Task 1)

$\Pr(\delta|x) = P_D(x)$, $p(x)$: state-distribution

$p(x|\delta) \neq p(x)$?

$$p(x|\delta) = \frac{\Pr(\delta|x)\, p(x)}{\Pr(\delta)}$$

$$= \frac{p(\delta|x)\, p(x)}{\int P_D(x) p(x)\, dx}$$

$$= \frac{P_D(x)\, p(x)}{\int P_D(x) p(x)\, dx}$$

Synse om radar detektion alltid av hvor radar er i regionen. nearme vs langt unna ~~båten~~.

If $P_D(x)$ is a constant & not a function of x, then (setter $P_D(x) = P_D$)

$$= \frac{\cancel{P_D}\, p(x)}{\cancel{P_D} \underbrace{\int p(x)\, dx}_{=1}} = \underline{p(x)}$$

← if $P_D(x)$ is a function of x, then this does not hold.

Task2)

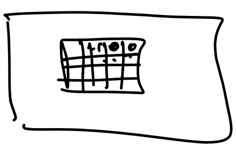
Sensor - N cells.

↓

same volume & same P_{FA}.

Wikipedia : $X_1, \dots, X_n$ i.i.d. RV, all Bernoulli trials with success prob. p, then their sum is distributed Binomically

a)



$\sim B(\varphi, P_{FA})$

$FA \sim \underbrace{\text{Bernoulli}(P_{FA})}$

$\uparrow$ false alarm

$\underline{\mu(\varphi) = B(\varphi, P_{FA})}$

b) $\lambda V_k = M_k P_{FA}$

Posterior event probabilities

Theorem 7.3.1 - Association probabilities for single-target tracking

$$\Pr\{a_k | z_{1:k}\} \propto \begin{cases} (1-P_D) m_k \dfrac{\mu(m_k)}{\mu(m_k - 1)} & \text{if } a_k = 0 \\ \dfrac{P_D}{c(z_k^{a_k})} l^{a_k} & \text{if } a_k > 0 \end{cases}$$

↓

$$l^{a_k} = \int f_z(z_k^{a_k} | x_k)\, p_{k|k-1}(x_k)\, dx_k$$

Case $a_k = 0$:

sett inn for poisson

↓

$$(1-P_D) m_k\, \mu(m_k) = (1-P_D) \cancel{m_k}\, \cancel{e^{-\lambda V_k}}\, (\lambda V)^{m_k} \quad \cancel{(m_k - 1)!}$$

$$\overline{\mu(m_k-1)} \qquad = (1-P_D)\lambda V_k \qquad \cancel{m_k!} \qquad (\lambda V_k)^{(m_k-1)} \cancel{e^{-\lambda V_k}}$$

Case $a_k > 0$:

$$\frac{P_D}{c(z_k^{a_k})}\,\ell^{a_k} = \underbrace{\frac{P_D}{C(z_k^{a_k})}}_{\frac{1}{V_k}\ (7.14)} \int \underbrace{f_z(z_k^{a_k}|x_k)}_{\mathcal{N}(z_k^{a_k};\,Hx_k,R)}\ \underbrace{p_{k|k-1}(x_k)}_{\mathcal{N}(x_k;\,\hat{x}_{k|k-1},P_{k|k-1})}\,dx_k \qquad (7.15)$$

$$= P_D V_k \int \mathcal{N}(z_k^{a_k};Hx_k,R)\,\mathcal{N}(x_k;\hat{x}_{k|k-1},P_{k|k-1})\,dx_k$$

$\downarrow$ product-rule

$$(7.17) \quad = P_D V_k\, \mathcal{N}(z_k^{a_k};\hat{z}_{k|k-1},S_k), \quad \hat{z}_{k|k-1} = H\hat{x}_{k|k-1}$$

$$S_k = HP_{k|k-1}H^T + R$$

$\Downarrow$

$$\Pr\{a_k|z_{1:k}\} \propto \begin{cases} (1-P_D)\lambda \cancel{V_k}, & \text{if } a_k = 0 \\ P_D \cancel{V_k}\, \mathcal{N}(z_k^{a_k};\hat{z}_{k|k-1},S_k) & \text{if } a_k > 0 \end{cases}$$

$\square$

Corollary 7.3.3

c) Setting in the binomial distribution in theorem 7.3.1 we get:

$$\Pr\{a_k|z_{1:k}\} \propto \begin{cases} \frac{1}{V_k}(1-P_D)m_k \frac{\mu(m_k)}{\mu(m_k-1)} & \text{if } a_k=0 \\ \frac{1}{\cancel{V_k}}\cancel{V_k} P_D \mathcal{N}(z_k^{a_k};\hat{z}_{k|k-1},S_k) & \text{if } a_k>0 \end{cases}$$

Divide by V_k to simplify

Since the expression for $a_k>0$ is the same as in the problem, which I deducted in b), I will only deduct for $a_k=0$ here:

$a_k = 0$.

$$\Pr\{a_k|z_{1:k}\} \propto \frac{(1-P_D)}{V_k}\, m_k\, \frac{\mu(m_k)}{\mu(m_k-1)}$$

$$= \left(\frac{1-P_D}{V_k}\right) \cancel{m_k} \frac{\left(\frac{\cancel{M_k!}}{\cancel{m_k!}(M_k-m_k)!}\right) P_{FA}^{m_k}(1-P_{FA})^{M_k-m_k}}{\left(\frac{\cancel{M_k!}}{(m_k-1)!(M_k+1-m_k)!}\right) P_{FA}^{m_k-1}(1-P_{FA})^{M_k+1-m_k}}$$

$$= \left(\frac{1-P_D}{}\right) \quad (M_k+1-m_k)\, P_{FA}$$

$$= \left(\frac{1-P_D}{V_k}\right) M_k \left(1 - \frac{M_k - 1}{M_k}\right) \frac{P_{FA}}{1-P_{FA}}$$

$\square$

d) for $a_k > 0$ they correspond as this is shown in b).

For $a_k = 0$:

True distribution

$$Pr\{a_k | Z_{1:k}\} \propto (1-P_D) \frac{P_{FA} M_k}{V_k} \frac{1 - \frac{m_k - 1}{M_k}}{(1-P_{FA})} = \frac{(1-P_D)\lambda}{V_k} \frac{1 - \frac{m_k-1}{M_k}}{(1-P_{FA})}$$

& from Corollary 7.3.3

$$= (1-P_D)\lambda$$

We see that (*) expression approximates to $(1-P_D)\lambda$ for Large M_k and small P_{FA}. $\frac{(1-P_D)\lambda}{V_k} \frac{1-0}{1-0}$

Conclusion is that poisson is a good approximation for Large M_k & small P_{FA}.

Task 3: IPDA vs PDAF

a) - Why posterior becomes a mixture in single target tracking?

$p(x_k) = p(x_k | Z_{1:k})$

We recieve a set of measurements $Z_k = \{z_k^1 \ldots z_k^{m_k}\}$ at each timestep. Based on assumptions that only one target exists, and the concept of association variable a_k, which tells the # of measurements originating from the target.

$a_k = 0$, no measurement from target
$a_k = 1$, measurement 1 from target
$a_k = 2$, — " — 2 — " —
⋮

The data association problem can be understood as making inference about a_k.
Posterior expressed using total prob. theorem:

$$p(x_k) = \sum_{i=0}^{m_k} \underbrace{p(x_k | a_k = i, Z_{1:k})}_{\text{comp}} \underbrace{Pr\{a_k = i | Z_{1:k}\}}_{\text{weight}}$$

This has the same form as those encountered in section 6 $\left(p(x_k|z_{1:k}) = \sum_{S_{1:k}} p(x_k, S_{1:k} | z_{1:k})\right)$

- Interpretation of a component and its weight:

$$p(x_k) = \sum^{m_k} p(x_k | a_k = i, Z_{1:k}) Pr\{a_k = i | Z_{1:k}\}$$

$$\underbrace{\cdot\ \text{component}}_{\downarrow\ \hat{x}_k} \qquad \underbrace{\text{weight}}_{\downarrow}$$

- Main complicating factor of this mixture.
  # of mixtures to deal with at each step grows exponentially, generating many hypotheses for tracking path. Mixture reduction simplifies this.

b) It tries to solve the target existence problem. The PDAF state vector is appended with a new state: "target existence". In PDA, a binary decision is sought, which is whether a track exists or not.

→ Solves track initialization?

2/2

c) 1) Extra state in the PDAF-state vector, "target existence" with it's markov transition model.

2) 2-extra steps in workflow compared to PDA. existence prediction & update

Task 4)

a) Predicted pdf: $p_{k|k-1}(x_k) = p(x_k|z_{1:k-1}) = \int f_x(x_k|x_{k-1})\, p_k(x_{k-1})\, dx_{k-1}$

$\downarrow$ if $a_k = 0$

$= \mathcal{N}\left(x_k ; F x_{k-1}\, P_{k-1}\right)$

b) Use NIS

c) "log of corollary 7.3.3" - association probabilities for Poisson clutter model

$$l^{a_k} = \int f_z(z_k^{a_k}|x_k)\, p_{k|k-1}(x_k)\, dx_k \;>\; \mathcal{N}(z_k^{a_k};\ \hat{z}_{k|k-1},\ S_k)$$

d) Posterior event probability

e) Normalised loglikelihoods

$$p_k^{a_k} \triangleq p\{a_k|z_{1:k}\} = \frac{p\{a_k, z_{1:k}\}}{p\{z_{1:k}\}} \quad (7.30) \quad =$$

Task 4)

a)

```
% Standard EKF prediction (7.6)
[xp, Pp] = obj.ekf.predict(x, P, Ts);
```

b)

```
for j = 1:m
    % NIS from EKF
    gated(j) = obj.ekf.NIS(Z(:,j), x, P) <= gSquared;
end
```

c)

```
% calculate log likelihood ratios

% for a_k=0, no detection
ll(1) = logPND + logClutter;

% for a_k>0
for j = 1:m

    llCond(j) = obj.ekf.loglikelihood(Z(j), x, P); %l^a / log(l^a)
    ll(j + 1) = logPD + llCond(j);

end
```

d)

```
%log likelihoods
lls = obj.loglikelihoodRatios(Z, x, P);

% probabilities
beta = exp(lls-logSumExp(lls)); | %%%%%%%%%%%%%%%%%%
```

e)

```
% detected
for j = 1:m
    [xupd(:, j + 1), Pupd(:, :, j + 1)] = obj.ekf.update(Z(j), x, P);
end
```

f)

```
[xred, Pred] = reduceGaussMix(beta, x, P);
```

g)

```
% remove the not gated measurements from consideration
gated = obj.gate(Z, x, P);
Zg = Z(:, gated);

% find association probabilities
beta = obj.associationProbabilities(Zg, x, P);

% find the mixture components pdfs
[xcu, Pcu] = obj.reduceMixture(beta, x, P);

% reduce mixture
[xupd, Pupd] = obj.ekf.update([redacted] Zg, xcu, Pcu);
```