

Principles of Robotic Autonomy

Pre-Knowledge Assessment Solutions

Fall 2020

1. (Linear Algebra) What are the eigenvalues of $\begin{bmatrix} 5 & 3 & 1 \\ 0 & -2 & -4 \\ 0 & 0 & 1 \end{bmatrix}$?

(a) 5, -2, 1,

For an upper/lower diagonal matrix, the eigenvalues are the diagonal elements.

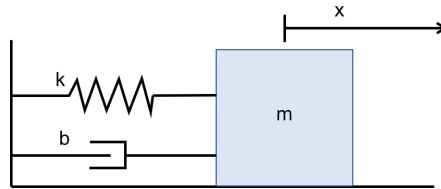


Figure 1: Mass-spring-damper system.

2. (Ordinary Differential Equations) Consider the mass-spring-damper system given in Figure 1. The block has mass $m > 0$ and is rigidly attached to a spring and a damper. Assume the x -coordinate is centered at the equilibrium position. The spring has a spring constant $k > 0$ and provides a force negatively proportional to the positional displacement x . The damper has a damping coefficient $b > 0$ and provides a force negatively proportional to the velocity \dot{x} . The equation of motion is

$$m\ddot{x} + b\dot{x} + kx = 0. \quad (1)$$

For appropriate constants $A, B, \omega > 0, \lambda > 0$, what is the general form of the solution to this second-order ordinary differential equation?

(d) $x(t) = Ae^{-\lambda t} \sin(\omega t) + Be^{-\lambda t} \cos(\omega t)$

Since there is a damper, the solution will decay. Since there is a spring, the solution will be oscillatory.

3. (Ordinary Differential Equations) Given the ODE in Equation (1), let $m = 1$ and using the change of variables $y_1 = x, y_2 = \dot{x}$, which following linear dynamical system is equivalent?

(e) $\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k & -b \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$

$\dot{y}_1 = y_2$ and $\ddot{x} = \dot{y}_2 = -by_2 - ky_1$.

4. (Vector Calculus) Let $f(x, y) = 2xy + 6x^2y - 3xy^2$. What is the Hessian of $f(x, y)$?

(a) $H_f(x, y) = \begin{bmatrix} 12y & 2 + 12x - 6y \\ 2 + 12x - 6y & -6x \end{bmatrix}$

$f_{xx} = 12y, f_{yy} = -6x, f_{xy} = 2 + 12x - 6y$. The Hessian has the form $\begin{bmatrix} f_{xx} & f_{xy} \\ f_{xy} & f_{yy} \end{bmatrix}$

5. (Probability) Bayes' rule says, $P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}$. You tested positive for a disease, and the test is 80% accurate (i.e., the probability of testing positive given that you have the disease is 0.80, as is the probability of testing negative given that you do not have the disease). This disease strikes one in five people. What is the probability of you having the disease, given that your test was positive?

(b) $\frac{1}{2}$

Suppose $Y \in \{disease, healthy\}$, and $X \in \{positive, negative\}$. We want $P(disease|positive)$. By Bayes rule,

$$\begin{aligned}
 P(disease|positive) &= \frac{P(positive|disease)P(disease)}{P(positive)} \\
 &= \frac{P(positive|disease)P(disease)}{P(positive|disease)P(disease) + P(positive|healthy)P(healthy)} \\
 &= \frac{0.8 \times 0.2}{0.8 \times 0.2 + 0.2 \times 0.8} \\
 &= \frac{1}{2}
 \end{aligned}$$

6. (Programming: Recursion¹) Using your favorite programming language (e.g., Python, Julia, C++), write a recursive function named `even_factorial` that accepts an integer n as a parameter and returns the even factorial of n . An even factorial of an integer is defined as the product of all even integers from 1 through that integer inclusive. For example, the call of `even_factorial(7)` should return $2 \times 4 \times 6$, or 48. The factorial of 0 and 1 are defined to be 1. You may assume that the value passed is non-negative and that its factorial can fit in the range of type `int`. You may define a helper function if you like.

Do not use loops or auxiliary data structures to solve the problem recursively. Do not declare any global variables.

```

# Python
def even_factorial_helper(n, ans):
    if n <= 1:
        return ans
    elif n % 2 == 1:
        return even_factorial_helper(n-1, ans)
    else:
        return even_factorial_helper(n-1, ans*n)

def even_factorial(n):
    return even_factorial_helper(n, 1)

```

¹Taken from StepByStep <https://www.codestepbystep.com/>