Exercises

The exercises are categorized into **O**rientation or **A**dvanced ones, and in **T**heoretical or **C**omputer exercises. An exercise **OT** is thus of general theoretical interest, while **AC** is recommended for people who want to study the practical aspects in more detail. There are three **projects**.

Part II

Exercise 1 (OC) Use the toolbox to examine a simulated signal. Start with the default one when `guidetect` is started. Continue by adding more changes of different sizes, so that the signal contains 4 segments of length 50 each, with mean levels 0,1,3,6, respectively. Let the noise variance be 1. Compare and tune different algorithms with respect to the mean square error (use the table function in `adaptive tool`). Which is the best algorithm with respect to estimation? Most reliable results are obtained by using Monte Carlo simulations.

Exercise 2 (OC) As above, but evaluate performance with respect to change detection. Use the table function in `change tool`.

Exercise 3 (OC) Load the fuel consumption signal in `guidetect`. Try the default filter in the one-model approach (change `filter structure`). The signal estimate looks fine, but perhaps there are too many alarms? Try to tune the change detector to resolve this problem. Compare with the results using the default settings of the parallel filter (`detect2`) and multi-model (`detectM`) approaches.

Exercise 4 (OT) A often used rule of thumb for choosing the forgetting factor in RLS is an equivalent sliding window size $N \approx \frac{1}{1-\lambda}$. As an alternative, a natural, statistically based, rule of thumb is

$$N \approx \frac{2}{1-\lambda} \leftrightarrow \lambda \approx 1 - \frac{2}{N},$$

for λ close to one. Here N is interpreted as the size of the corresponding sliding window, for which the variance of WLS and RLS would be the same.

a. Prove this relation for the change in the mean model, using the definition of the loss functions ((5.43) for RLS), matching expressions like (5.48) and (5.49).

b. Try also to use the RLS algorithm 5.3 directly, matching $P(t)$ with the variance of WLS. The result is different basically because $P(t)$ is not the covariance matrix of $\hat{\theta}_t$.

Exercise 5 (OT) To minimize the transient in RLS, it would be natural to try a forgetting factor that starts with a low value and increases with time. Assuming that the true parameter θ is constant in time, how should one choose λ_t to minimize the transient? Hint: compare the RLS and off-line LS loss functions.

Exercise 6 (OT) The fast and slow filters in Example 1.1 were in fact RLS filters with forgetting factors 0.7 and 0.9, respectively. One might be tempted to average the filter outputs to improve the performance. The new estimate is still a linear operation on the measurements. Which linear filter does this estimate correspond to?

Exercise 7 (OT) In Example 12.1 a linear filter (Butterworth LP type) was applied to noisy measurements of a signal, and then Monte Carlo simulations were used to find the mean filter output. An alternative could be to apply the filter directly to the signal part of y, to see what the limiting estimate of the filtered signal is. Motivate why this approach would give the same result, asymptotically.

More generally, let M be the Monte Carlo operation, and F the filter operation. The result can then be written $M(F(y))$. When is it possible to reverse the operations and study $F(M(y))$? For design of linear filters? For change detectors?

Exercise 8 (OT) Prove the expression for the MLR and GLR tests in equations (3.47) and (3.48).

Exercise 9 (OT) In Example 3.1, the estimate from the Kalman filter differs slightly from the estimates from RLS and LMS. How should Q be chosen to give identical results, at least after transient effects have faded away? Can we design a Kalman filter to give the same transient as well, and how can this be done?

Exercise 10 (OT) Here we will derive an adaptive filter for a problem that does not fit into the standard models.

a. What is the expected value of x^2 of the exponential distribution

$$f(x) = 1/a \exp(-x/a), x > 0, a > 0?$$

b. What is the maximum likelihood estimate of x^2 based on N samples?

c. Give a recursive filter for the estimate.

d. What is the variance of the estimate in b?

Exercise 11 (AT) This exercise is similar to 10, with the difference that it is impossible to give a recursive implementation and to analyse the variance properties analytically.

a. What is the median of the exponential distribution

$$f(x) = 1/a \exp(-x/a), x > 0, a > 0?$$

Hint: compute the probability distribution function.

b. What is the point estimate of the median based on only one sample?

c. Try to formulate the variance of the median estimate from a sample of size N (assume N odd) as an integral, without trying to solve it. Start with writing down the definition of variance of an estimate. As you soon realize, the variance is hardly possible to compute analytically, and this is one simple example where numerical procedures as bootstrap must be used.

Exercise 12 (AT) Design a CUSUM test that gives a delay for detection of 6 samples in the case the signal mean changes from 0 to 3. Assume noise variance 1. How should ν be chosen if this is the change we would like to detect, typically?

a. Choose the threshold from (12.22).

b. Use the data in Example 12.6 to approximate the threshold. Interpolate in the table if necessary.

Exercise 13 (AT) Consider a signal abruptly changing from 0 to θ at time $t = 0$, with Gaussian measurement noise. What is the theoretical mean time to detection (MTD) if an exponential filter (3.12) or sliding window, respectively, is applied as the filter in Figure 3.1, and the stopping rule is defined as $s(t) = \varepsilon(t)$ with the threshold decision (3.19). The result is a function $MTD(\theta, h, \lambda)$ or $MTD(\theta, h, L)$.

Exercise 14 (AC) As Exercise 12, but use `cusumdesign` to design the threshold and `cusumarl` and `cusumMC` to evaluate the design.

Exercise 15 (AC) As Exercise 11, but use Monte Carlo and bootstrap to compute the median and its variance from, say, 1000 random numbers from the exponential distribution. Hint: for simulation, use the inverse of the probability distribution function and uniformly distributed random numbers. $F^{-1}(u)$ will then be exponentially distributed.

Exercise 16 (AC) Compute the non-causal, causal and one-step ahead predictive Wiener filters for the one-dimensional tracking model, in Matlab defined as

```
x(t+T) = [1 T;0 1] x(t) + [T^2/2;T] v(t)
y(t) = [1 0] x(t) + e(t)
```

Assume $\mathrm{Var}(e) = \mathrm{Var}(v) = 1$. If you need a numerical value of T, as you probably will, use $T = 5$s.

Exercise 17 (Project) We will here in detail study the problem of detecting abrupt changes in Poisson processes, as illustrated in Section 4.6.1. Another application of Poisson processes is burst-like traffic in data networks. We will assume that the arrival times are differentiated, such that the intensity shows up as a piecewise constant mean in the signal. Figure 3.12 shows the result from `detectM`, which assumes Gaussian noise. This is clearly suboptimal. Derive the marginalized likelihood for a segment of data with the same intensity, analogously to the calculations in Appendix 3.A. Use this to modify one of the change detection algorithms in the toolbox. Perform a simulation study, where both small and large intensity changes are present. Finally, apply the method to the photon emission data (`photons`). A short report with the derivation, the algorithm modification, the simulation study and the application example is expected.

Part III

Exercise 18 (OC) Study the ARX example in the toolbox GUI. Tune the adaptive filters in `adfilter` as well as possible. Also test the functions `detect1,detect2,detectM`.

Exercise 19 (OC) Consider the `eeg_human` signal in the toolbox. The data `Y` consists of 20 experiments under similar conditions. Apply different change detectors assuming an underlying AR(2) model, and compare the delay for detection.

Exercise 20 (OC) Consider the `equake` signal in the toolbox. The data `y` consists of 14 different signals from the same earthquake. Which one is the best one to use for detection and estimation of the starting time? Compare delay for detection (you don't know the truth here!) and false alarm rate for your designed test. Can you suggest an algorithm that uses all available signals?

Exercise 21 (OC) *The inverse of spline interpolation.* Consider the example given in the help text to Matlab's `spline` function. A coarsely sampled sinusoid is interpolated using polynomial spline functions. Use change detection to implement the inverse operation! That is, use `yy` as the input, construct an appropriate regressor and use the model format `nnn=-2` (regression model). The output should be change times corresponding to the vector x (or rather its indeces in `xx`) and the spline parameters.

The result may seem discouraging in that the interpolation points are not recovered. To understand what the problem is, do a plot like:

```
plot(x,y,'o',xx,yy,'-',xx(jhat),yy(jhat),'rx',...
     xx,diag(phi*thseg),'r--')
```

That is, the non-exact segmentation is due to numerical problems. The use of orthonormal polynomials may improve the result. Try with Chebyshev polynomials. Also, simply try another function with less numerical problems (the sinusoid is too similar polynomials)

Exercise 22 (OC) As a continuation of OC 21 try a piecewise linear model. Compare in `detectM` the different assumptions of noise variance; unknown and known. In the latter case, the assumed variance determines the modeling accuracy.

Exercise 23 (OT) Consider the implementation of RLS in Algorithm 5.3.

a. How should the calculations be organized to get the most efficient code? Measure complexity as the number of multiplications, and assume scalar measurements, $n_y = 1$.

b. Show that the gain K_t in (5.45) can be written $K_t = P_t\varphi_t$. Does this insight imply the excistence of a more efficient implementation?

Exercise 24 (OT) Repeat Exercise 4.a with vector valued θ. Use the assumption of quasi-stationary regressor, so

$$\frac{1}{N}\sum_{k=0}^{t}\varphi_k\varphi_k^T \to Q,$$

for some Q.

Exercise 25 (OT) Assume that we want to estimate the time-varying noise variance of a signal estimation problem (θ_t scalar). We can then for instance use an RLS filter with forgetting factor λ to estimate θ, and then a second RLS filter to estimate the noise variance as

$$\hat{R}_t = \lambda_R\hat{R}_{t-1} + (1-\lambda)\varepsilon_t^2.$$

Based on the performance analysis is Section 5.5, derive an approximate expression for the expected value of $\hat{R}_t$. Assume for simplicity that R_t is constant over the time horizon implied by the forgetting factor λ_R. Would the result be different if LMS is used for parameter estimation?

Exercise 26 (OT) One advantage of the Kalman filter is that we can get different adaptation gains for different parameters, as illustrated in the friction estimation case study. One idea to get this property of RLS and LMS is to introduce a diagonal transformation matrix T and writing $\bar{\theta} = T\theta$. Show that RLS and LMS get the same relative update size anyhow.

Exercise 27 (OT) Prove (5.81) by direct calculations, starting with taking the expected value of the LS parameter estimate.

Exercise 28 (OT) Prove (7.9) by direct calculations.

Exercise 29 (OT) Prove the result in Example 7.1.

Exercise 30 (OT) Consider the process

$$y(t) = e^{-\theta y(t-1)} + e(t),$$

where $e(t)$ is white noise with zero mean and variance 1.

1. What is the one-step ahead predictor $\hat{y}(t;\theta) = \mathrm{E}(y(t)|\theta, y(t-1), y(t-2), \ldots, y(0))$? Give an LMS algorithm for adaptively estimating θ.
2. Simulate $N = 1000$ data and tune the step size μ in LMS such that a nice convergence in $\hat{\theta}(t)$ from $\theta(0) = 0$ to $\theta = 1$ is achieved. (4p)

Exercise 31 (AT) Consider diagnosis according to Example 6.9. Assume a scalar measurement and offset and drift as possible sensor faults. That is, the model is $y_t = \theta_a + t\theta_b + e_t$. Compute the projection matrices explicitely. What are the test statistics for detection and isolation? What happens if the noise variance tends to zero?

Exercise 32 (AT) Suppose a fixed batch of data is either white noise or generated by white noise filtered by an AR(2) model. Propose at least three different tests for this problem based on hypothesis test, confidence interval and likelihood, respectively. This can be seen as a *model validation* problem.

Exercise 33 (AC) Continuation of Exercise 31. Simulate signals of length, say, 100 with abrupt changes at time 40 in offset and drift, respectively. Use any detector for change detection. Then use the estimated change time to pick out data for isolation and use for instance `adfilter` to compute the different projections V (with `fflam=1` the noise variance estimate is related to V as $V = N\hat{\lambda}$). Finally, compute the identified fault (this requires no more function calls). What is the point with this procedure compared to taking the parameter estimate from the change detector directly?

Exercise 34 (Project) Use polynomials of different degrees in the regression vector for the task of smoothing in spectral analysis. That is, the signal y is the periodogram for a given system. Try both adaptive filters and change detectors. Compare at least to standard methods as `etfe`.

Part IV

Exercise 35 (OT) Consider the following system:

$$x(t+1) = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} v(t)$$
$$y(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} x(t) + e(t)$$

where e and v are independent white noise processes with variance 1.

Compute the stationary Kalman filter and give explicitely the filter $F(q)$ in the relation $\hat{x}(t|t) = F(q)y(t)$, and plot the filter's impulse response.

Exercise 36 (OT) Fuel consumption in cars. One sensor measures the level $y_1(t) = h(t)+e_1(t)$, and another one the flow $y_2(t) = f(t)+b(t)+e_2(t)$. Here, $h(t)$ denotes the true level, $f(t)$ the true flow and $b(t)$ is an unknown offset. Both measurement errors $e_1(t), e_2(t)$ are independent with variances $\sigma_1^2 = 1$ and $\sigma_2^2 = 0.1$, respectively. That is, the flow is more accurately measured, buth on the other hand has a time-varying offset $b(t)$.

We measure once per second $T = 1$ [s]. The time variation in the offset can be modelled as a slowly varying random walk $b(t+1) = b(t) + w_2(t)$, where $\mathrm{E}(w_2^2(t)) = 0.01$. The flow is also modeled as a random walk $f(t+1) = f(t) + w_1(t)$ där $\mathrm{E}(w_1^2(t)) = 1$. Use the relation $h(t+1) = h(t) + Tf(t)$.

1. Formulate a state space model using the state vector $x(t) = (h(t), f(t), b(t))^T$.
2. Compute the stationary Kalman filter.
3. What is the expected variance in the level estimate $\hat{h}(t|t)$? How large is the improvement, compared to using $\hat{h}(t|t) = y_1(t)$?

Exercise 37 (OT)

Exercise 38 (OT)

Exercise 39 (OT) Derive the one-dimensional form of the Kalman filter in (8.44) using the projection method in Section 13.1.3.

Exercise 40 (OT) Square root filtering.

a. Prove Algorithm 8.7.

b. Suppose the left hand side in the QR factorization step is augmented with the row

$$\begin{pmatrix} -y_t^T R_t^{-T/2} & \hat{x}_{t|t-1}^T P_{t|t-1}^{-T/2} & 0 \end{pmatrix}.$$

Show that R^T after QR factorization now contains the normalized innovation and normalized state estimate. That is, all that is needed comes out from the factorization directly!

Exercise 41 (OT) Consider the DC motor in Example 8.4, with fault model according to Section 11.5.1. Formulate an augmented state space model for tracking the two fault modes. Hint: see Section 9.1 and equation (9.3).

Exercise 42 (OT) Prove (8.102). Assume $\hat{x} = L_1\hat{x}_1 + L_2\hat{x}_2$. From unbiasedness it follows that $L_1 + L_2 = I$. Optimize the covariance of $\hat{x}$ with respect to L_1, L_2 under this constraint!

Exercise 43 (OT) Consider the tracking example and in particular the measurement equation as described in Example 8.14. Suppose the following sensors are available:

- Radar measuring bearing and range with standard deviations $\sigma_\theta = 1^o$ and $\sigma_r = 100$m, respectively.
- Infra-red (IR) sensor measuring bearing with standard deviation $\sigma_\theta = 0.1^o$.
- Radar warning system measuring bearing with standard deviation $\sigma_\theta = 1^o$.
- Estimated position from another aircraft in cartesian coordinates with standard deviations $\sigma_1 = 100$m and $\sigma_2 = 100$m, respectively.

Propose a way to fuse these information sources in a centralized Kalman filter. What is the main problem here with the decentralized filter structure?

Exercise 44 (OT) Generate the signal $y(t)$ as

```
y0=[ones(N,1); (1:1/N:2)'; 2*ones(N-1,1)];
y=y0+0.1*randn(3*N,1);
```

The signal is piecewise constant of the form $y(t) = a_t + b_t t$. A suitable state space model with $x_t = (a_t, b_t)^T$ is given by

$$\begin{aligned} x(t+1) &= x(t) + w(t) \\ y(t) &= (1,\ t)x(t) + e(t) \\ \hat{x}(0) &= 0. \end{aligned}$$

Apply the time-varying Kalman filter `adkalman` and tune Q, R, P_0. Present a plot of the ten step ahead predictor $\hat{y}(t+10|t) = C(10)\hat{x}(t+10|t)$ together with the signal.

Exercise 45 (OC) Consider the DC motor in Example 8.4, implemented in `nnnDCm0`.

a. What is the stationary Kalman filter? Hint: $\bar{P}$ is computed by `adkalman` using `kf=[1,0]`.

b. What is the transfer function of the stationary Kalman filter? Hint: use Control System ToolBox, `sys=ss(A,B,C,D,T)` (what are the matrices A, B, C, D for the Kalman filter?), `bode(sys)` and/or `tf(sys)`.

c. Examine the bias error induced by using $A^{(1,2)} = 0.35$ in the design. How does this affect the Bode plot? What is the stationary mean square error $\bar{P}^p$ in Section 8.6.3, assuming that the stationary Kalman gain $\bar{K}^d$ is used?

Exercise 46 (AC) Consider the DC motor in Example 8.4, with fault model according to Section 11.5.1. Using the augmented state space model from Exercise 41, it is easy to apply different filters and detectors using the functions `adkalman, detect1, detect2, detectM`. Compare this approach and the results obtained to the GLR test in Example 9.3 and parity space fault detection in Section 11.5.1. Hints: For simulation, extract the relevant code from the function `book` after the switch `faultdetect2`. For estimation, modify the model in `nnnDCm0` to the augmented model.

Exercise 47 (AC) Consider the example in 7.7.3. Design a Kalman filter for tracking the car and detecting manoeuvres. Use a fifth order motion model with turn rate as the extra state and try to take care of the fact that velocity changes are relatively unimportant manoeuvres for navigation on a map. That is, accept larger lateral than longitudinal accelerations.

Exercise 48 (Project) Consider the vertical aircraft dynamics in Section 11.5.3. Reformulate the problem so that `glr, adkalman, detect1, detect2, detectM` can be used. See Exercises 41 and 46. Apply it to the cases with measurement error and model error. This is easiest done by modifying the code in the function `book` after the switch `faultdetect2`. The task is to compare pros and cons of statistical and algebraical approaches to fault detection and isolation. In particular the GLR test is interesting to compare with because isolation can be accomodated as outlined in Chapter 9.