F2E5216/TS1002 Adaptive Filtering and Change Detection

Fredrik Gustafsson (LiTH) and Bo Wahlberg (KTH)

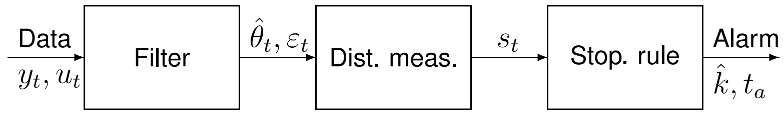
Linköpings universitet

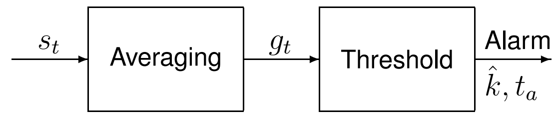
Lecture 3

Change detection methods for change in the mean.

- The CUSUM test
- Filter and detector evaluation
- The likelihood concept
- Maximum likelihood and likelihood ratio based CD
- Information based CD

Change detection based on whiteness test





Example on 'named' algorithms:

- Exponential forgetting of $s_t = \varepsilon_t$ gives GMA.
- Sliding window of $s_t = \varepsilon_t^2$ gives a $\chi^2(L)$ test.
- Sliding window of $s_t = K_t \varepsilon_t$ gives asymptotic local approach.

Cumulative Sum (CUSUM)

To test for a positive change in mean:

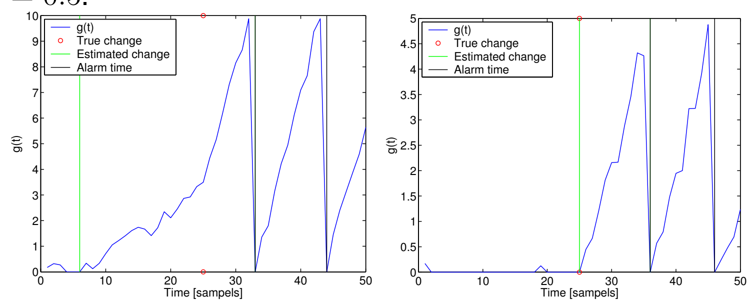
$$\begin{aligned}
s_t &= y_t - \nu \quad \text{(Subtract a drift term to prevent positive drift)}\\
g_t &= g_{t-1} + s_t \quad \text{(Sum)}\\
g_t &= 0, \text{ if } g_t < 0 \quad \text{(To prevent negative drift)}\\
& \quad \hat{k} = t \text{ if } g_t < 0 \quad \text{(Possible estimate of change time)}\\
g_t &= 0, \text{ and } t_a = t \text{ and alarm if } g_t > h > 0.
\end{aligned}$$

Rule of thumb: The drift term should be chosen as one half the expected change magnitude

Example: CUSUM

$$y_t = \begin{cases} 0 + e(t), & \text{for } 0 < t \le 25 \\ 1 + e(t), & \text{for } 25 < t \le 50. \end{cases}$$

where $e \in N(0, 0.1)$. Compare $h = 10$ and $\nu = 0$ with $h = 5$ and $\nu = 0.5$.



CUSUM Change Time Estimation

Since g_t is linearly increasing (in the mean) after a change, take $\hat{k}$ to the last time the CUSUM test was reseted ($g_t < 0$).

Two-sided tests

Apply two tests in parallel, where the second one has $-y_t$ as the input.

Tuning

Start with a large threshold and ν equal to half the expected change magnitude. Then reduce the threshold so the required number of false alarms or acceptable delay of detection are obtained

- For fewer false alarms, increase ν
- For faster detection decrease ν

The Likelihood Concept

Likelihood is a measure of likeliness of what we have observed, given the assumptions we have made.

For independent observations, the likelihood is computed by

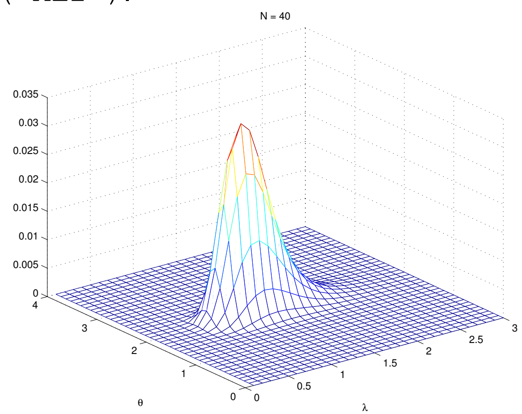
$$
\begin{aligned}
y_t &= \theta + e_t, \ \ \mathrm{Var}(e_t) = R \\
l_t(\theta, R) &= p(y^t|\theta, R) = \prod_{i=1}^{t} p(y_t|\theta, R) \\
&= l_{t-1}(\theta, R) p(y_t|\theta, R)
\end{aligned}
$$

To avoid numerical problems ($|l_t| > 10^{128}$ out of range!) and to get nicer expressions (sum of squared residuals), the negative log likelihood is often used:

$$-\log l_t(\theta, R) = -\log l_{t-1}(\theta, R) - \log p(y_t|\theta, R)$$

Point-Mass Approach

Evaluate the function $p(y_t|\theta, R)$ on a 2D grid for θ and R. Run `lecture('ML2')`.



Maximum Likelihood Estimator

The ML estimate is defined as the maximizing argument of the likelihood

$$
\begin{aligned}
\widehat{(\theta, R)}^{ML} &= \arg\max_{\theta,R} l_t(\theta, R) \\
&= \arg\min_{\theta,R} -\log l_t(\theta, R).
\end{aligned}
$$

The example gives $\widehat{(\theta, R)}^{ML} = (2.1, 1.1)$

Marginalization

Joint likelihood for θ, R: $p(y^t|\theta, R)$

Marginalization gives the likelihood for one variable only, *e.g.*

$$
\begin{aligned}
p(y^t|\theta) &= \int p(y^t|\theta, R)p(R)dR \\
p(y^t|R) &= \int p(y^t|\theta, R)p(\theta)d\theta
\end{aligned}
$$

Point Mass Approach

Just sum the rows or columns!

ML estimates $\hat{\theta}^{ML} = 2.0$ and $\hat{R}^{ML} = 1.1$.

Note $(\hat{\theta}^{ML}, \hat{R}^{ML}) \neq \widehat{(\theta, R)}^{ML}$!

A General Adaptive Likelihood Estimator

A recursive and adaptive version using a forgetting factor α is

$-\log l_t(\theta, R) = -\alpha \log l_{t-1}(\theta, R) - (1-\alpha)\log p(y_t|\theta, R)$

Run `lecture('ML3')` for an example with likelihood forgetting and one abrupt change.

Explicit Formulas for Gaussian Distribution

The noise $e_t \in \mathrm{N}(0, R)$ gives the likelihood

$$p(y^t|\theta, R) = (2\pi R)^{-t/2} e^{-\frac{1}{2R}\sum_{i=1}^{t}(y_i-\theta)^2}$$
$$-2\log p(y^t|\theta, R) = t\log(2\pi R) + \frac{1}{R}\sum_{i=1}^{t}(y_i-\theta)^2$$

and (joint) ML estimates
$$\hat{\theta}^{ML} = \overline{y} = \frac{1}{t}\sum_{i=1}^{t} y_i$$
$$\hat{R}^{ML} = \overline{y^2} - \overline{y}^2 = \frac{1}{t}\sum_{i=1}^{t}(y_i - \hat{\theta}^{ML})^2$$

Compare to the formula $\mathrm{Var}(X) = E(X^2) - (E(X))^2$.

Likelihood based Change Detection

Basic idea: ML estimation of jump time k, where

$$p(y^t|k, \theta_1, \theta_2, R_1, R_2) = p(y_1^k|\theta_1, R_1)p(y_{k+1}^t|\theta_2, R_2)$$

We need to compute the likelihood for subsets of the observations!

$$\text{Data} \quad \underbrace{y_1, y_2, .., y_k}_{p(y_1^k|\theta_1, R_1)} \underbrace{y_{k+1}, y_{k+2}, .., y_t}_{p(y_{k+1}^t|\theta_2, R_2)}$$

Compute the product for all possible change times k.

Back to standard detection problem: $H(0)$: no jump;
$H(k)$: jump at time t=k

Nuisance Parameters

For change detection, the parameters θ_1, R_1 before the change and θ_2, R_2 after the change are irrelevant.

- **Prior knowledge:** One or both parameters can be known
- **Maximization.** Replace one or both parameters by its ML estimate in each interval.
- **Marginalization.** Integrate out one or both parameters in each interval.

Interesting Cases 1 and 2

1. θ is unknown, R is known.

$$-2\log l_t^{MGL} \approx t\log(2\pi R) + \frac{t\hat{R}}{R}$$
$$-2\log l_t^{MML} \approx (t-1)\log(2\pi R) + \log(t) + \frac{t\hat{R}}{R}$$

Best if R is approximately known.

2. Unknown θ, unknown R, and might change after the change time

$$-2\log l_t^{MGL} \approx t\log(2\pi) + t + t\log(\hat{R})$$
$$-2\log l_t^{MML} \approx t\log(2\pi) + (t-5) + \log(t) - (t-3)\log t(t-5) + (t-3)\log(\hat{R})$$

Most general and requires no prior knowledge.

Interesting Cases 3 and 4

3. θ is known (typically to be zero), R is unknown and abruptly changing.

$$
\begin{aligned}
-2\log l_t^{MGL} &\approx t\log(2\pi) + t + t\log(\hat{R}) \\
-2\log l_t^{MML} &= t\log(2\pi) + (t-4) - (t-2)\log t(t-4) + \\
&\quad (t-2)\log(\hat{R})
\end{aligned}
$$

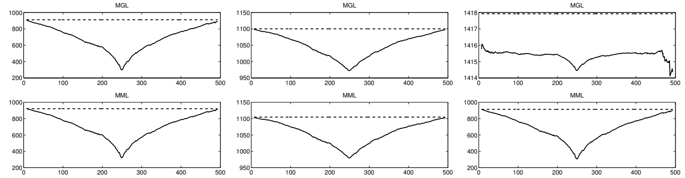
Suitable for detecting variance changes.

4. θ is unknown, R is unknown and constant. See Chapter 7.

Example

$$
y_t = \begin{cases} 0 + e(t), & \text{for } 0 < t \le 25 \\ 1 + e(t), & \text{for } 25 < t \le 50. \end{cases}
$$

$p(y_1^k)p(y_{k+1}^t)$ as a function of k, for methods 1,2,3.



Conclusion: marginalization (lower row) works (sometimes) better than maximization.

Likelihood based Segmentation

Segmentation = multiple change point estimation

Applications:
I. Off-line data analysis

II. Gives recursive algorithms with natural recovery after alarm (there is no initialization problem).

Toolbox: `detectM` with `lf=[1 lambda]` for case 1 above and `lf=[2]` for case 2.

Change Detection based on Model Validation

$$
\text{Data} \quad \underbrace{y_1, y_2, .., y_{t-L}}_{\hat{\theta}_1, \hat{R}_1} \underbrace{y_{t-L+1}, .., y_t}_{\hat{\theta}_2, \hat{R}_2}
$$

A two-filter approach. What distance measures between the models are available?

Gaussian Test:

$$\hat{\theta}_1 - \hat{\theta}_2 \in \mathrm{N}(0, P_1 + P_2), \quad \text{under } H_0$$

χ^2 test:

$$\frac{(\hat{\theta}_1 - \hat{\theta}_2)^2}{P_1 + P_2} \in \chi^2(1), \quad \text{under } H_0$$

GLR or MLR tests

Example

Design a test with a probability of false alarm of 95%.

Gaussian test:

$$P\left(\underbrace{\frac{|\hat{\theta}_1 - \hat{\theta}_2|}{\sqrt{P_1 + P_2}}}_{N(0,1)} > 1.96\right) = 0.95$$

χ^2 test (**Toolbox:** `chi2(1,0.95) = 3.79`):

$$P\left(\underbrace{\frac{(\hat{\theta}_1 - \hat{\theta}_2)^2}{P_1 + P_2}}_{\chi^2(1)} > 3.79 = 1.96^2\right) = 0.95$$

Summary

- Whiteness based CD (`detect1`) defined by the **distance measure** between the residuals and zero, and the **stopping rule**.

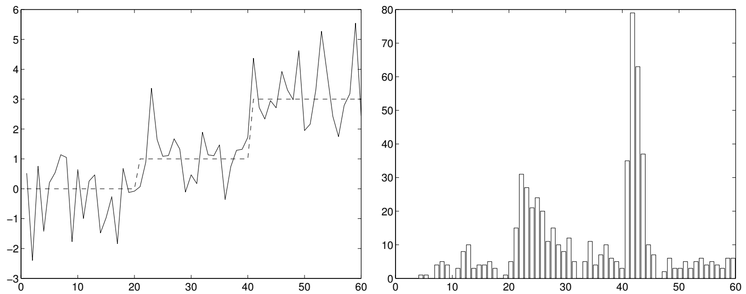
- Parallel filter based CD (`detect2`) defined by the **distance measure** between the hypothesis change and no change and the **stopping rule**.

- Segmentation approaches (`detectM`) defined by the **loss function** that is minimized w.r.t. k^n.

Techniques: likelihoods, likelihood ratios, hypothesis tests, least squares

Change Detection Evaluation

A certain CUSUM test is applied to 250 realizations of the signal below. The alarm times are shown in the histogram.



How to measure the performance of the detector?

Performance measures

- Mean time between false alarms (MTFA)

$$\mathrm{MTFA} = \mathrm{E}(t_a | \text{no change})$$

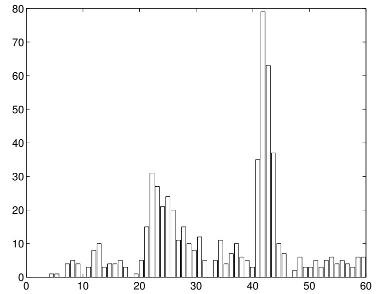
Related to MTFA is the false alarm rate (FAR).

- Mean time to detection (MTD).

$$\mathrm{MTD} = \mathrm{E}(t_a - k | \text{a given change at time } k)$$

How long do we have to wait after a change until we get the alarm?

- Missed detection rate (MDR). What is the probability of not receiving an alarm, when there has been a change. Note that in practice, a large t_a can be confused with a missed detection and a false alarm.



DFD1	DFD2	MDR1	MDR2	FAR
4.7	3.0	0.27	0	0.01

As can be expected, the delay for detection (DFD) and missed detection rate (MDR) are larger for the smaller first change.

Note the characteristic distribution of alarm times.

Off-line measures

Accuracy of change point location, like

$$\frac{1}{n}\sum_{i=1}^{n}(\hat{k}_i - k_i^o)^2$$

The Minimum Description Length (MDL). How much information is needed to store a given signal? The latter measure is relevant in data compression and communication areas.

ARL

Average run length function, ARL(θ).

$$\mathrm{ARL}(\theta) = \mathrm{E}(t_a - k | \text{a change of magnitude } \theta \text{ at time } k)$$

A function that generalizes MTFA and MTD. How long does it take before we get an alarm after a change of size θ. A very large value could be interpreted as that a missed detection is quite likely.

$MTFA = ARL(0)$

$MTD(\theta) = ARL(\theta)$

ARL for CUSUM

Recall the CUSUM test

$$
\begin{aligned}
g_t &= g_{t-1} + y_t - \nu \\
g_t &= 0, \text{ if } g_t < 0 \\
g_t &= 0, \text{ and } t_a = t \text{ and alarm if } g_t > h > 0.
\end{aligned}
$$

Rough approximation (noise-free case): alarm when $g_t = (t-k)(\theta - \nu) > h$.

Reality, alarm time depends on ν, h and σ = std(y_t).

Anyway, ARL is a function of only two variables:

$$
\text{ARL}(\theta; h, \nu, \sigma) = f\left(\frac{h}{\sigma}, \frac{\theta - \nu}{\sigma}\right)
$$

Approximations of the ARL function

The theoretical function is given by an integral equation which can be solved numerically (see `cusumarl`).

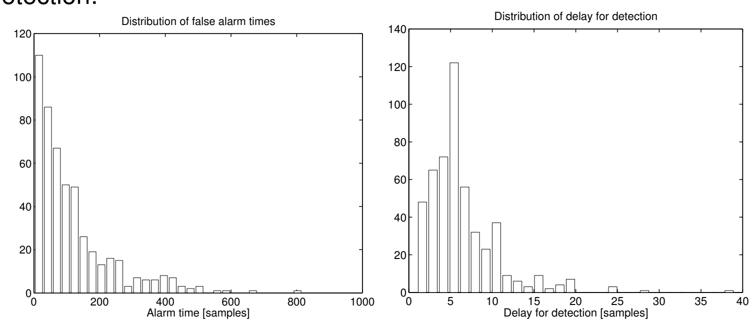
Wald's approximation is very accurate (see `cusumarl`).

$$
\text{ARL} = \frac{e^{-2(h/\sigma + 1.166)\mu/\sigma} - 1 + 2(h/\sigma + 1.166)\mu/\sigma}{2\mu^2/\sigma^2}
$$

$\mu = \theta - \nu$. MC simulations (see `cusumMC`).

Run length distribution

The run length distribution says more than just the average ARL value. Monte Carlo simulations give false alarms and mean delay for detection:



Exercises for Lectures 2 and 3

Link on homepage

`http://www.control.isy.liu.se/~fredrik/detect/exercises.pdf`

Exercise: 4, 5, 6, 8, 9, 10, 13

Next Time: Adaptive Filtering

- Linear regression models
- Application areas
- Algorithms
- Properties
- Application examples