COMPSCI 371D Homework 0 (Prerequisites)

Collaborators

- Chavez Cheong
- Khoo Qi Xuan
- Jake Heller

Part 1: Algebra and Multivariate Calculus

Problem 1.1

$g = \nabla V(x) = \begin{bmatrix} (1+k)x_0 - kx_1 \\ (1+k)x_1 - kx_0 \end{bmatrix}$

$H(x) = \begin{bmatrix} k+1 & -k \\ -k & k+1 \end{bmatrix}$x} $

Problem 1.2

$\lambda_0 = 1, u_0 = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}$

$\lambda_1 = 2k+1, u_1 = \begin{bmatrix} \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}$

Problem 1.3

We see that $\lambda_0 = 1 > 0$ and $\lambda_1 = 2k+1 > 1 > 0$ since $k > 0$, and thus we have that all eigenvalues of H are real. Since H is also a symmetric matrix, H is positive definite.

Problem 1.4

$U = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$

$\Lambda = \begin{bmatrix} 1 & 0 \\ 0 & 2k+1 \end{bmatrix}$

Problem 1.5

As U is an orthonormal vector, we know that $U^{-1} = U^T$.

Hence, $U^{-1} = U^T = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$

Problem 1.6

$HU = U\Lambda$

Multiplying both sides by U^{-1}

$HUU^{-1} = U\Lambda U^{-1}$

$H = U\Lambda U^{-1}$

From Problem 1.5, we know that U is an orthonormal vector, hence $U^{-1} = U^T$

$H = U\Lambda U^T$

Hence, we see that $W = U$ and $D = \Lambda$.

Problem 1.7

$y(t) = a\cos(\omega t + \phi)$

$\dot{y}(t) = -a\omega\sin(\omega t + \phi)$

$\ddot{y}(t) = -a\omega^2\cos(\omega t + \phi)$

We have $dy + \ddot{y} = 0$

$ad\cos(\omega t + \phi) - a\omega^2\cos(\omega t + \phi) = 0$

$a(d - \omega^2)\cos(\omega t + \phi) = 0$

$\therefore \omega = \sqrt{d}$

Problem 1.8

$Dy - \ddot{y} = 0$

$\begin{bmatrix} 1 & 0 \\ 0 & 2k+1 \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \end{bmatrix} - \begin{bmatrix} \ddot{y}_0 \\ \ddot{y}_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

We get the system of equations:

$y_0 - \ddot{y}_0 = 0$

$(2k+1)y_1 - \ddot{y}_1 = 0$

Following Problem 1.7:

$d_0 = 1$

$\omega_0 = \sqrt{d_0} = \sqrt{1} = 1$

$d_1 = 2k + 1$

$\omega_1 = \sqrt{d_1} = \sqrt{2k+1}$

Hence, we have:

$y_0(t) = a_0 \cos(t + \phi_0)$

$y_1(t) = a_1 \cos(\sqrt{(2k+1)}t + \phi_1)$
We know that $y = W^T x$. From Problem 1.6, we know that $W = U$, and hence $y = U^T x$. Multiplying both sides of the equation by U,

$Uy = UU^T x$

Since U is orthonormal, $UU^T = UU^{-1} = I$

$x = Uy$

$$\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} y_0 - \frac{1}{\sqrt{2}} y_0 \\ \frac{1}{\sqrt{2}} y_0 + \frac{1}{\sqrt{2}} y_0 \end{bmatrix}$$

Hence, we have

$x_0(t) = \frac{1}{\sqrt{2}} a_0 \cos(t + \phi_0) - \frac{1}{\sqrt{2}} a_1 \cos(\sqrt{(2k+1)}t + \phi_1)$

$x_1(t) = \frac{1}{\sqrt{2}} a_0 \cos(t + \phi_0) + \frac{1}{\sqrt{2}} a_1 \cos(\sqrt{(2k+1)}t + \phi_1)$

Problem 1.9

In [1]:

```
from matplotlib import pyplot as plt
import numpy as np
%matplotlib inline
```

In [2]:

```
def springs(k, a, phi, t):
    x_0 = (np.cos(t + phi[0]) * a[0] - np.cos((2*k+1)*t + phi[1]) * a[1]) / np
    x_1 = (np.cos(t + phi[0]) * a[0] + np.cos((2*k+1)*t + phi[1]) * a[1]) / np
    return np.vstack((x_0,x_1))
```

In [3]:

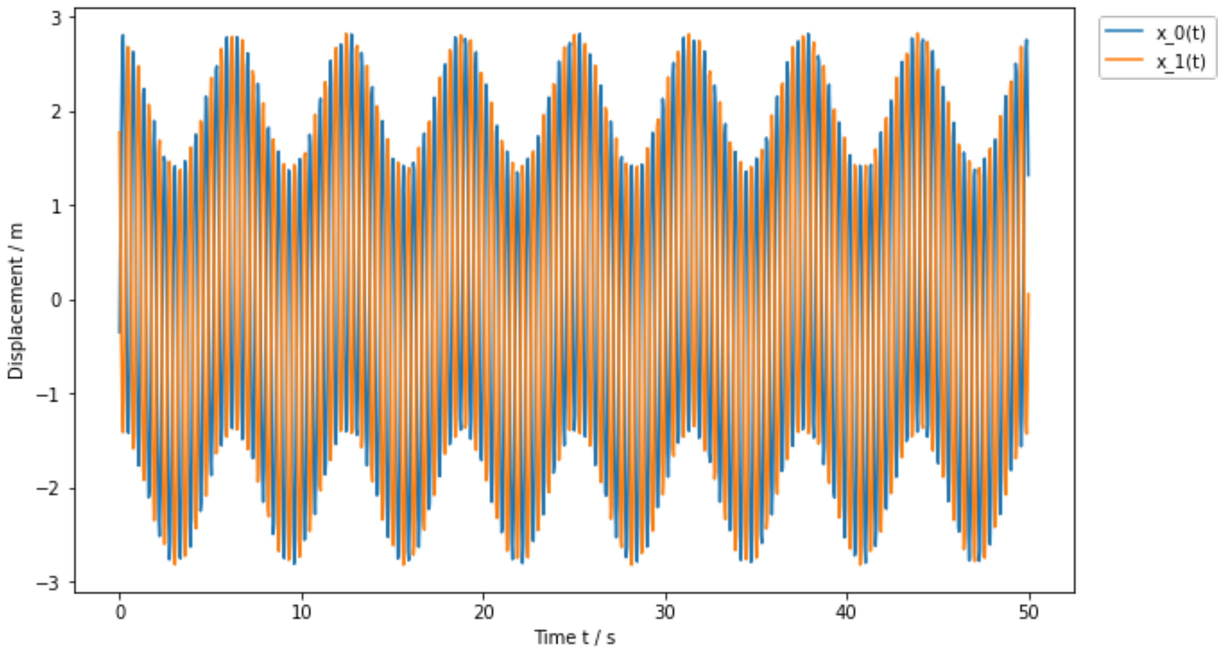
```
t = np.linspace(0,50,1000)
a = (1,3)
phi = (0, np.pi/3)
k = 5
```

In [4]:

```
plt.figure(figsize=(10,6))
points = springs(k,a,phi,t)

x_0 = points[0]
x_1 = points[1]

plt.plot(t,x_0,label = "x_0(t)")
plt.plot(t,x_1,label = "x_1(t)")
plt.legend(bbox_to_anchor=(1.15,1), loc="upper right")
plt.xlabel('Time t / s')
plt.ylabel('Displacement / m')
plt.show()
```

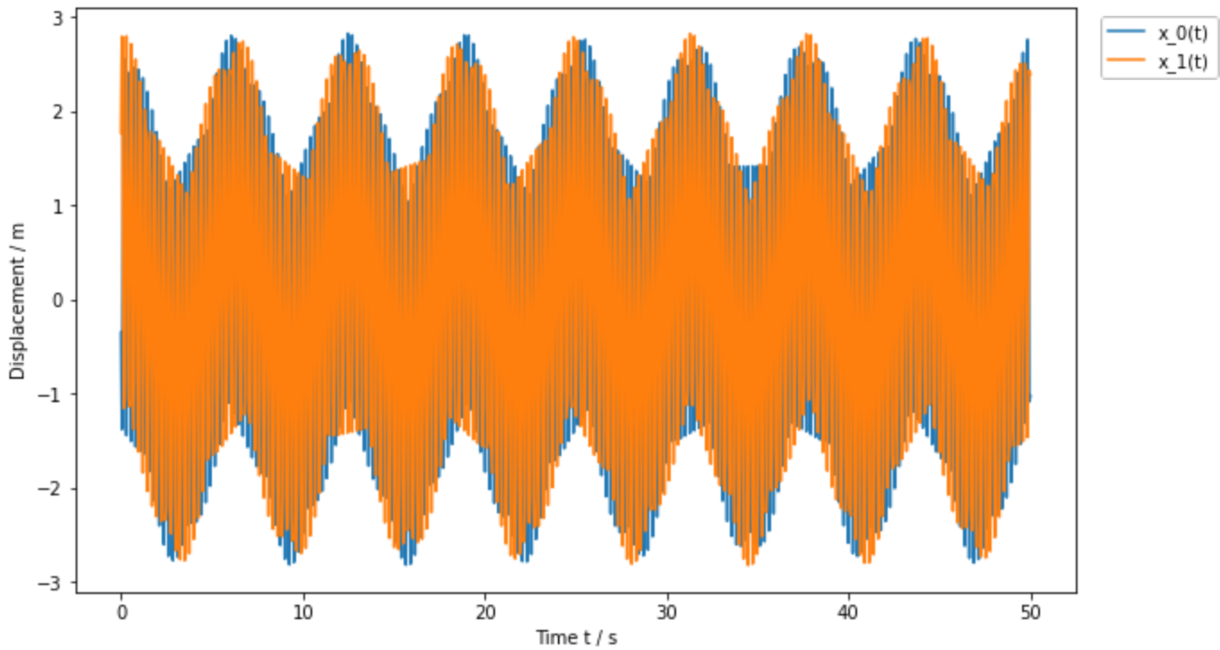


In [5]:

```
k = 50
points = springs(k,a,phi,t)

x_0 = points[0]
x_1 = points[1]

plt.figure(figsize = (10,6))
plt.plot(t,x_0,label = "x_0(t)")
plt.plot(t,x_1,label = "x_1(t)")
plt.legend(bbox_to_anchor=(1.15,1), loc="upper right")
plt.xlabel('Time t / s')
plt.ylabel('Displacement / m')
plt.show()
```



Part 2: Probability

Problem 2.1

$p(k,n|h) = \binom{n}{k} h^k (1-h)^{n-k}$

$p(0,n|h) = \binom{n}{0} (1-h)^n = (1-h)^n$

$p(n,n|h) = \binom{n}{n} (h)^n = (h)^n$

Problem 2.2

Let $p_h(x)$ represent the probability density function of h.

Since h is uniformly distributed from 0 to 1.

$p_h(x) = 1$, for $x \in [0,1]$ and 0 elsewhere.

$p(0,n) = \int_0^1 p(0,n|h) p_h dh = \int_0^1 (1-h)^n dh = \frac{1}{n+1}$

$p(n,n) = \int_0^1 p(n,n|h) p_h dh = \int_0^1 h^n dh = \frac{1}{n+1}$

Problem 2.3

Using Bayes' Theorem: $p_h(x|E_n) = \frac{p_h(E_n|x)p_h(x)}{p_h(E_n)} = \frac{x^n}{\frac{1}{n+1}} = x^n(n+1)$

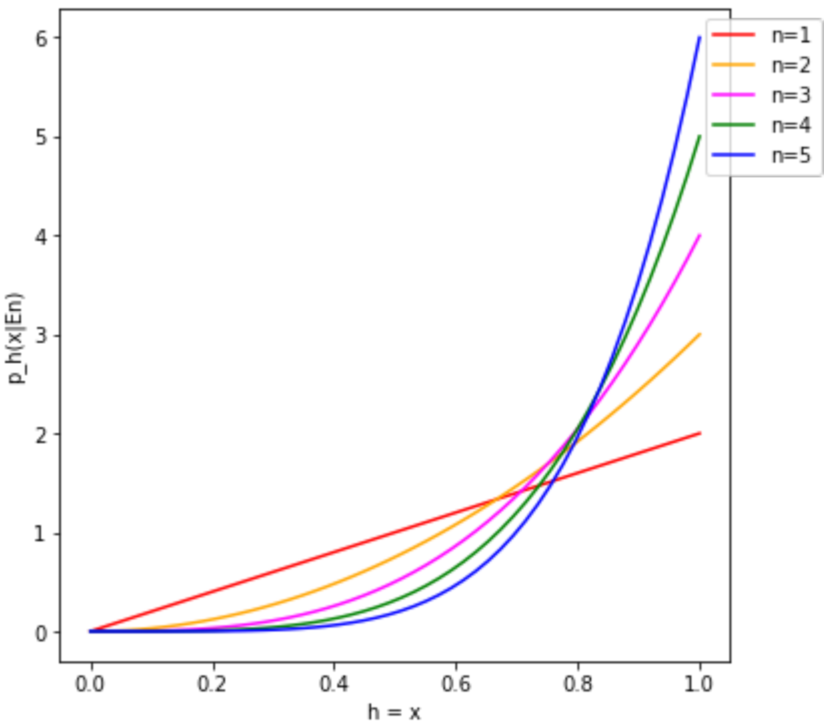
In [6]:

```
x = np.linspace(0,1,50)

ret = []
for n in range(1,6):
    ret.append((n+1)*x**n)

plt.figure(figsize = (6,6))
plt.plot(x, ret[0], color='red', label='n=1')
plt.plot(x, ret[1], color='orange', label='n=2')
plt.plot(x, ret[2], color='magenta', label='n=3')
plt.plot(x, ret[3], color='green', label='n=4')
plt.plot(x, ret[4], color='blue', label='n=5')
plt.legend(bbox_to_anchor=(1.15,1), loc="upper right")
plt.xlabel('h = x')
plt.ylabel('p_h(x|En)')

plt.show()
```



Problem 2.4

In [7]:

```
def sample(n, m=100000):
    rng = np.random.default_rng()
    h = rng.uniform(0,1,100000)
    b = rng.binomial(n, h)
    ret = []
    for i in range(len(b)):
        if b[i] == n:
            ret.append(h[i])
    return ret
```

In [8]:

```
x = np.linspace(0,1,50)

for n in [1,3,5]:
    plt.figure(figsize = (3,3))
    plt.plot(x,  (n+1)*x**n,   label=f"{n}")
    plt.hist(sample(n),x,density=True)
    plt.title(f"n = {n}")
    plt.show()
```