
Decomposing the Algorithm

\[ P(S_t, X_{0:t}) = P(X_t | S_t) \sum_{S_{t-1}} P(S_{t} | S_{t-1}) P(S_{t-1}, X_{0:t-1}) \]

Predict: \[ P(S_t | X_{0:t-1}) = \sum_{S_{t-1}} P(S_{t} | S_{t-1}) P(S_{t-1} | X_{0:t-1}) \]

Update: \[ P(S_t | X_{0:t}) = \frac{P(S_t | X_{0:t-1}) P(X_t | S_t)}{\sum_S P(S | X_{0:t-1}) P(X_t | S)} \]
Estimating a Unique state

• What we have estimated is a *distribution* over the states

• If we had to guess a state, we would pick the most likely state from the distributions

• State(T=0) = Accelerating

• State(T=1) = Cruising
Estimating the state

The state is estimated from the updated distribution

- The updated distribution is propagated into time, not the state
A continuous state model

• HMM assumes a very coarsely quantized state space
  – Idling / accelerating / cruising / decelerating

• Actual state can be finer
  – Idling, accelerating at various rates, decelerating at various rates, cruising at various speeds

• Solution: Many more states (one for each acceleration / deceleration rate, cruising speed)?

• Solution: A continuous valued state
Tracking and Prediction: The wind and the target

- **Aim**: measure wind velocity
- **Using a noisy wind speed sensor**: E.g. arrows shot at a target

- **State**: Wind speed at time $t$ depends on speed at time $t-1$
  \[
  S_t = S_{t-1} + \epsilon_t
  \]

- **Observation**: Arrow position at time $t$ depends on wind speed at time $t$
  \[
  Y_t = AS_t + \gamma_t
  \]
The real-valued state model

• A state equation describing the dynamics of the system

\[ s_t = f(s_{t-1}, \varepsilon_t) \]

  – \( s_t \) is the state of the system at time \( t \)
  – \( \varepsilon_t \) is a driving function, which is assumed to be random

• The state of the system at any time depends only on the state at the previous time instant and the driving term at the current time

• An observation equation relating state to observation

\[ o_t = g(s_t, \gamma_t) \]

  – \( o_t \) is the observation at time \( t \)
  – \( \gamma_t \) is the noise affecting the observation (also random)

• The observation at any time depends only on the current state of the system and the noise
States are still “hidden”

- The state is a continuous valued parameter that is not directly seen
  - The state is the position of the automobile or the star

- The observations are dependent on the state and are the only way of knowing about the state
  - Sensor readings (for the automobile) or recorded image (for the telescope)

\[ s_t = f(s_{t-1}, \varepsilon_t) \]
\[ o_t = g(s_t, \gamma_t) \]
Discrete vs. Continuous state systems

Prediction at time 0:

\[ P(S_0) = \pi(S_0) \]

Update after \( O_0 \):

\[ P(S_0|O_0) = C \cdot \pi(S_0)P(O_0|S_0) \]

Prediction at time 1:

\[ P(S_1|O_0) = \sum_{S_0} P(S_0|O_0)P(S_1|S_0) \]

Update after \( O_1 \):

\[ P(S_1|O_{0:1}) = C \cdot P(S_1|O_0)P(O_1|S_1) \]

\[ s_t = f(s_{t-1}, \varepsilon_t) \]

\[ o_t = g(s_t, \gamma_t) \]
Discrete vs. Continuous State Systems

Prediction at time $t$:

$$P(S_t|O_{0:t-1}) = \sum_{S_{t-1}} P(S_{t-1}|O_{0:t-1})P(S_t|S_{t-1})$$

Update after observing $O_t$:

$$P(S_t|O_{0:t}) = C \cdot P(S_t|O_{0:t-1})P(O_t|S_t)$$

$$P(S_t|O_{0:t}) = C \cdot P(S_t|O_{0:t-1})P(O_t|S_t)$$

$$s_t = f(s_{t-1}, \varepsilon_t)$$

$$o_t = g(s_t, \gamma_t)$$
**Discrete vs. Continuous State Systems**

\[ \pi = \begin{array}{c|ccc} & 0 & 1 & 2 & 3 \\ \hline 0 & 0.1 & 0.2 & 0.3 & 0.4 \\ \end{array} \]

**Parameters**

Initial state prob. \( \pi \)

Transition prob \( \{T_{ij}\} = P(s_t = j \mid s_{t-1} = i) \)

Observation prob \( P(O \mid s) \)

\[ s_t = f(s_{t-1}, \varepsilon_t) \]

\[ o_t = g(s_t, \gamma_t) \]

\[ P(s) \]

\[ P(s_t \mid s_{t-1}) \]

\[ P(o \mid s) \]
Special case: Linear Gaussian model

\[ s_t = A_t s_{t-1} + \varepsilon_t \]

\[ o_t = B_t s_t + \gamma_t \]

\[ P(\varepsilon) = \frac{1}{\sqrt{(2\pi)^d |\Theta_\varepsilon|}} \exp\left( -0.5(\varepsilon - \mu_\varepsilon)^T \Theta_\varepsilon^{-1}(\varepsilon - \mu_\varepsilon) \right) \]

\[ P(\gamma) = \frac{1}{\sqrt{(2\pi)^d |\Theta_\gamma|}} \exp\left( -0.5(\gamma - \mu_\gamma)^T \Theta_\gamma^{-1}(\gamma - \mu_\gamma) \right) \]

- A **linear** state dynamics equation
  - Probability of state driving term \( \varepsilon \) is Gaussian
  - Sometimes viewed as a driving term \( \mu_\varepsilon \) and additive zero-mean noise

- A **linear** observation equation
  - Probability of observation noise \( \gamma \) is Gaussian

- \( A_t, B_t \) and Gaussian parameters assumed known
  - May vary with time
Linear model example
The wind and the target

• **State:** Wind speed at time $t$ depends on speed at time $t-1$

  $$S_t = S_{t-1} + \epsilon_t$$

• **Observation:** Arrow position at time $t$ depends on wind speed at time $t$

  $$O_t = BS_t + \gamma_t$$
Model Parameters:

The initial state probability

\[
P_0(s) = \frac{1}{\sqrt{(2\pi)^d | R|}} \exp\left(-0.5(s - \bar{s})R^{-1}(s - \bar{s})^T\right)
\]

\[
P_0(s) = \text{Gaussian}(s; \bar{s}, R)
\]

- We also assume the \textit{initial} state distribution to be Gaussian
  - Often assumed zero mean

\[
s_t = A_ts_{t-1} + \varepsilon_t
\]

\[
o_t = B_ts_t + \gamma_t
\]
Model Parameters: The observation probability

\[ o_t = B_t s_t + \gamma_t \]

\[ P(\gamma) = Gaussian(\gamma; \mu_\gamma, \Theta_\gamma) \]

\[ P(o_t \mid s_t) = Gaussian(o_t; \mu_\gamma + B_t s_t, \Theta_\gamma) \]

- The probability of the observation, given the state, is simply the probability of the noise, with the mean shifted
  - Since the only uncertainty is from the noise

- The new mean is the mean of the distribution of the noise + the value of the observation in the absence of noise
Model Parameters: State transition probability

\[ s_{t+1} = A_t s_t + \varepsilon_t \]

\[ P(\varepsilon) = \text{Gaussian}(\varepsilon; \mu_\varepsilon, \Theta_\varepsilon) \]

\[ P(s_{t+1} \mid s_t) = \text{Gaussian}(s_t; \mu_\varepsilon + A_t s_t, \Theta_\varepsilon) \]

- The probability of the state at time \( t \), given the state at \( t-1 \), is simply the probability of the driving term, with the mean shifted
Gaussian Continuous State Linear Systems

\[ s_{t+1} = A_t s_t + \epsilon_t \]
\[ o_t = B_t s_t + \gamma_t \]

Prediction at time \( t \):
\[ P(S_t | O_{0:t-1}) = \int_{-\infty}^{\infty} P(S_{t-1} | O_{0:t-1}) P(S_t | S_{t-1}) dS_{t-1} \]

Update after observing \( O_t \):
\[ P(S_t | O_{0:t}) = C \cdot P(S_t | O_{0:t-1}) P(O_t | S_t) \]
Gaussian Continuous State Linear Systems

Prediction at time $t$:

$$P(S_t | O_{0:t-1}) = N(\bar{s}_t, R_t)$$

Update after observing $O_t$:

$$P(S_t | O_{0:t}) = N(\hat{s}_t, \hat{R}_t)$$

1. Update after observing $O_t$:
   $$\bar{s}_t = A\hat{s}_{t-1} + \mu_\varepsilon$$
   $$R_t = \theta_\varepsilon + A\hat{R}_{t-1}A^T$$

2. Update after observing $O_t$:
   $$\hat{s}_t = \bar{s}_t + K_t (O_t - B\bar{s}_t - \mu_\gamma)$$
   $$\hat{R}_t = (I - K_t B) R_t$$

$$s_{t+1} = A_t s_t + \varepsilon_t$$

$$o_t = B_t s_t + \gamma_t$$
Gaussian Continuous State Linear Systems

Prediction at time $t$:

$$P(S_t|O_{0:t-1}) = N(\bar{s}_t, R_t)$$

Update after observing $O_t$:

$$P(S_t|O_{0:t}) = N(\hat{s}_t, \hat{R}_t)$$

\[ s_{t+1} = A_t s_t + \varepsilon_t \]
\[ o_t = B_t s_t + \gamma_t \]

\[
\bar{s}_t = A\hat{s}_{t-1} + \mu\varepsilon \\
R_t = \Theta\varepsilon + A\hat{R}_{t-1}A^T
\]

\[
K_t = R_1B^T(BR_1B^T + \Theta\gamma)^{-1} \\
\hat{s}_t = \bar{s}_t + K_t(O_t - B\bar{s}_t - \mu\gamma) \\
\hat{R}_t = (I - K_tB) R_t
\]
The Kalman filter

• Prediction (based on state equation)

\[ \bar{s}_t = A_t \hat{s}_{t-1} + \mu_\varepsilon \]

\[ s_t = A_t s_{t-1} + \varepsilon_t \]

\[ R_t = \Theta_\varepsilon + A_t \hat{R}_{t-1} A_t^T \]

• Update (using observation and observation equation)

\[ K_t = R_t B_t^T \left( B_t R_t B_t^T + \Theta_\gamma \right)^{-1} \]

\[ o_t = B_t \bar{s}_t + \gamma_t \]

\[ \hat{s}_t = \bar{s}_t + K_t \left( o_t - B_t \bar{s}_t - \mu_\gamma \right) \]

\[ \hat{R}_t = (I - K_t B_t) R_t \]
Explaining the Kalman Filter

- **Prediction**

\[ \bar{s}_t = A_t \hat{s}_{t-1} + \mu_\epsilon \]

\[ R_t = \Theta_\epsilon + A_t \hat{R}_{t-1} A_t^T \]

- The Kalman filter can be explained intuitively without working through the math

\[ \hat{s}_t = \bar{s}_t + K_t (o_t - B_t \bar{s}_t - \mu_\gamma) \]

\[ \hat{R}_t = (I - K_t B_t) R_t \]
The Kalman filter

- Prediction

\[ \tilde{s}_t = A_t \hat{s}_{t-1} + \mu \varepsilon \]

\[ s_t = A_t s_{t-1} + \varepsilon_t \]

\[ o_t = B_t s_t + \gamma_t \]

The *predicted* state at time \( t \) is obtained simply by propagating the estimated state at \( t-1 \) through the state dynamics equation:

\[ K_t = K_t B_t (B_t K_t B_t + \Theta \gamma) \]

\[ \hat{s}_t = \tilde{s}_t + K_t (o_t - B_t \tilde{s}_t - \mu \gamma) \]

\[ \hat{R}_t = (I - K_t B_t) R_t \]
The Kalman filter

• Prediction

\[ s_t = A_t s_{t-1} + \varepsilon_t \]

\[ o_t = B_t s_t + \gamma_t \]

\[ R_t = \Theta \varepsilon + A_t \hat{R}_{t-1} A_t^T \]

This is the uncertainty in the prediction. The variance of the predictor = variance of \( \varepsilon_t \) + variance of \( As_{t-1} \)

The two simply add because \( \varepsilon_t \) is not correlated with \( s_t \)
The Kalman filter

• Prediction

\[ \bar{s}_t = A_t \hat{s}_{t-1} + \mu_{\varepsilon} \]

\[ R_t = \Theta_{\varepsilon} + A_t \hat{R}_{t-1} A_t^T \]

\[ o_t = B_t s_t + \gamma_t \]

\[ \hat{o}_t = B_t \bar{s}_t + \mu_{\gamma} \]

We can also predict the observation from the predicted state using the observation equation

\[ s_t = A_t s_{t-1} + \varepsilon_t \]

\[ \hat{R}_t = (I - K_t B_t) R_t \]
MAP Recap (for Gaussians)

- If $P(x,y)$ is Gaussian:

$$P(x,y) = N\left( \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} C_{xx} & C_{xy} \\ C_{yx} & C_{yy} \end{bmatrix} \right)$$

$$P(y \mid x) = N(\mu_y + C_{yx} C^{-1}_{xx} (x - \mu_x), C_{yy} - C_{yx} C^{-1}_{xx} C_{xy})$$

$$\hat{y} = \mu_y + C_{yx} C^{-1}_{xx} (x - \mu_x)$$
MAP Recap: For Gaussians

- If $P(x,y)$ is Gaussian:

$$P(y, x) = N \left( \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} C_{xx} & C_{xy} \\ C_{yx} & C_{yy} \end{bmatrix} \right)$$

$$P(y \mid x) = N \left( \mu_y + C_{yx} C_{xx}^{-1} (x - \mu_x), C_{yy} - C_{yx} C_{xx}^{-1} C_{xy} \right)$$

$$\hat{y} = \mu_y + C_{yx} C_{xx}^{-1} (x - \mu_x)$$

“Slope” of the line
The Kalman filter

- Prediction

\[ \bar{s}_t = A_t \hat{s}_{t-1} + \mu_\varepsilon \]

\[ R_t = \Theta_\varepsilon + A_t \hat{R}_{t-1} A_t^T \]

- Update

\[ s_t = A_t s_{t-1} + \varepsilon_t \]

\[ o_t = B_t s_t + \gamma_t \]

This is the slope of the MAP estimator that predicts \( s \) from \( o \)

\[ RB^T = C_{so}, \quad (BRB^T + \Theta) = C_{oo} \]

This is also called the Kalman Gain
The Kalman filter

- Prediction

\[ s_t = A_t s_{t-1} + \varepsilon_t \]

\[ R_t = \Theta_\varepsilon + A_t \bar{R}_{t-1} A_t^T \]

\[ \bar{s}_t = A_t \hat{s}_{t-1} + \mu_\varepsilon \]

We must correct the predicted value of the state after making an observation

\[ \hat{s}_t = \bar{s}_t + K_t (o_t - B_t \bar{s}_t - \mu_\gamma) \]

\[ \hat{o}_t = B_t \bar{s}_t + \mu_\gamma \]

The correction is the difference between the actual observation and the predicted observation, scaled by the Kalman Gain
The Kalman filter

• Prediction

\[ \bar{s}_t = A_t \hat{s}_{t-1} + \mu_\varepsilon \]

\[ s_t = A_t s_{t-1} + \varepsilon_t \]

\[ R_t = \Theta_\varepsilon + A_t \hat{R}_{t-1} A_t^T \]

• Update:

The uncertainty in state decreases if we observe the data and make a correction

The reduction is a multiplicative “shrinkage” based on Kalman gain and B

\[ \hat{R}_t = (I - K_t B_t)R_t \]

\[ \hat{o}_t = B_t \bar{s}_t + \mu_\gamma \]
The Kalman filter

• Prediction

\[ \bar{s}_t = A_t \hat{s}_{t-1} + \mu_\varepsilon \]

\[ R_t = \Theta_\varepsilon + A_t \hat{R}_{t-1} A_t^T \]

• Update:

\[ K_t = R_t B_t^T \left( B_t R_t B_t^T + \Theta_\gamma \right)^{-1} \]

\[ \hat{s}_t = \bar{s}_t + K_t \left( o_t - B_t \bar{s}_t - \mu_\gamma \right) \]

• Update

\[ \hat{R}_t = \left( I - K_t B_t \right) R_t \]
The Kalman Filter

• Very popular for tracking the state of processes
  – Control systems
  – Robotic tracking
    • Simultaneous localization and mapping
  – Radars
  – Even the stock market..

• What are the parameters of the process?
Kalman filter contd.

\[ s_t = A_t s_{t-1} + \varepsilon_t \]
\[ o_t = B_t s_t + \gamma_t \]

- Model parameters A and B must be known
  - Often the state equation includes an *additional* driving term:
    \[ s_t = A_t s_{t-1} + G_t u_t + \varepsilon_t \]
  - The parameters of the driving term must be known

- The initial state distribution must be known
Defining the parameters

• State state must be carefully defined
  – E.g. for a robotic vehicle, the state is an extended vector that includes the current velocity and acceleration
    • $S = [X, dX, d^2X]$

• State equation: Must incorporate appropriate constraints
  – If state includes acceleration and velocity, velocity at next time = current velocity + acc. * time step
    – $S_t = AS_{t-1} + e$
      • $A = [1 \ t \ 0.5t^2; \ 0 \ 1 \ t; \ 0 \ 0 \ 1]$
Parameters

• Observation equation:
  – Critical to have accurate observation equation
  – Must provide a valid relationship between state and observations

• Observations typically high-dimensional
  – May have higher or lower dimensionality than state
Problems

\[ s_t = f(s_{t-1}, e_t) \]
\[ o_t = g(s_t, y_t) \]

• \( f() \) and/or \( g() \) may not be nice linear functions
  – Conventional Kalman update rules are no longer valid

• \( e \) and/or \( y \) may not be Gaussian
  – Gaussian based update rules no longer valid
### Linear Gaussian Model

- **$s_t = A_t s_{t-1} + \epsilon_t$**
- **$o_t = B_t s_t + \gamma_t$**

**Prior Distribution:** 
\[
P(s_0) = P(s)
\]

**Transition Probability:** 
\[
P(s_1 | O_0) = C P(s_0) P(O_0 | s_0)
\]

\[
P(s_1 | O_0) = \int_{-\infty}^{\infty} P(s_0 | O_0) P(s_1 | s_0) ds_0
\]

**State Output Probability:** 
\[
P(s_1 | O_{0:1}) = C P(s_1 | O_0) P(O_1 | s_0)
\]

\[
P(s_2 | O_{0:1}) = \int_{-\infty}^{\infty} P(s_1 | O_{0:1}) P(s_2 | s_1) ds_1
\]

**State Output Probability:** 
\[
P(s_2 | O_{0:2}) = C P(s_2 | O_{0:1}) P(O_2 | s_2)
\]

All distributions remain Gaussian.
Problems

\[ s_t = f(s_{t-1}, \varepsilon_t) \]
\[ o_t = g(s_t, \gamma_t) \]

- Nonlinear \( f() \) and/or \( g() \) : The Gaussian assumption breaks down
  - Conventional Kalman update rules are no longer valid
The problem with non-linear functions

\[ s_t = f(s_{t-1}, e_t) \]

\[ o_t = g(s_t, \gamma_t) \]

\[
P(s_t \mid o_{0:t-1}) = \int_{-\infty}^{\infty} P(s_{t-1} \mid o_{0:t-1})P(s_t \mid s_{t-1})ds_{t-1}
\]

\[
P(s_t \mid o_{0:t}) = CP(s_t \mid o_{0:t-1})P(o_t \mid s_t)
\]

• Estimation requires knowledge of \( P(o \mid s) \)
  – Difficult to estimate for nonlinear \( g() \)
  – Even if it can be estimated, may not be tractable with update loop

• Estimation also requires knowledge of \( P(s_t \mid s_{t-1}) \)
  – Difficult for nonlinear \( f() \)
  – May not be amenable to closed form integration
The problem with nonlinearity

\[ o_t = g(s_t, \gamma_t) \]

- The PDF may not have a closed form

\[ P(o_t \mid s_t) = \sum_{\gamma : g(s_t, \gamma) = o_t} \frac{P_\gamma(\gamma)}{|J_{g(s_t, \gamma)}(o_t)|} \]

- Even if a closed form exists initially, it will typically become intractable very quickly
Example: a simple nonlinearity

\[ o = \gamma + \log(1 + \exp(s)) \]

- \( P(o \mid s) = ? \)
  - Assume \( \gamma \) is Gaussian
  - \( P(\gamma) = Gaussian(\gamma; \mu_\gamma, \Theta_\gamma) \)
Example: a simple nonlinearity

\[ o = γ + \log(1 + \exp(s)) \]

\[ P(o \mid s) = ? \]

\[ P(γ) = \text{Gaussian}(γ; μ_γ, Θ_γ) \]

\[ P(o \mid s) = \text{Gaussian}(o; μ_γ + \log(1 + \exp(s)), Θ_γ) \]
Example: At T=0.

\[ o = \gamma + \log(1 + \exp(s)) \]

- Assume initial probability \( P(s) \) is Gaussian

\[ P(s_0) = P_0(s) = \text{Gaussian}(s; \bar{s}, R) \]

- Update

\[ P(s_0 | o_0) = CP(o_0 | s_0)P(s_0) \]

\[ P(s_0 | o_0) = C\text{Gaussian}(o; \mu_\gamma + \log(1 + \exp(s_0)), \Theta_\gamma) \text{Gaussian}(s_0; \bar{s}, R) \]
UPDATE: At $T=0$.

\[ o = \gamma + \log(1 + \exp(s)) \]

\[
P(s_0 \mid o_0) = CGaussian(o; \mu_\gamma + \log(1 + \exp(s_0)), \Theta_\gamma)Gaussian(s_0; \bar{s}, R)
\]

\[
P(s_0 \mid o_0) = C \exp\left( -0.5(\mu_\gamma + \log(1 + \exp(s_0)) - o)^T \Theta^{-1}_\gamma (\mu_\gamma + \log(1 + \exp(s_0)) - o) - 0.5(s_0 - \bar{s})^T R^{-1}(s_0 - \bar{s}) \right)
\]

- = Not Gaussian
Prediction for $T = 1$

\[ S_t = S_{t-1} + \varepsilon \quad \text{and} \quad P(\varepsilon) = \text{Gaussian}(\varepsilon; 0, \Theta_{\varepsilon}) \]

- Trivial, linear state transition equation
  \[ P(s_t \mid s_{t-1}) = \text{Gaussian}(s_t; s_{t-1}, \Theta_{\varepsilon}) \]

- Prediction
  \[ P(s_1 \mid o_0) = \int_{-\infty}^{\infty} P(s_0 \mid o_0) P(s_1 \mid s_0) ds_0 \]

\[ P(s_1 \mid o_0) = \int_{-\infty}^{\infty} C \exp \left( -0.5 \left( \mu_\gamma \log(1 + \exp(s_0)) - o \right)^T \Theta_{\gamma}^{-1} \left( \mu_\gamma \log(1 + \exp(s_0)) - o \right) \ight) \exp \left( (s_1 - s_0)^T \Theta_{\varepsilon}^{-1} (s_1 - s_0) \right) ds_0 \]

- = intractable
Update at T=1 and later

• Update at T=1

\[ P(s_t \mid o_{0:t}) = CP(s_t \mid o_{0:t-1})P(o_t \mid s_t) \]

– Intractable

• Prediction for T=2

\[ P(s_t \mid o_{0:t-1}) = \int_{-\infty}^{\infty} P(s_{t-1} \mid o_{0:t-1})P(s_t \mid s_{t-1})ds_{t-1} \]

– Intractable
The State prediction Equation

\[ s_t = f(s_{t-1}, \zeta_t) \]

• Similar problems arise for the state prediction equation

• \( P(s_t|s_{t-1}) \) may not have a closed form

• Even if it does, it may become intractable within the prediction and update equations
  – Particularly the prediction equation, which includes an integration operation
Simplifying the problem: Linearize

• The *tangent* at any point is a good *local* approximation if the function is sufficiently smooth

\[ o = \gamma + \log(1 + \exp(s)) \]
Simplifying the problem: Linearize

- The tangent at any point is a good local approximation if the function is sufficiently smooth.

\[ o = \gamma + \log(1 + \exp(s)) \]
Simplifying the problem: Linearize

- The tangent at any point is a good local approximation if the function is sufficiently smooth.

\[ o = \gamma + \log(1 + \exp(s)) \]
Simplifying the problem: Linearize

- The tangent at any point is a good local approximation if the function is sufficiently smooth.
Linearizing the observation function

\[
P(s_t \mid o_{0:t-1}) = \text{Gaussian} (\bar{s}_t, R_t)
\]

\[
o = \gamma + g(s) \quad \Rightarrow \quad o \approx \gamma + g(\bar{s}_t) + J_g(\bar{s}_t)(s - \bar{s}_t)
\]

• Simple first-order Taylor series expansion
  – \( J() \) is the Jacobian matrix
    • Simply a determinant for scalar state

• Expansion around current predicted \emph{a priori} (or predicted) mean of the state
  – Linear approximation changes with time
Most probability is in the low-error region

\[ P(s_t \mid o_{0:t-1}) = \text{Gaussian}(\bar{s}_t, R_t) \]

- Most probability mass close to mean

- \( P(s_t) \) is small where approximation error is large
  - Most of the probability mass of \( s \) is in low-error regions
The state equation?

\[ s_t = f(s_{t-1}) + \epsilon \]

\[ P(\epsilon) = \text{Gaussian}(\epsilon; 0, \Theta_\epsilon) \]

- Again, direct use of f() can be disastrous

- Solution: Linearize

\[ P(s_{t-1} \mid o_{0:t-1}) = \text{Gaussian}(s_{t-1} \mid \hat{s}_{t-1}, \hat{R}_{t-1}) \]

\[ s_t = f(s_{t-1}) + \epsilon \quad \Rightarrow \quad s_t \approx \epsilon + f(\hat{s}_{t-1}) + J_f(\hat{s}_{t-1})(s_{t-1} - \hat{s}_{t-1}) \]

- Linearize around the mean of the updated distribution of s at t-1
  - Converts the system to a linear one
Linearized System

\[ o = \gamma + g(s) \]

\[ s_t = f(s_{t-1}) + \varepsilon \]

\[ o \approx \gamma + g(\bar{s}_t) + J_g(\bar{s}_t)(s - \bar{s}_t) \]

\[ s_t \approx \varepsilon + f(\hat{s}_{t-1}) + J_f(\hat{s}_{t-1})(s_{t-1} - \hat{s}_{t-1}) \]

• Now we have a simple time-varying linear system
• Kalman filter equations directly apply
The Extended Kalman filter

• Prediction

\[ \bar{s}_t = f(\hat{s}_{t-1}) \]

\[ R_t = \varnothing \varnothing + A_t \hat{R}_{t-1} A_t^T \]

• Update

\[ K_t = R_t B_t^T \left( B_t R_t B_t^T + \varnothing \varnothing \right)^{-1} \]

\[ \hat{s}_t = \bar{s}_t + K_t \left( o_t - g(\bar{s}_t) \right) \]

\[ \hat{R}_t = (I - K_t B_t) R_t \]

\[ s_t = f(s_{t-1}) + \varnothing \]

\[ o_t = g(s_t) + \gamma \]

\[ A_t = J_f(\hat{s}_{t-1}) \]

\[ B_t = J_g(\bar{s}_t) \]

Jacobiians used in Linearization

Assuming \( \varnothing \) and \( \gamma \) are 0 mean for simplicity
The Extended Kalman filter

• Prediction

\[ s_t = f(s_{t-1}) + \epsilon \]
\[ o_t = g(s_t) + \gamma \]
\[ A_t = I - (\hat{K}_t \hat{B}_t) \]

The predicted state at time \( t \) is obtained simply by propagating the estimated state at \( t-1 \) through the state dynamics equation.

\[ \hat{s}_t = \bar{s}_t + K_t (o_t - g(\bar{s}_t)) \]
\[ \hat{R}_t = (I - K_t \hat{B}_t)R_t \]
The Extended Kalman filter

• Prediction

\[ \hat{s}_t = f(\hat{s}_{t-1}) \]

\[ R_t = \Theta \varepsilon + A_t \hat{R}_{t-1} A_t^T \]

Uncertainty of prediction.

The variance of the predictor = variance of \( \varepsilon_t \) + variance of \( As_{t-1} \)

A is obtained by linearizing \( f() \)

\[ A_t = J_f(\hat{s}_{t-1}) \]

\[ B_t = J_g(\bar{s}_t) \]

\[ s_t = f(s_{t-1}) + \varepsilon \]

\[ o_t = g(s_t) + \varepsilon \]

\[ R_t = (I - K_t B_t)A_t \]
The Extended Kalman filter

• Prediction

\[ \bar{s}_t = f(\hat{s}_{t-1}) \]

\[ s_t = f(s_{t-1}) + \varepsilon \]

\[ o_t = g(s_t) + \varepsilon \]

\[ R_t = \Theta \varepsilon + A_t \hat{R}_{t-1} A_t^T \]

• Update

\[ K_t = R_t B_t^T \left( B_t R_t B_t^T + \Theta \gamma \right)^{-1} \]

\[ B_t = J_g(\bar{s}_t) \]

The Kalman gain is the slope of the MAP estimator that predicts \( s \) from \( o \)

\[ RBT = C_{so}, \quad (BRB^T+\Theta) = C_{oo} \]

\( B \) is obtained by linearizing \( g() \)
The Extended Kalman filter

• Prediction

\[
\hat{s}_t = f(\hat{s}_{t-1})
\]

\[
s_t = f(s_{t-1}) + \varepsilon
\]

\[
R_t = \Theta \varepsilon + A_t \hat{R}_{t-1} A_t^T
\]

We can also predict the observation from the predicted state using the observation equation

\[
\hat{o}_t = g(\hat{s}_t)
\]

\[
\hat{s}_t = \hat{s}_t + K_t (o_t - g(\hat{s}_t))
\]

\[
\hat{R}_t = (I - K_t B_t) R_t
\]

\[
\bar{o}_t = g(\bar{s}_t)
\]
The Extended Kalman filter

• Prediction

\[ s_t = f(s_{t-1}) + \varepsilon \]

\[ o_t = g(s_t) + \varepsilon \]

\[ \bar{s}_t = f(\hat{s}_{t-1}) \]

\[ R_t = \Theta \varepsilon + A_t \hat{R}_{t-1} A_t^T \]

We must correct the predicted value of the state after making an observation

\[ \hat{s}_t = \bar{s}_t + K_t (o_t - g(\bar{s}_t)) \]

\[ \bar{o}_t = g(\bar{s}_t) \]

The correction is the difference between the actual observation and the predicted observation, scaled by the Kalman Gain
The Extended Kalman filter

- Prediction

\[
\hat{s}_t = f(\hat{s}_{t-1})
\]

\[
s_t = f(s_{t-1}) + \epsilon
\]

\[
o_t = g(s_t) + \epsilon
\]

\[
R_t = \Theta \varepsilon + A_t \hat{R}_{t-1} A_t^T
\]

\[
B_t = J_g(\hat{s}_t)
\]

The uncertainty in state decreases if we observe the data and make a correction.

The reduction is a multiplicative "shrinkage" based on Kalman gain and \(B\).

\[
\hat{R}_t = (I - K_t B_t) R_t
\]
The Extended Kalman filter

• Prediction

\[ \hat{s}_t = f(\hat{s}_{t-1}) \]

[Equation]

\[ R_t = \Theta \varepsilon + A_t \hat{R}_{t-1} A_t^T \]

[Equation]

\[ s_t = f(s_{t-1}) + \varepsilon \]

[Equation]

• Update

\[ K_t = R_t B_t^T \left( B_t R_t B_t^T + \Theta \gamma \right)^{-1} \]

[Equation]

\[ \hat{s}_t = \bar{s}_t + K_t \left( o_t - g(\bar{s}_t) \right) \]

[Equation]

\[ \hat{R}_t = (I - K_t B_t) R_t \]

[Equation]
EKFs

• EKFs are probably the most commonly used algorithm for tracking and prediction
  – Most systems are non-linear
  – Specifically, the relationship between state and observation is usually nonlinear
  – The approach can be extended to include non-linear functions of noise as well

• The term “Kalman filter” often simply refers to an extended Kalman filter in most contexts.

• But..
EKFs have limitations

- If the non-linearity changes too quickly with s, the linear approximation is invalid
  - Unstable
- The estimate is often biased
  - The true function lies entirely on one side of the approximation

- Various extensions have been proposed:
  - Invariant extended Kalman filters (IEKF)
  - Unscented Kalman filters (UKF)
Conclusions

• HMMs are predictive models
• Continuous-state models are simple extensions of HMMs
  – Same math applies
• Prediction of linear, Gaussian systems can be performed by Kalman filtering
• Prediction of non-linear, Gaussian systems can be performed by Extended Kalman filtering