Problem: Growing Memory Disaster:

\[ m \text{ data, } N \text{ hypotheses } \rightarrow N^{m+1} \text{ continuations} \]

radical solution: mono-hypothesis approximation

- **gating:** Exclude competing data with \( \| \nu^i_k|k-1 \| > \lambda \)!
  
  \[ \text{KALMAN filter (KF)} \]

  + very simple, – \( \lambda \) too small: loss of target measurement

- Force a **unique interpretation** in case of a conflict!
  
  look for *smallest statistical distance*: \( \min_i \| \nu^i_k|k-1 \| \)
  
  \[ \text{Nearest-Neighbor filter (NN)} \]

  + one hypothesis, – hard decision, – not adaptive

- **global combining:** Merge all hypotheses!
  
  \[ \text{PDAF, JPDAF filter} \]

  + all data, + adaptive, – reduced applicability
**Moment Matching**: Approximate an arbitrary pdf $p(x)$ with $\mathbb{E}[x] = x$, $\mathbb{C}[x] = P$ by $p(x) \approx \mathcal{N}(x; x, P)$!

here especially: $p(x) = \sum_H p_H \mathcal{N}(x; x_H, P_H)$ (normal mixtures)

\[
x = \sum_H p_H x_H
\]

\[
P = \sum_H p_H \left\{ P_H + (x_H - x)(x_H - x)^\top \right\}
\]
PDAF Filter: formally analogous to Kalman Filter

Filtering (scan $k-1$):
\[ p(x_{k-1} | Z^{k-1}) = \mathcal{N}(x_{k-1}; x_{k-1|k-1}, P_{k-1|k-1}) \quad (\rightarrow \text{initiation}) \]

Prediction (scan $k$):
\[ p(x_k | Z^{k-1}) \approx \mathcal{N}(x_k; x_{k-1|k-1}, P_{k|k-1}) \quad (\text{like Kalman}) \]

Filtering (scan $k$):
\[ p(x_k | Z^k) \approx \sum_{j=0}^{m_k} p_j^* \mathcal{N}(x_k; x_{k|k}^j, P_{k|k}^j) \]

\[
x_{k|k}^j = \begin{cases} x_{k|k-1} & j = 0 \\ x_{k|k-1} + W_k \nu_k^j & j \neq 0 \end{cases}
\]

\[
\nu_k^j = z_k^j - Hx_k, \quad W_k = P_{k|k-1} H^T S_k^{-1}, \quad S_k = HP_{k|k-1} H^T + R_k
\]

\[
p_j^* = \frac{p_{k}^{*}}{\sum_j p_j^{*}}, \quad p_{k}^{*} = \begin{cases} (1 - P_D) \rho_F & j = 0 \\ \frac{P_D}{\sqrt{2\pi S_{hk}}} e^{-\frac{1}{2} \nu_{hk}^T S_{hk}^{-1} \nu_{hk}} & j \neq 0 \end{cases}
\]
**PDAF Filter: formally analog to Kalman Filter**

**Filtering (scan $k - 1$):** 
\[ p(x_{k-1} | Z^{k-1}) = \mathcal{N}(x_{k-1}; x_{k-1|k-1}, P_{k-1|k-1}) \]  
($\rightarrow$ initiation)

**prediction (scan $k$):** 
\[ p(x_k | Z^{k-1}) \approx \mathcal{N}(x_k; x_{k|k-1}, P_{k|k-1}) \]  
(like Kalman)

**Filtering (scan $k$):** 
\[ p(x_k | Z^k) \approx \sum_{j=0}^{m_k} p^j_k \mathcal{N}(x_k; x^j_{k|k}, P^j_{k|k}) \approx \mathcal{N}(x_k; x_{k|k}, P_{k|k}) \]

\[
\nu_k = \sum_{j=0}^{m_k} p^j_k \nu_k^j, \quad \nu_k^j = z_k^j - H x_{k|k-1} \quad \text{combined innovation}
\]

\[
W_k = P_{k|k-1} H^T S_k^{-1}, \quad S_k = H P_{k|k-1} H^T + R_k \quad \text{Kalman gain matrix}
\]

\[
p^j_k = p^*_k / \sum_j p^*_k, \quad p^*_k = \begin{cases} 
(1 - P_D) \rho_F 
\sqrt{2\pi S_{n_k} \nu_{n_k}} \exp \left( -\frac{1}{2} \nu_{n_k}^T S_{n_k} \nu_{n_k} \right)
\end{cases}
\]

weighting factors

\[
x_k = x_{k|k-1} + W_k \nu_k \quad \text{(Filtering Update: Kalman)}
\]

\[
P_k = P_{k|k-1} - (1-P^0_k) W_k S W_k^T \quad \text{(Kalman part)}
\]

\[+ W_k \left\{ \sum_{j=0}^{m_k} p^j_k \nu_k^j \nu_k^j - \nu_k \nu_k^T \right\} W_k^T \quad \text{(Spread of Innovations)}
\]
The qualitative shape of $p(x_k | \mathcal{Z}_k)$ is often much simpler than its correct representation: *a few pronounced modes*.

**adaptive solution: nearly optimal approximation**
The qualitative shape of \( p(x_k | Z^k) \) is often much simpler than its correct representation: a few pronounced modes

**adaptive solution: nearly optimal approximation**

- **individual gating**: Exclude *irrelevant data!*
  
  before continuing existing track hypotheses \( H_{k-1} \)
  
  \( \rightarrow \) **limiting case**: KALMAN filter (KF)
The qualitative shape of $p(x_k | \mathcal{Z}_k)$ is often much simpler than its correct representation: *a few pronounced modes*

**adaptive solution: nearly optimal approximation**

- **individual gating**: Exclude *irrelevant data*!  
  before continuing existing track hypotheses $H_{k-1}$  
  $\rightarrow$ *limiting case*: KALMAN filter (KF)

- **pruning**: Kill hypotheses of very *small weight*!  
  after calculating the weights $p_{H_k}$, before filtering  
  $\rightarrow$ *limiting case*: Nearest Neighbor filter (NN)
The qualitative shape of $p(x_k|\mathcal{Z}_k^k)$ is often much simpler than its correct representation: *a few pronounced modes*

**adaptive solution: nearly optimal approximation**

- **individual gating:** Exclude *irrelevant data!* before continuing existing track hypotheses $H_{k-1}$
  
  → *limiting case:* KALMAN filter (KF)

- **pruning:** Kill hypotheses of very *small weight!* after calculating the weights $p_{H_k}$, before filtering
  
  → *limiting case:* Nearest Neighbor filter (NN)

- **local combining:** Merge *similar hypotheses!* after the complete calculation of the pdfs
  
  → *limiting case:* PDAF (global combining)
Successive Local Combining

Partial sums of similar densities → moment matching:

$$\sum_{H_k \in \mathcal{H}^k} p_{H_k} \mathcal{N}(x_k; x_{H_k}, P_{H_k}) \approx p_{H_k^*} \mathcal{N}(x_k; x_{H_k^*}, P_{H_k^*})$$

$$\mathcal{H}^k \subset \mathcal{H}^k \rightarrow H_k^*: \text{effective hypothesis}$$
Successive Local Combining

Partial sums of similar densities → moment matching:

\[
\sum_{H_k \in \mathcal{H}^k^*} p_{H_k} \mathcal{N}(x_k; x_{H_k}, P_{H_k}) \approx p_{H_k^*} \mathcal{N}(x_k; x_{H_k^*}, P_{H_k^*})
\]

\(\mathcal{H}^k^* \subset \mathcal{H}^k \rightarrow H_k^*: \text{effective hypothesis}\)

similarity: \(d(H_1, H_2) < \mu\) mit (z.B.):

\[
d(H_1, H_2) = (x_{H_1} - x_{H_2})^\top (P_{H_1} + P_{H_2})^{-1} (x_{H_1} - x_{H_2})
\]

Start: Hypothesis of highest weight \(H_1\) → search similar hypothesis \((p_H \backslash \mu)\) → merge: \((H_1, H) > H_1^*\) → continue search \((p_H \backslash \mu)\) ...

→ restart: hypothesis with next to highest weight \(H_2\) → ...
Successive Local Combining

Partial sums of similar densities $\rightarrow$ moment matching:

$$\sum_{H_k \in \mathcal{H}^k} p_{H_k} \mathcal{N}(x_k; x_{H_k}, P_{H_k}) \approx p_{H_k}^* \mathcal{N}(x_k; x_{H_k}^*, P_{H_k}^*)$$

$\mathcal{H}^k \subset \mathcal{H}^k \rightarrow H_k^*: \text{effective hypothesis}$

similarity: $d(H_1, H_2) < \mu \quad \text{mit (z.B.):}$

$$d(H_1, H_2) = (x_{H_1} - x_{H_2})^\top (P_{H_1} + P_{H_2})^{-1} (x_{H_1} - x_{H_2})$$

Start: Hypothesis of highest weight $H_1 \rightarrow$ search similar hypothesis $(p_{H \backslash \chi}) \rightarrow$ merge: $(H_1, H) > H_1^* \rightarrow$ continue search $(p_{H \backslash \chi}) \ldots$

$\rightarrow$ restart: hypothesis with next to highest weight $H_2 \rightarrow \ldots$

- In many cases: good approximations $\rightarrow$ quasi-optimality
- PDAF, JPDAF: $\mathcal{H}^k = \mathcal{H}^k \rightarrow$ limited applicability
- robustness $\rightarrow$ detail mostly irrelevant

Sensor Data Fusion - Methods and Applications, 8th Lecture on December 14, 2016
Recapitulation: Detection Process for Sensors

**Detector:** receives signals and decides on object existence

**Processor:** processes detected signals and produces measurements

‘\( D \)’: detector detects an object

\( D \): object actually existent

error of 1. kind: \( P_1 = P(\neg 'D' | D) \)

error of 2. kind: \( P_{II} = P('D' | \neg D) \)

measure of detection performance: \( P_D = P('D' | D) \)

detector properties characterized by two parameters:

– detection probability \( P_D = 1 - P_1 \)

– false alarm probability \( P_F = P_{II} \)

example (Swerling I model): \( P_D = P_D(P_F, \text{SNR}) = P_F^{1/(1+\text{SNR})} \)

**detector design:** Maximize detection probability \( P_D \) for a given, predefined false alarm probability \( P_F \).
Track Extraction: Initiation of the PDF Iteration

**extraction of target tracks:** detection on a higher level of abstraction

**start:** data sets $Z_k = \{z_{jk}\}_{j=1}^{m_k}$ (sensor performance: $P_D$, $\rho_F$, $R$)

**goal:** Detect a target trajectory in a time series: $Z^k = \{Z_i\}_{i=1}^k$!

at first simplifying assumptions:

- The targets in the sensors’ field of view (FoV) are well-separated.
- The sensor data in the FoV in scan $i$ are produced simultaneously.
Track Extraction: Initiation of the PDF Iteration

**extraction of target tracks:** detection on a higher level of abstraction

**start:** data sets \( Z_k = \{ z_j^k \}_{j=1}^{m_k} \) (sensor performance: \( P_D, \rho_F, R \))

**goal:** Detect a target trajectory in a time series: \( \mathcal{Z}^k = \{ Z_i^k \}_{i=1}^k \)

at first simplifying assumptions:

- The targets in the sensors’ field of view (FoV) are well-separated.
- The sensor data in the FoV in scan \( i \) are produced simultaneously.

decision between two competing hypotheses:

- \( h_1 \): Besides false returns \( \mathcal{Z}^k \) contains also target measurements.
- \( h_0 \): There is no target existing in the FoV; all data in \( \mathcal{Z}^k \) are false.

statistical decision errors:

\[
P_1 = \text{Prob}(\text{accept } h_1 | h_1) \quad \text{analogous to the sensors’ } P_D
\]
\[
P_0 = \text{Prob}(\text{accept } h_1 | h_0) \quad \text{analogous to the sensors’ } P_F
\]
Practical Approach: Sequential Likelihood Ratio Test

Goal: Decide as fast as possible for given decision errors $P_0$, $P_1$.

Consider the ratio of the conditional probabilities $p(h_1|Z^k)$, $p(h_0|Z^k)$ and the likelihood ratio $LR(k) = p(Z^k|h_1)/p(Z^k|h_0)$ as an intuitive decision function:

$$\frac{p(h_1|Z^k)}{p(h_0|Z^k)} = \frac{p(Z^k|h_1)}{p(Z^k|h_0)} \frac{p(h_1)}{p(h_0)}$$

a priori: $p(h_1) = p(h_0)$
Practical Approach: Sequential Likelihood Ratio Test

**Goal:** Decide as fast as possible for given decision errors $P_0, P_1$!

Consider the ratio of the conditional probabilities $p(h_1|Z^k)$, $p(h_0|Z^k)$ and the likelihood ratio $LR(k) = p(Z^k|h_1)/p(Z^k|h_0)$ as an intuitive decision function:

\[
\frac{p(h_1|Z^k)}{p(h_0|Z^k)} = \frac{p(Z^k|h_1) p(h_1)}{p(Z^k|h_0) p(h_0)} \quad \text{a priori: } p(h_1) = p(h_0)
\]

Starting from a time window with length $k = 1$, calculate the test function $LR(k)$ successively and compare it with two thresholds $A$, $B$:

- If $LR(k) < A$, accept hypothesis $h_0$ (i.e. no target is existing)!
- If $LR(k) > B$, accept hypothesis $h_1$ (i.e. target exists in FoV)!
- If $A < LR(k) < B$, wait for new data $Z_{k+1}$, repeat with $LR(k+1)$!
Iterative Calculation of the Likelihood Ratio

\[ LR(k) = \frac{p(Z^k|h_1)}{p(Z^k|h_0)} = \int dx_k \frac{p(Z_k, m_k, x_k, Z^{k-1}|h_1)}{p(Z_k, m_k, Z^{k-1}, h_0)} \]
Iterative Calculation of the Likelihood Ratio

\[ \text{LR}(k) = \frac{p(Z^k|h_1)}{p(Z^k|h_0)} = \frac{\int dx_k p(Z_k, m_k, x_k, Z^{k-1}|h_1)}{p(Z_k, m_k, Z^{k-1}, h_0)} = \frac{\int dx_k p(Z_k, m_k|x_k) p(x_k|Z^{k-1}, h_1) p(Z^{k-1}|h_1)}{|\text{FoV}|^{-m_k} p_F(m_k) p(Z^{k-1}|h_0)} \]
Iterative Calculation of the Likelihood Ratio

\[ LR(k) = \frac{p(Z^k|h_1)}{p(Z^k|h_0)} = \frac{\int dx_k p(Z_k, m_k, x_k, Z^{k-1}|h_1) p(Z_k, m_k, Z^{k-1}, h_0)}{p(Z_k, m_k, Z^{k-1}, h_0)} = \frac{\int dx_k p(Z_k, m_k|x_k) p(x_k|Z^{k-1}, h_1) p(Z^{k-1}|h_1)}{|\text{FoV}|^{-m_k} p_F(m_k) p(Z^{k-1}|h_0)} \]

\[ = \frac{\int dx_k p(Z_k, m_k|x_k, h_1) p(x_k|Z^{k-1}, h_1)}{|\text{FoV}|^{-m_k} p_F(m_k)} LR(k - 1) \]

basic idea: iterative calculation!
Iterative Calculation of the Likelihood Ratio

\[
\text{LR}(k) = \frac{p(Z^k|h_1)}{p(Z^k|h_0)} = \frac{\int dx_k \, p(Z_k, m_k, x_k, Z^{k-1}|h_1)}{p(Z_k, m_k, Z^{k-1}, h_0)} = \frac{\int dx_k \, p(Z_k, m_k|x_k) \, p(x_k|Z^{k-1}, h_1) \, p(Z^{k-1}|h_1)}{|\text{FoV}|^{-m_k} \, p_F(m_k) \, p(Z^{k-1}|h_0)} \]

\[
= \frac{\int dx_k \, p(Z_k, m_k|x_k, h_1) \, p(x_k|Z^{k-1}, h_1)}{|\text{FoV}|^{-m_k} \, p_F(m_k)} \, \text{LR}(k - 1)
\]

basic idea: iterative calculation!

Let \( H_k = \{E_k, H_{k-1}\} \) be an interpretation history of the time series \( Z^k = \{Z_k, Z^{k-1}\} \).

\( E_k = E^0_k \): target was not detected, \( E_k = E^j_k \): \( z^j_k \in Z_k \) is a target measurement.

\[ p(x_k|Z^{k-1}, h_1) = \sum_{H_{k-1}} p(x_k|H_{k-1}Z^{k-1}, h_1) \, p(H_{k-1}|Z^{k-1}, h_1) \quad \text{The standard MHT prediction!} \]

\[ p(Z_k, m_k|x_k, h_1, h_1) = \sum_{E_k} p(Z_k, E_k|x_k, h_1) \quad \text{The standard MHT likelihood function!} \]

The calculation of the likelihood ratio is just a by-product of Bayesian MHT tracking.
Iteration Formula for $\text{LR}(k) = p(\mathcal{Z}^k| h_1)/p(\mathcal{Z}^k| h_0)$

initiation: \begin{align*}
k &= 0, & j_0 &= 0, & \lambda_{j_0} &= 1
\end{align*}

recursion: $\text{LR}(k + 1) = \sum_{j_{k+1}} \lambda_{j_{k+1}} = \sum_{j_{k+1}=0}^{m_k+1} \sum_{j_k} \lambda_{j_{k+1}j_k} \lambda_{j_k}$

with: $\lambda_{j_{k+1}j_k} = \begin{cases} 1 - P_D & \text{for } j_{k+1} = 0 \\
\frac{P_D}{\rho_F} \mathcal{N}(\mathbf{v}_{j_{k+1}j_k}, \mathbf{S}_{j_{k+1}j_k}) & \text{for } j_{k+1} \neq 0 \end{cases}$

convenient notation: \begin{align*}
\text{with } j_k = (j_k, \ldots, j_1) \text{ let } \sum_{j_k} \lambda_{j_k} = \sum_{j_k=0}^{m_k} \ldots \sum_{j_1=0}^{m_1} \lambda_{j_k \ldots j_1}
\end{align*}
Iteration Formula for $LR(k) = \frac{p(Z^k|h_1)}{p(Z^k|h_0)}$

**initiation:**  
$k = 0, \ j_0 = 0, \ \lambda_{j_0} = 1$

**recursion:**  
$LR(k + 1) = \sum_{j_{k+1}} \lambda_{j_{k+1}} = \sum_{j_{k+1}=0}^{m_{k+1}} \sum_{i_k} \lambda_{j_{k+1}i_k} \lambda_{j_k}$

with:  
$
\lambda_{j_{k+1}i_k} = \begin{cases} 
1 - P_D & \text{for } j_{k+1} = 0 \\
\frac{P_D}{\rho_F} \mathcal{N}(\nu_{j_{k+1}i_k}, S_{j_{k+1}i_k}) & \text{for } j_{k+1} \neq 0 
\end{cases}
$

**innovation:**  
$\nu_{j_{k+1}i_k} = z_{j_{k+1}} - H_{j_{k+1}} x_{j_{k+1}|k}$

**innov. cov.:**  
$S_{j_{k+1}i_k} = H_{j_{k+1}} P_{j_{k+1}|k} H_{j_{k+1}}^\top + R_{j_{k+1}}$

**state update:**  
$x_{j_{k+1}|k} = F_{j_{k+1}} x_{j_k}$  
$x_{j_k} = x_{j_{k}\mid k-1} + W_{j_k|k-1} \nu_{j_k, j_{k-1}}$

**covariances:**  
$P_{j_{k+1}|k} = F_{j_{k+1}} P_{j_k} F_{j_{k+1}}^\top + D_{j_{k+1}}$  
$P_{j_k} = P_{j_{k}\mid k-1} - W_{j_{k}\mid k-1} S_{j_k, j_{k-1}} W_{j_k, j_{k-1}}^\top$
**Iteration Formula for** $\text{LR}(k) = p(Z^k|h_1)/p(Z^k|h_0)$

**initiation:**

$k = 0, \quad j_0 = 0, \quad \lambda_{j_0} = 1$

**recursion:**

$\text{LR}(k + 1) = \sum_{j_{k+1}} \lambda_{j_{k+1}} = \sum_{j_{k+1}=0}^{m_{k+1}} \sum_{j_k} \lambda_{j_{k+1}j_k} \lambda_{j_k}$

with:

$\lambda_{j_{k+1}j_k} = \begin{cases} 1 - P_D & \text{for } j_{k+1} = 0 \\ \frac{P_D}{\rho_F} \mathcal{N}(\nu_{j_{k+1}j_k}, S_{j_{k+1}j_k}) & \text{for } j_{k+1} \neq 0 \end{cases}$

**innovation:**

$\nu_{j_{k+1}j_k} = z_{j_{k+1}} - H_{j_{k+1}} x_{j_{k+1}|k}$

**innov. cov.:**

$S_{j_{k+1}j_k} = H_{j_{k+1}} P_{j_{k+1}|k} H_{j_{k+1}}^T + R_{j_{k+1}}$

**state update:**

$x_{j_{k+1}|k} = F_{j_{k+1}} x_{j_k} \quad \quad x_{j_k} = x_{j_{k-1}|k} + W_{j_{k-1}} \nu_{j_{k-1}}$

**covariances:**

$P_{j_{k+1}|k} = F_{j_{k+1}} P_{j_k} F_{j_{k+1}}^T + D_{j_{k+1}} \quad P_{j_k} = P_{j_{k-1}|k} - W_{j_{k-1}} j_{k-1} S_{j_{k-1}|k-1} W_{j_{k-1}}^T$

**Exercise 8.1** Show that this recursion formulae for calculating the decision function is true.
Sequential Track Extraction: Discussion

- $LR(k)$ is given by a growing number of summands, each related to a particular interpretation history. The tuple $\{\lambda_{jk}, \mathbf{x}_{jk}, \mathbf{P}_{jk}\}$ is called a sub-track.
Sequential Track Extraction: Discussion

- \( LR(k) \) is given by a growing number of summands, each related to a particular interpretation history. The tuple \( \{ \lambda_{jk}, x_{jk}P_{jk} \} \) is called a *sub-track*.

- For mitigating growing memory problems all approximations discussed for track maintenance can be used if they do not significantly affect \( LR(k) \):
  - *individual gating*: Exclude data not likely to be associated.
  - *pruning*: Kill sub-tacks contributing marginally to the test function.
  - *local combining*: Merge similar sub tracks:
    \[
    \{ \lambda_i, x_i, P_i \}_i \rightarrow \{ \lambda, x, P \} \quad \text{with:} \quad \lambda = \sum_i \lambda_i, \\
    x = \frac{1}{\lambda} \sum_i \lambda_i x_i, \quad P = \frac{1}{\lambda} \sum_i \lambda_i [P_i + (x_i - x)(\ldots)^\top].
    \]
Sequential Track Extraction: Discussion

- $\text{LR}(k)$ is given by a growing number of summands, each related to a particular interpretation history. The tuple $\{\lambda_{jk}, x_{jk}, P_{jk}\}$ is called a sub-track.

- For mitigating growing memory problems all approximations discussed for track maintenance can be used if they do not significantly affect $\text{LR}(k)$:
  - **individual gating**: Exclude data not likely to be associated.
  - **pruning**: Kill sub-tacks contributing marginally to the test function.
  - **local combining**: Merge similar sub tracks:
    
    $$\{\lambda_i, x_i, P_i\}_i \rightarrow \{\lambda, x, P\} \quad \text{with:} \quad \lambda = \sum_i \lambda_i, \quad x = \frac{1}{\lambda} \sum_i \lambda_i x_i, \quad P = \frac{1}{\lambda} \sum_i \lambda_i [P_i + (x_i - x)(\ldots)^\top].$$

- The LR test ends with a decision in favor of or against the hypotheses: $h_0$ (no target) or $h_1$ (target existing). Intuitive interpretation of the thresholds!
track extraction at $t_k$: Decide in favor of $h_1$!

initiation of pdf iteration (track maintenance):

Normalize coefficients $\lambda_{jk}$:

$$p_{jk} = \frac{\lambda_{jk}}{\sum_{j_k} \lambda_{jk}}!$$

$$(\lambda_{jk}, x_{jk}, P_{jk}) \rightarrow p(x_k|Z^k) = \sum_{j_k} p_{jk} \mathcal{N}(x_k; x_{jk}, P_{jk})$$

Continue track extraction with the remaining sensor data!

sequential LR test for track monitoring:

After deciding in favor of $h_1$ reset $LR(0) = 1$! Calculate $LR(k)$ from $p(x_k|Z^k)$!

track confirmation: $LR(k) > \frac{P_1}{P_0}$: reset $LR(0) = 1$!

track deletion: $LR(k) < \frac{1-P_1}{1-P_0}$; ev. track re-initiation
DEMONSTRATION (simulated)
DEMOnSTRATION (simulated)

Exercise 8.2 (voluntary)

Simulate a detection process with a given $P_D$, target measurements with a given $R$, a detection process with a given $P_D$ and realize the track extraction procedure.
Generalization to Target Cluster (Perfect Resolution)

Scheme directly extendable to clusters consisting of \( n \) targets, \textit{if} \( n \) is known!

\textbf{principal approach in case of unknown} \( n \):

1. Start with sensor measurements \( Z_1 \).

2. Assume for a target cluster \( n \leq N \)! A-priorily: \( P(n) = \frac{1}{N} \)

3. hypothesis \( h_n \): there exist \( n \) targets; the data set \( Z_1 \) contains at least one target measurement; \( h_0 \): no target existing at all

4. Consider the following ratio (at least 1, at most \( N \) targets):

\[
\frac{p(h_1 \lor \ldots \lor h_N|Z^k)}{p(h_0|Z^k)} = \frac{\sum_{n=1}^{N} p(h_n|Z^k)}{p(h_0|Z^k)} = \frac{\sum_{n=1}^{N} \frac{p(Z^k|h_n) \cdot p(h_n)}{p(h_0)}}{p(h_0)}
\]
Generalization to Target Cluster (Perfect Resolution)

Scheme directly extendable to clusters consisting of $n$ targets, if $n$ is known!

principal approach in case of unknown $n$:

1. Start with sensor measurements $Z_1$.

2. Assume for a target cluster $n \leq N$! A-priori: $P(n) = \frac{1}{N}$

3. hypothesis $h_n$: there exist $n$ targets; the data set $Z_1$ contains at least one target measurement; $h_0$: no target existing at all

4. generalized LR test function: $LR(k) = \frac{1}{N} \sum_{n=1}^{N} \frac{p(Z^k|h_n)}{p(Z^k|h_0)}$

5. Calculate $LR_n(k) = p(Z^k|h_n)/p(Z^k|h_0)$ in analogy to $n = 1$.

6. ‘Cardinality’ of having $n$ objects in the cluster: $c_k(n) = \frac{LR_n(k)}{\sum_{n=1}^{N} LR_n(k)}$
DEMONSTRATION (simulated)

**Moment Matching:** Approximate an arbitrary pdf $p(x)$ with $\mathbb{E}[x] = x$, $\mathbb{C}[x] = P$ by $p(x) \approx \mathcal{N}(x; x, P)$!

Here especially: $p(x) = \sum_i p_i \mathcal{N}(x; x_i, P_i)$ (GAUSSian mixtures)

\[ x = \sum_i p_i x_i \]

**Exercise** Show:

\[ P = \sum_i p_i \left\{ P_i + (x_i - x)(x_i - x)^\top \right\} \]
2nd Order Approximation:

\[ p(x) = \sum_i p_H \mathcal{N}(x; x_i, P_i) \approx \mathcal{N}(x; \mathbb{E}_p[x], C_p[x]) \]

\[ \mathbb{E}_p[x] = \int dx \ x p(x) = \sum_i p_i \int dx \ x \mathcal{N}(x; x_i, P_i) = \sum_i p_i x_i =: x \]

\[ C_p[x] = \int dx \ p(x) (x - \mathbb{E}_p[x])(x - \mathbb{E}_p[x])^\top = \sum_i p_i \int dx \ (x - x)(x - x)^\top \mathcal{N}(x; x_i, P_i) \]

\[ = \sum_i p_i \int dx \ \{ (x - x)(x - x)^\top - 2(x - x_i)(x_i - x)^\top \} \mathcal{N}(x; x_i, P_i) \]

since we have: \[ \int dx \ (x - x_i)(x_i - x)^\top \mathcal{N}(x; x_i, P_i) = 0 \]

\[ = \sum_i p_i \int dx \ \{ x x^\top - 2x x_i^\top + x_i x_i^\top + x_i x_i^\top - 2x_i x_i^\top + x x^\top \} \mathcal{N}(x; x_i, P_i) \]

\[ = \sum_i p_i \int dx \ \{ (x - x_i)(x - x_i)^\top + (x_i - x)(x_i - x)^\top \} \mathcal{N}(x; x_i, P_i) \]

\[ = \sum_i p_i \{ P_i + (x_i - x)(x_i - x)^\top \} = P \]