DTU Compute

02425 Diffusions and SDE's
September 21, 2023
UHT/uht

Exercise 4: Linear systems

Steady-state variance structure for a mass-spring-damper system

Consider the mass-spring-damper system in the notes (5.1), (5.2), p. 90, with the force $\{F_t : t \geq 0\}$ being white noise with a given intensity $S_{FF}(\omega) = \sigma^2$.

Question 1: Write the system in the standard form $dX_t = AX_t\ dt + G\ dB_t$, i.e. specify A and G. *Note:* Yes, the answer is *almost* given in the textbook.

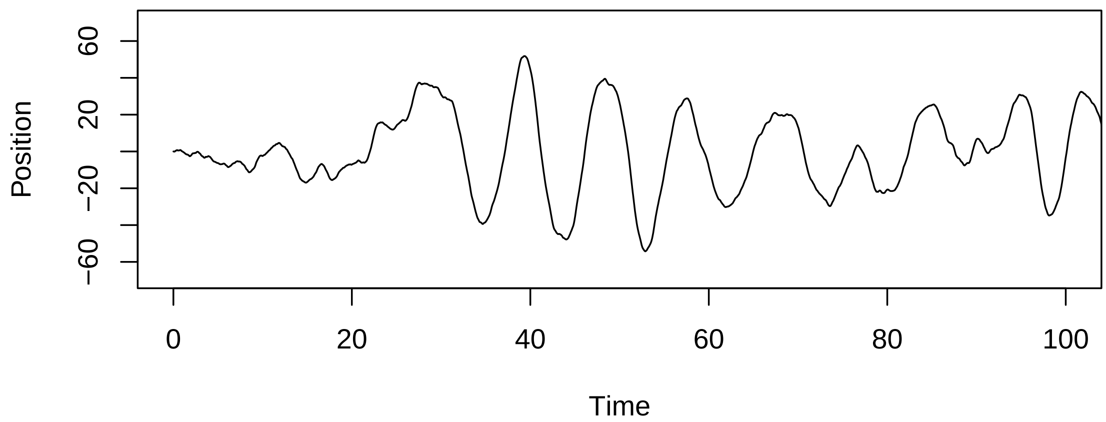
Solution: The system is written as $dX_t = AX_t\ dt + GdB_t$ where

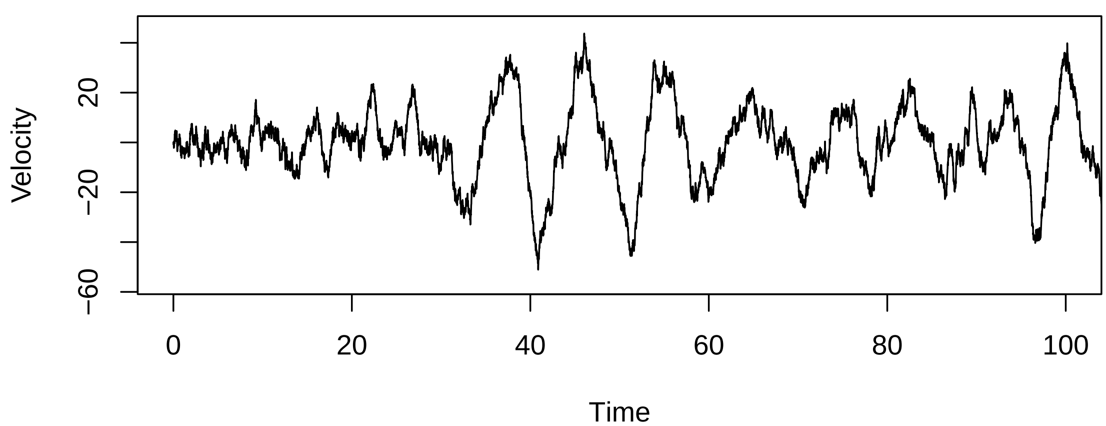
$$A = \begin{bmatrix} 0 & 1 \\ -k/m & -c/m \end{bmatrix}, G = \begin{bmatrix} 0 \\ \sigma/m \end{bmatrix},$$

Question 2: Using the general form, simulate the system on the time interval $t \in [0, 1000]$ using the Euler-Maruyama method. Take system parameters $m = 1$ kg, $k = 0.5$ N/m, $c = 0.2$ Ns/m, $\sigma^2 = 100$ N$^2 s$. Let the system start at rest at $t = 0$. Use a time step of $\Delta t = 0.01$ s. Plot the first part of the sample path, $t \in [0, 100]$.

Solution: The following code simulates the system using the Euler-Maruyama method. We time step also the Lyapunov differential equation governing the variance.

```r
require(SDEtools)

## Loading required package: SDEtools

## System parameters
m <- 1    # [kg]
k <- 0.5  # [N/m]
c <- 0.2  # [N*s/m]
sigma <- 10 # [N sqrt(s)]

## Q 1.1
A <- array(c(0,-k/m,1,-c/m),c(2,2))
G <- array(c(0,sigma/m),c(2,1))


## Simulation parameters
T <- 1000
dt <- 0.01

## Setup arrays
tvec <- seq(0,T,dt)
P <- array(0,c(2,2,length(tvec)))  # This also computes the solution of the Lyapunov equation
X <- array(0,c(2,length(tvec)))

## Simulate sample path of Brownian motion
B <- rBM(tvec)
dB <- diff(B)

## Main time loop, Euler stepping the SDE and the Lyapunov equation
for(i in 1:(length(tvec)-1))
{
    X[,i+1] <- X[,i] + A %*% X[,i] * dt + G * dB[i]
    P[,,i+1] <- P[,,i] + (A %*% P[,,i] + P[,,i] %*% t(A) + G %*% t(G)) * dt
}

## Q 2.2: Plot the sample path; first part only for clarity
par(mfrow=c(2,1))
plot(tvec,X[1,],type="l",xlim=c(0,100),xlab="Time",ylab="Position")
plot(tvec,X[2,],type="l",xlim=c(0,100),xlab="Time",ylab="Velocity")
```





Question 3: Estimate from your simulation the steady-state variance of position Q_t, of velocity V_t, and the covariance between the two. Compare with the solution of the algebraic Lyapunov equation governing the variance. *Note:* In `Matlab`, use built-in `lyap.m`. In `R`, use the function `lyap.R` in `SDEtools`. In `python`, use `scipy.linalg.solve_continuous_lyapunov`.

Solution:

```
Pinf <- P[,,length(tvec)]
Pinf2 <- lyap(A,G%*%t(G))

## Print the empirical covariance and compare with the two solutions of the Lyapunov equation
print(cov(t(X)))
```

```
##             [,1]        [,2]
## [1,] 483.1264334  -0.6406581
## [2,]  -0.6406581 236.8937188
```

```
print(Pinf)
```

```
##               [,1]          [,2]
## [1,]  5.000000e+02 -1.315445e-12
## [2,] -1.315445e-12  2.500000e+02
```

```
print(Pinf2)
```

```
##      [,1] [,2]
## [1,]  500    0
## [2,]    0  250
```

Question 4: The kinetic energy is $\frac{1}{2}mV_t^2$ while the potential energy is $\frac{1}{2}kQ_t^2$. In steady-state, what is the expected kinetic energy and the expected potential energy? *Note:* The result is an example of equipartitioning of energy, a general principle in statistical mechanics, both quantum and classical.

Solution:

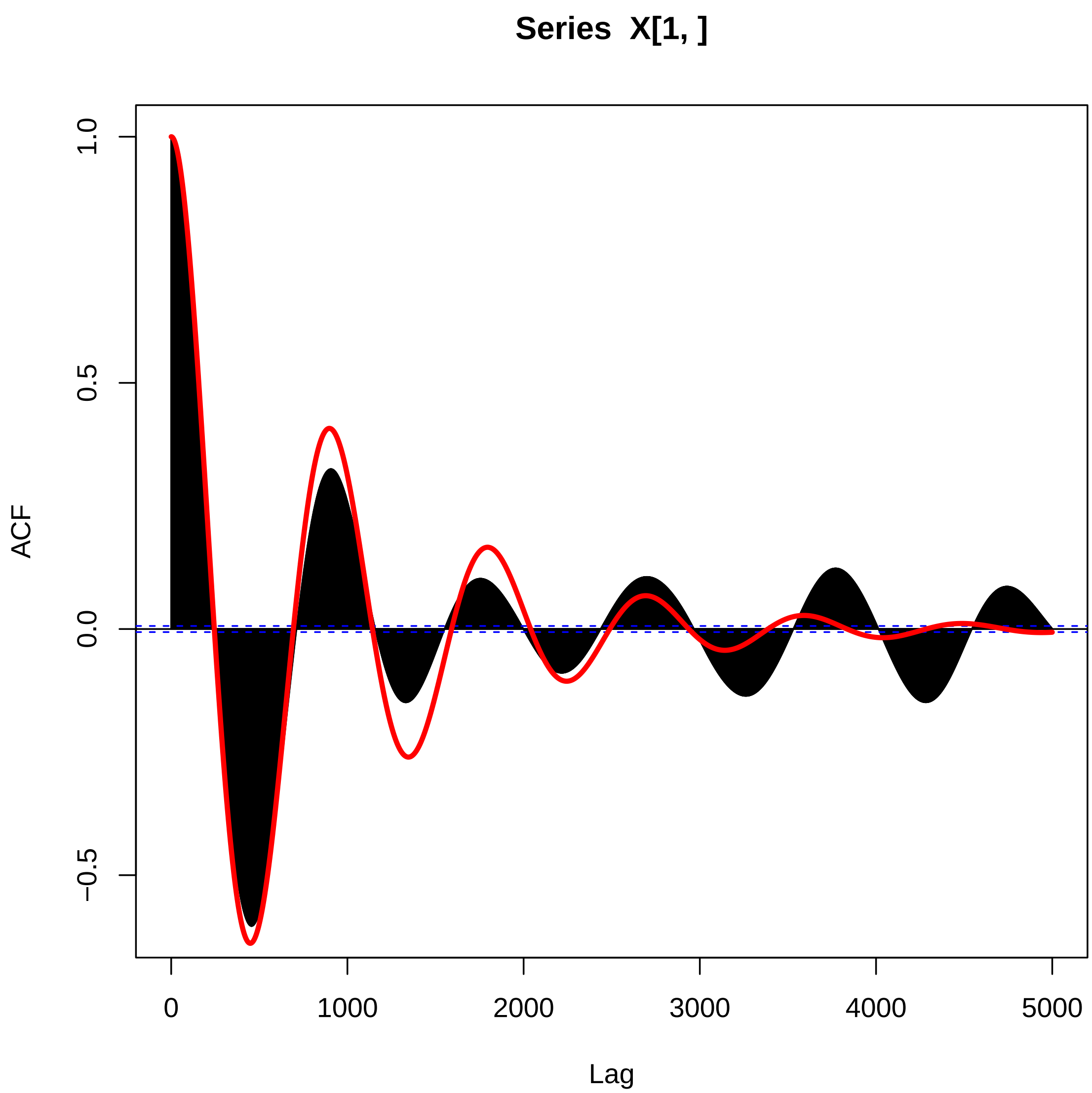
```
print(Ekin <- 0.5*m*var(X[2,]))

## [1] 118.4469

print(Epot <- 0.5*k*var(X[1,]))

## [1] 120.7816

## ... add the analytical predictions:
print(0.5*m*Pinf2[2,2])

## [1] 125

print(0.5*k*Pinf2[1,1])

## [1] 125
```

Note that analytically, the expected kinetic and potential energies are equal, and that they are also quite close, emperically. The empirical result agree reasonably with the analytical result, considering that the simulation is somewhat short compared to the period of oscillations.

Question 5: For the simulation, compute and plot the empirical a.c.f. of $\{Q_t\}$ up to lag 50 s. *Hint:* In `Matlab` and `R`, use `acf`. In `python`, use e.g. `statsmodels.tsa.stattools.acf`. Add to the plot the theoretical prediction.

Solution: We plot empirical a.c.f. of the solution. Note: Lags up to 50 seconds, not 50 time steps!

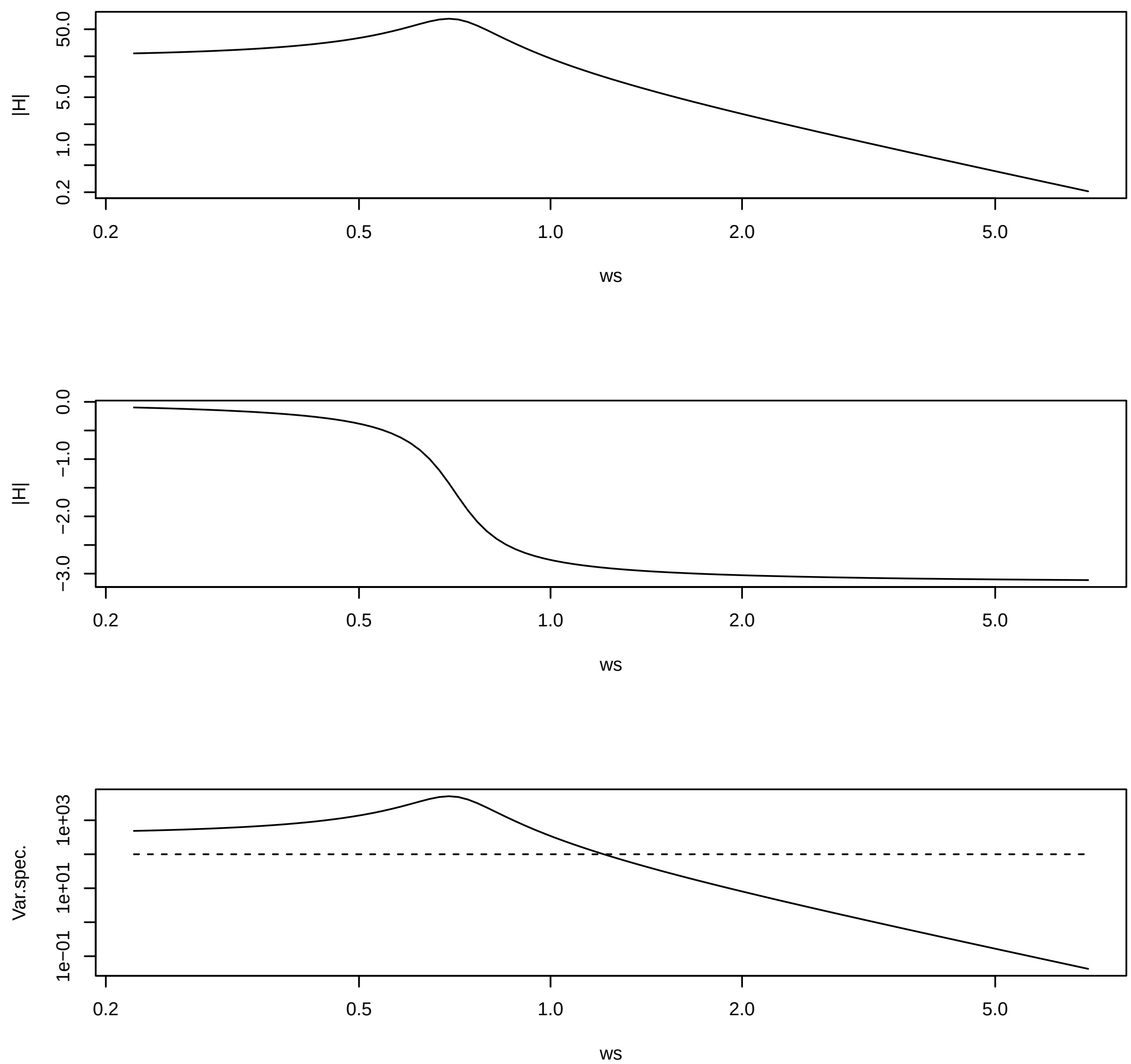
```
acf(X[1,],lag.max=50/dt)

ivec <- seq(0,50/dt,1)
tvec <- ivec * dt

require(Matrix)

## Loading required package: Matrix

rhovec <- sapply(tvec,function(t) (Pinf2 %*% expm(t(A)*t) )[1,1])
lines(ivec,rhovec/rhovec[1],col="red",lwd=3)
```



Question 6: Plot, as a function of the frequency ω, the amplitude and phase of the frequency response from the noise to the position. Plot also the theoretical variance spectrum of the position.

Solution:

```
I <- diag(c(1,1))
H <- function(w) (solve(1i*w*I-A) %*% G)[1]
omegaR <- abs(Im(eigen(A)$values[1]))
ws <- omegaR*10^(seq(-0.5,1,length=101))
Hs <- sapply(ws,H)
par(mfrow=c(3,1))
plot(ws,abs(Hs),log="xy",type="l",ylab="|H|")
plot(ws,Arg(Hs),log="x",type="l",ylab="|H|")
plot(ws,abs(Hs)^2,type="l",log="xy",ylab="Var.spec.")
lines(ws,rep(sigma^2,length(ws)),lty="dashed")
```



Variance in a scalar linear system

The objective of this question is to reproduce figure 5.7 in the book (p. 108). So, consider the scalar linear system

$$\dot{X}_t = -\lambda X_t + \sigma U_t, \quad X_0 = x$$

where $\{U_t : t \geq 0\}$ is Gaussian "white noise", i.e. the formal derivative of standard Brownian motion. For numerical work, we take parameters $x = 1$, $\lambda = 1$, $\sigma = 1$.

Question 7: Plot the mean $\mathbf{E}X_t$ as a function of time.

Solution: The analytical mean is $\mathbb{E}X_t = xe^{-\lambda t}$. See next answer for the plot.

Question 8: Write up the differential Lyapunov equation governing the variance $\mathbf{V}X_t$, and its solution. Add to the plot the mean plus/minus the standard deviation.

Solution: The differential Lyapunov equation is

$$\dot{\Sigma} = -2\lambda\Sigma + \sigma^2$$

and its solution (with $\Sigma(0) = 0$ is the initial condition is deterministic) is

$$\Sigma(t) = \frac{\sigma^2}{2\lambda}(1 - e^{-2\lambda t})$$

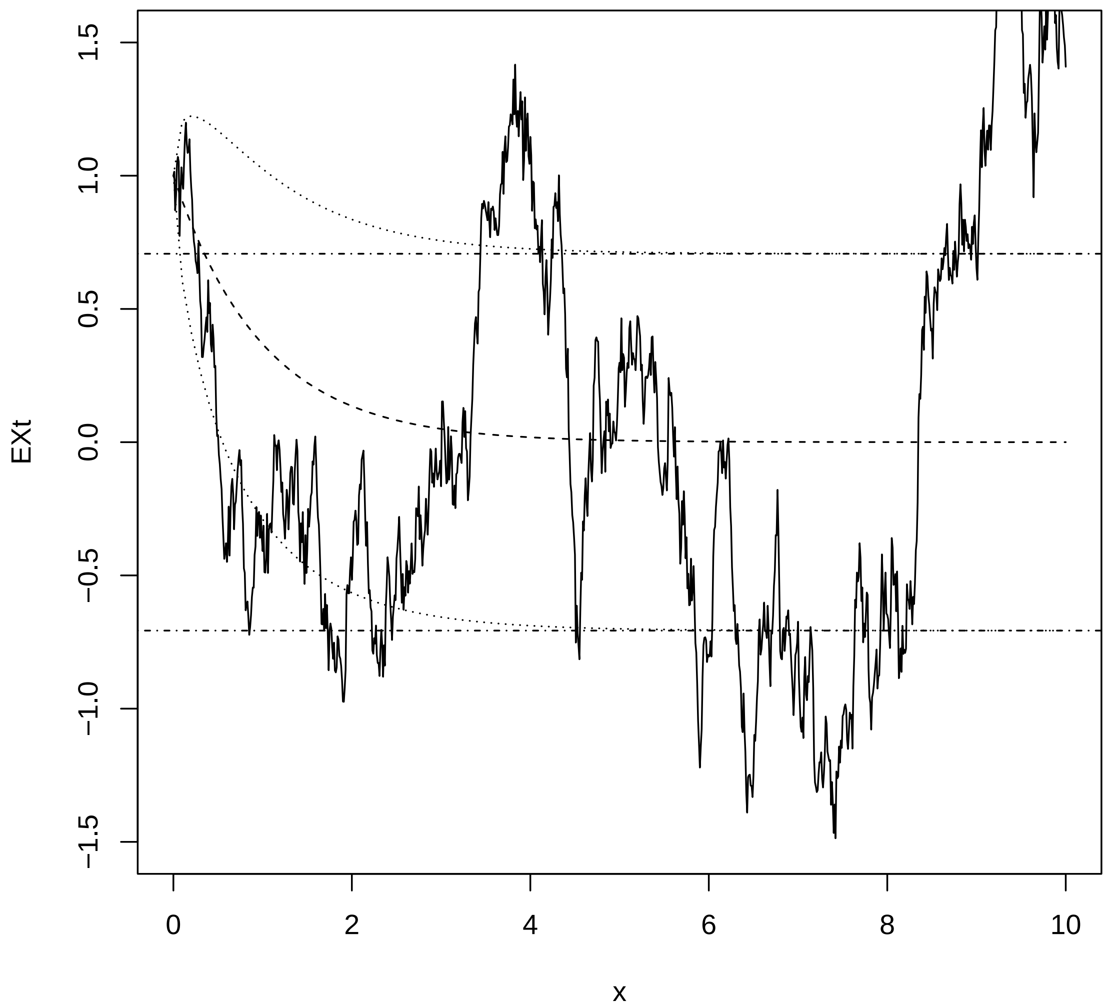
See the next answer for the plot.

Question 9: Assume that $\lambda > 0$, write up the steady-state variance, $\lim_{t\to\infty} \mathbf{V}X_t$. Compute its numerical value and compare with the value of $\Sigma(t)$ for t large.

Solution: If $\lambda > 0$, then $\Sigma(\infty) = \sigma^2/(2\lambda)$. This is both the limit of $\Sigma(t)$ as $t \to \infty$, and the solution to the algebraic Lyapunov equation for the steady-state variance.

The following is a numerical example. It includes also the answer to the following question, where we add a sample path.

```
## Specific examples of system parameters
lambda <- 1
sigma <- 1
x <- 1

EXt <- function(t) x*exp(-lambda*t)

## Differential lyapunov equation
Lyap <- function(V) -2*lambda*V + sigma^2

## Analytical solution of the equation
VXt <- function(t) sigma^2/2/lambda*(1-exp(-2*lambda*t))

## Limit as time goes to infinity, assuming a<0
VXinf <- sigma^2/2/lambda

## Test that it is an equilibrium point for the Lyapunov equation
print(Lyap(VXinf))


## [1] 0


plot(EXt,from=0,to=10,lty="dashed",ylim=c(-1,1)*1.5)
plot(function(t) EXt(t)+sqrt(VXt(t)),from=0,to=10,add=TRUE,lty="dotted")
plot(function(t) EXt(t)-sqrt(VXt(t)),from=0,to=10,add=TRUE,lty="dotted")

abline(h= sqrt(VXinf),lty="dotdash")
abline(h=-sqrt(VXinf),lty="dotdash")

times <- seq(0,10,0.01)
sim <- euler(f=function(x)-lambda*x,g=function(x) sigma,times=times,x0=x)
lines(sim$times,sim$X)
```



Question 10: Simulate a sample path of $\{X_t\}$ and add it to the graph.

Solution: See the code in the previous answer.