Applied Estimation – EL2320

Help Session 1 – Lab 1 Theory

Jannis Hoppe – jannish@kth.se

Jannis Hoppe – jannish@kth.se

Outline & General Information

Dates, Times etc.

- 5 Help sessions:
  - Wednesday, Nov 7, 10-12, V12 → Lab 1 – Theory
  - Wednesday, Nov 14, 10-12, Q13 → Lab 1 – Implementation
  - Monday, Nov 19, 10-12, B21, → Lab 2 – Theory
  - Wednesday, Nov 28, 10-12, M37 → Lab 2 – Implementation
  - Wednesday, Dec 12, 10-12, V12 → Exam review
- Theory sessions will be wrap ups of the material you need for the labs followed by Q&A
- Implementation sessions will be little demos of the labs followed by Q&A
- During the exam session we (basically I) will solve last year's exam on the board

Nothing new will be presented here! If you start doing the lab and feel confident, you don‘t need to come. The sessions are supposed to be for those who need help!

Jannis Hoppe – jannish@kth.se

Outline & General Information

About passing this course…

To pass the course (and learn something) you need to:

- Make sure you understand the basics of probability theory!
- Have a good “feeling” for KF, EKF, PF, SLAM… not necessarily know all mathematical derivations.
- Do the labs on your own (and on time → Bonus points!)
- Show us that you can implement what you have learned (Project)
- <u>Read the book and attend the lectures</u>

You are not alone on this! If you get stuck on something or have any question, don’t hesitate to contact us!

Jannis Hoppe – jannish@kth.se

Lab 1

Content, exercises, covered material

Content:

- Questions to check if you understand the theory **(today)**
- Smaller exercise to play around with the Kalman Filter **(next session)**
- Implement the Extended Kalman Filter and use it for robot localization **(next session)**

Covered material (that you should review before starting):

- Book chapter 1,2, 3.1-3.3, 7.1-7.5 and Lectures 1-4 (and a bit of 5)

Today's session:

- Review on basic Probability Theory, KF and EKF
- Work on the theory part on your own and ask me if you need help or want to discuss your answer.

Lab 1 - Basics

Jannis Hoppe – jannish@kth.se

Comments on basic probability theory

Do you get what the following (really important!) statements *mean*?

- We assume that our measurements generate a sequence of independent and identically distributed (i.i.d.) random variables $z_1, z_2, \ldots, z_n$. (Does this hold in reality?)
- The random variables x and y are independent of each other: $p(x, y) = p(x) * p(y)$
- The variables x and y are conditionally independent given z : $p(x, y|z) = p(x|z) * p(y|z)$
- The latter does **NOT** imply independence of x and y (why?)
- The Maximum Likelihood Estimate (MLE) and Maximum a Posteriori Estimate (MAP) are not the same

$$\lambda_{MLE} = argmax_\lambda \, p(\{z_i\}|\lambda) \qquad \lambda_{MAP} = argmax_\lambda p(\lambda|\{z_i\}) = argmax_\lambda p(\{z_i\}|\lambda) * p(\lambda)$$

Lab 1 – Basics

Jannis Hoppe – jannish@kth.se

MLE vs. MAP Estimate - Example

A person goes to the doctor and takes a cancer test. The result turns out positive. Since the person has taken Applied Estimation, he knows probability theory well and wants to find out if he should really assume that he has cancer.

From asking the doctor, the person knows that test returns a correct positive result in only 98% of the cases in which the disease is actually present and a correct negative result in only 97% of the cases in which the disease is not present.

With this results the person is able to calculate the Maximum Likelihood Estimate of having cancer. He defines the following events:

- The test turns out positive: $\mathrm{z} = \boldsymbol{pos}$ The test turns out negative: $\mathrm{z} = \overline{\boldsymbol{pos}}$
- He has cancer: $\lambda = \boldsymbol{cancer}$ He does not have cancer: $\lambda = \overline{\boldsymbol{cancer}}$

Help him to find the MLE Estimate regarding being sick or not ($\lambda_{MLE} = argmax_{\lambda}\, p(z|\lambda)$).

Lab 1 – Basics

MLE vs. MAP Estimate - Example

First, we can define the probabilities:

$$p(pos|cancer) = 0.98 \qquad p(\overline{pos}|cancer) = 0.02$$

$$p(\overline{pos}|\overline{cancer}) = 0.97 \qquad p(pos|\overline{cancer}) = 0.03$$

- The test turned out positive, so we need to find $\lambda_{MLE} = \underset{\lambda \in \{cancer, \overline{cancer}\}}{\operatorname{argmax}} p(pos|\lambda)$
- Since $p(pos|cancer) = 0.98 \geq 0.03 = p(pos|\overline{cancer})$, the MLE estimate is that the person has cancer.

Starting to become really concerned, he remembers the concept of MAP estimate, that is, taking into account the probability of having cancer regardless of any test. The person does advanced research and finds out that the probability of having cancer is $p(cancer) = 0.8\ \%$ ($\rightarrow$ $p(\overline{cancer}) = 99.2\ \%$).

Help to find the Maximum a Posteriori Estimate regarding having cancer.

Jannis Hoppe – jannish@kth.se

Lab 1 – Basics

MLE vs. MAP Estimate - Example

- This time we want to find $\underset{\lambda \in \{cancer, \overline{cancer}\}}{argmax} \; p(\lambda|pos)$. Therefore, we calculate the posterior probabilities using Bayes formula:

$$p(cancer|pos) = \frac{p(pos|cancer)p(cancer)}{p(pos)} = \frac{p(pos|cancer)p(cancer)}{(pos|cancer)p(cancer)+(pos|\overline{cancer})p(\overline{cancer})} = 0.21$$

$$p(\overline{cancer}|pos) = \frac{p(pos|\overline{cancer})p(\overline{cancer})}{p(pos)} = \frac{p(pos|\overline{cancer})p(\overline{cancer})}{(pos|cancer)p(cancer)+(pos|\overline{cancer})p(\overline{cancer})} = 0.79$$

- Since $p(\overline{cancer}|pos) = 0.79 \geq 0.21 = p(cancer|pos)$, the MAP estimate is that the person does **NOT** have cancer

After this calculation the person is entirely sure to be healthy and never consults the doctor again.

The MAP and MLE estimates can be completely different, if we have prior knowledge! They are only equal, if the priors are equal!

Jannis Hoppe – jannish@kth.se

Lab 1 – Theory

The Kalman Filter

Basic properties we need for the Kalman Filter

- Linear state transition model with gaussian noise: $x_t = A_t x_{t-1} + B_t u_t + \epsilon_t$
- Linear measurement model with gaussian noise: $z_t = C_t x_t + \delta_t$
- The initial belief has to be gaussian: $bel(x_0) = N(\mu_0, \Sigma_0)$

→ The belief at any time t can be represented by a gaussian.

- For correctness of our calculations, we also need the *Markov assumption* to be true:

> "The *Markov assumption* or *complete state assumption* […] postulates that past and future data are independent if one knows the current state x_t."

- The latter is a fundamental assumption in this course. Familiarize yourself with it (book ch. 2)!

Jannis Hoppe – jannish@kth.se

Lab 1 – Theory

The Kalman Filter – Basic considerations

- Why do we split up our algorithm in two parts?
- What is the intuition behind big entries of Q_t and R_t ?
- What happens for different combinations of Q_t and R_t ?
- Are the absolute values of Q_t and R_t of interest or just their ratios? What about the factor K_t ?

→ This and much more you should be able to answer after the lab!

→ To get a feeling for the equations, assume scalar quantities and try to simplify the equations in case of Q_t , R_t big/small

Btw: Is the Kalman Filter the optimal estimator in some sense?
Is it guaranteed to converge?

Assumptions:

$$x_t = A_t x_{t-1} + B_t u_t + \epsilon_t$$

$$z_t = C_t x_t + \delta_t$$

$$bel(x_0) = N(\mu_0, \Sigma_0)$$

$$\epsilon_t \sim N(0, R_t)\ ,\ \delta_t \sim N(0, Q_t)$$

Kalman Filter:

$$\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$$

$$\bar{\Sigma}_t = A_t \Sigma_t A_t^T + R_t$$

$$K_t = \bar{\Sigma}_t C_t^T \left(C_t \bar{\Sigma}_t C_t^T + Q_t\right)^{-1}$$

$$\mu_t = \bar{\mu}_t + K_t(z_t - C_t \bar{\mu}_t)$$

$$\Sigma_t = (I - K_t C_t)\bar{\Sigma}_t$$

Lab 1 – Theory

The Extended Kalman Filter

Jannis Hoppe – jannish@kth.se

- Non-linear state transition/measurement model → Use extended Kalman filter
- We essentially linearize the model and do the exact same thing as before

Why do we linearize ? Wouldn't it be much better (accurate) to keep the non-linear models?

→ NO! We need a linear model for closed form solutions of the updates and to ensure that we always have a gaussian distribution. NEVER forget this.

- Linearizing shifts our goal from exactly *computing the posterior* (KF) to efficiently *estimating the posterior* as a gaussian (EKF).
- A simple Taylor expansion is used to linearize the model <u>around the mean:</u>

In $x_t = g(u_t, x_{t-1}) + \epsilon_t$ use $g(u_t, x_{t-1}) \approx g(u_t, \mu_{t-1}) + \underbrace{\frac{\partial g(u_t, x_{t-1})}{\partial x_{t-1}}}_{:= G_t \text{ (Jacobian!)}}\Big|_{u_t, \mu_{t-1}} (x_{t-1} - \mu_{t-1})$

Lab 1 – Theory

The Extended Kalman Filter

Jannis Hoppe – jannish@kth.se

- Calculate the jacobians G and H (state transition and measurement model) and replace A and C by them in the KF algorithm

- The goodness of our linear approximation (EKF) depends mainly on:
  - Degree of uncertainty of the current belief (covariance)
  - Degree of local nonlinearity

It looks almost all the same in comparison to the KF, but…

- Is convergence guaranteed ? Can we recover from a really bad (initial) estimate?

- If I have many measurements, does the order in which I process them matter?

- In general: What if I have multiple Hypotheses?

Think about these questions (and the ones on the other slides) and ask if you are unsure!

Applied Estimation – EL2320

Help Session 2 – Lab 1 Implementation

Jannis Hoppe – jannish@kth.se

Lab 1

Content, exercises, covered material

Jannis Hoppe – jannish@kth.se

Content:

- Questions to check if you understand the theory **(last time)**
- Smaller exercise to play around with the Kalman Filter **(today)**
- Implement the Extended Kalman Filter and use it for robot localization **(today)**

Covered material (that you should review before starting):

- Book chapter 1,2, 3.1-3.3, 7.1-7.5 and Lectures 1-4 (and a bit of 5)

Today's session:

- Demo of what your program should do/ look like when it is done
- Work on the implementation on your own. If you have (bigger) problems, I will help you.

Lab 1 - Implementation

Tips

Jannis Hoppe – jannish@kth.se

When you write the code, you should…

- Do not blindly hack a solution – Think about what you are doing and interpret the results
- Basically put the given pseudo code in MATLAB
- When it seems like you have to do something really advanced, not stated in the instructions, you are probably on the wrong way.
- Document your results well, you will need to go over the labs as an exam preparation again in 2 months!
- **PLEASE, do not code too messily and name your variables appropriately (according to the lab instructions). Otherwise I will not be able to help you when you get stuck!**
- Implement everything in the suggested order and use the provided testing functions