Name: Ajayrangan Kasturirangan
NetID: ak10196

Complex Urban Systems
Assignment 1

Question 1

Given a utility function U(x,y)=2x+y. Let the price of good x by 2 per unit and the price of good y be 4 per unit. Let the income of the individual be w = 30 units. Compute the quantities of goods x and y that maximize utility with respect to the budget constraint of the individual and the corresponding maximal utility. Please verify your extremum is in fact a maximum by studying the bordered Hessian.

$U(x, y) = 2\sqrt{x} + \sqrt{y}$, the price of good x as 2 per unit and the price of good y as 4 per unit. The income of the individual is $w = 30$ units. The Lagrangian function is given by:

$$L(x, y, \lambda) = 2\sqrt{x} + \sqrt{y} - \lambda\,(2x + 4y - 30)$$

Taking the partial derivatives and setting them to zero, we get:

$$\frac{\partial L}{\partial x} = \frac{1}{\sqrt{x}} - 2\lambda = 0 \tag{1}$$

$$\frac{\partial L}{\partial y} = \frac{1}{\sqrt{y}} - 4\lambda = 0 \tag{2}$$

$$\frac{\partial L}{\partial \lambda} = 2x + 4y - 30 = 0 \tag{3}$$

```python
import sympy as sp
x, y, lamda = sp.symbols('x y lamda')
L = 2*sp.sqrt(x) + sp.sqrt(y) - lamda*(2*x + 4*y - 30)
dL_dx = sp.diff(L, x)
print('dl/dx=', dL_dx)
dL_dy = sp.diff(L, y)
print('dl/dy=', dL_dy)
dL_dlamda = sp.diff(L, lamda)
print('dl/dlambda=', dL_dlamda)
sol = sp.solve([dL_dx, dL_dy, dL_dlamda], (x, y, lamda))
print('Solutions for x,y,lambda', sol)
```

```
dl/dx= -2*lamda + 1/sqrt(x)
dl/dy= -4*lamda + 1/(2*sqrt(y))
dl/dlambda= -2*x - 4*y + 30
Solutions for x,y,lambda [(40/3, 5/6, sqrt(30)/40)]
```

This shows that the solution for x and y is 40/3 and 5/6 respectively.
Now let us verify using bordered hessian that this is indeed the maximum.

```
H = sp.Matrix([[sp.diff(dL_dx, x), sp.diff(dL_dx, y), sp.diff(dL_dx, lamda)],
               [sp.diff(dL_dy, x), sp.diff(dL_dy, y), sp.diff(dL_dy, lamda)],
               [sp.diff(dL_dlamda, x), sp.diff(dL_dlamda, y), sp.diff(dL_dlamda, lamda)
                   ]])
```

$$\begin{bmatrix} -\frac{1}{2x^{\frac{3}{2}}} & 0 & -2 \\ 0 & -\frac{1}{4y^{\frac{3}{2}}} & -4 \\ -2 & -4 & 0 \end{bmatrix}$$

From the above bordered hessian, it is clear that the determinant is positive by observation and hence the solution obtained is indeed a maximum.

Question 2

Consider the set of Figure 2.4 in your textbook, and simulate the model in the short run, that is, study the relationship between the nominal wage difference (yw1 yw2) and f1 for the same cases of figure 2.4. Generate six equivalent panels with nMF = 0.3 and T = 1.1,1.7, and 2.2 and nMF = 0.4 and T = 1.3,2.1, and 4.1 and always sigma = 4. Qualitatively compare your results with figure 2.4 and comment on them.

Answer:

Using Fsolve in Python, I have implemented the following code to solve the system of equations that is:

$$
\begin{aligned}
\mathrm{z}_{11} &= \left(\frac{l_1}{l_2}\left(\frac{w_1\tau}{w_2}\right)^{1-\sigma}\right)\\
\mathrm{z}_{12} &= \left(\frac{l_1}{l_2}\left(\frac{w_1}{w_2\tau}\right)^{1-\sigma}\right)\\
\mathrm{w}_1 l_1 &= \mu\left(\frac{z_{11}}{1+z_{11}}y_1 + \frac{z_{12}}{1+z_{12}}y_2\right)\\
\mathrm{w}_2 l_2 &= \mu\left(\frac{1}{1+z_{11}}y_1 + \frac{1}{1+z_{12}}y_2\right)\\
\mathrm{y}_1 &= \left(\frac{1-\mu}{2} + w_1 l_1\right)\\
\mathrm{y}_2 &= \left(\frac{1-\mu}{2} + w_2 l_2\right)
\end{aligned}
$$

for when

$\mu = [0.3, 0.4, 0.3, 0.4, 0.3, 0.4]$

$1/\tau = T = [1.1, 1.3, 1.7, 2.1, 2.2, 4.1]$

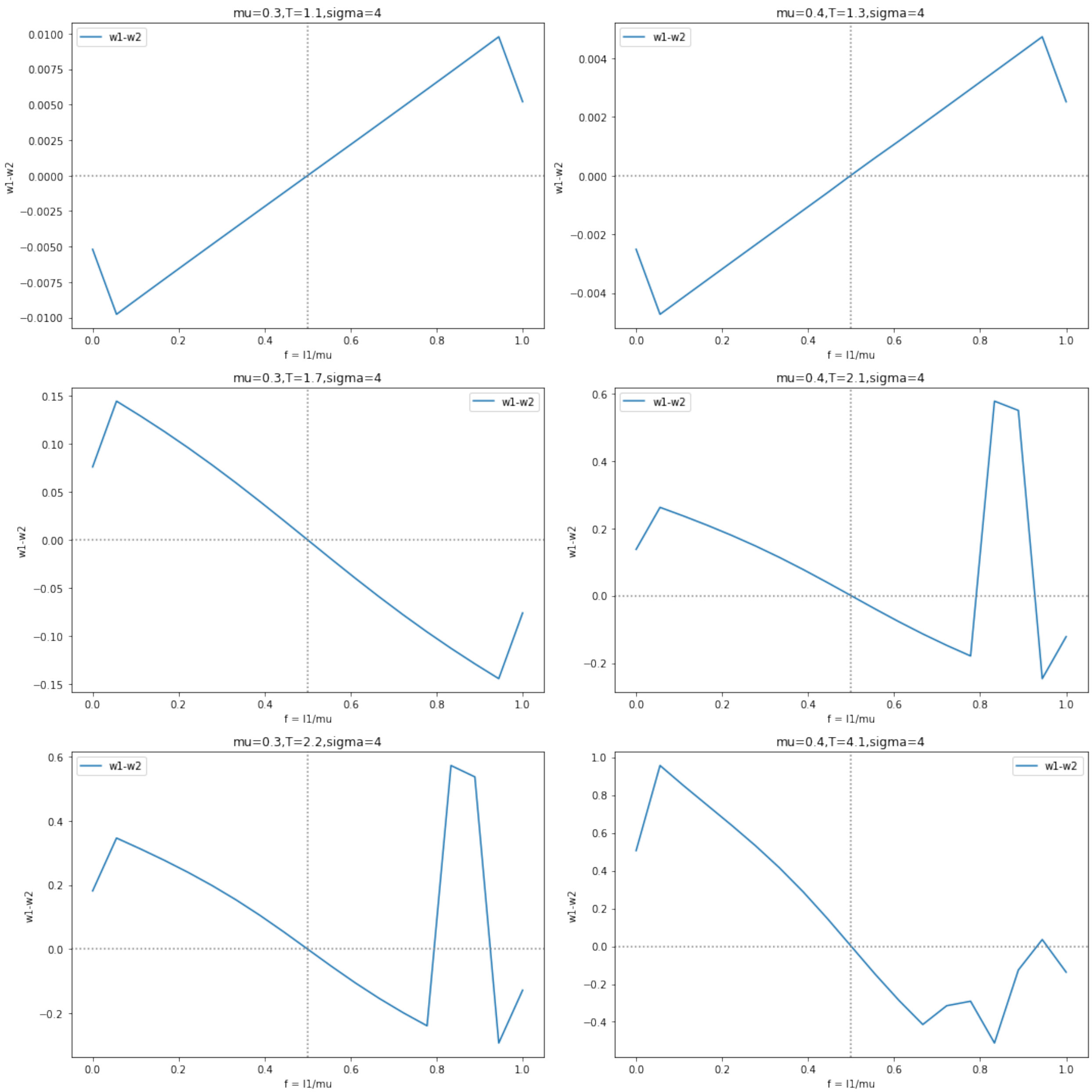
$\sigma = 4$

```python
import numpy as np
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import fsolve
from scipy.optimize import minimize
import warnings

warnings.filterwarnings("ignore", category=RuntimeWarning)


def equations(variables, mu, tau, sigma, l1, l2):
    w1, w2, z11, z12, y1, y2 = variables

    f1 = z11 - ((l1/l2)*(((w1*tau)/w2)**(1-sigma)))
    f2 = z12 - ((l1/l2)*(((w1)/(w2*tau))**(1-sigma)))
    f3 = w1*l1 - (mu*(((z11/(1+z11))*y1)+((z12/(1+z12))*y2)))
    f4 = w2*l2 - (mu*(((1/(1+z11))*y1)+((1/(1+z12))*y2)))
    f5 = y1 - (((1-mu)/2) + (w1*l1))
    f6 = y2 - (((1-mu)/2) + (w2*l2))
    return np.array([f1, f2, f3, f4, f5, f6])
```

```python

def solve_eqns(mu, tau, sigma, l1, l2,x0):
    args = (mu, tau, sigma, l1, l2)
    solution = fsolve(equations, x0, args=(mu, tau, sigma,l1,l2),xtol=1e-4, maxfev=5000)
    return solution

def moving_average(data, window_size):
    moving_averages = []
    for i in range(len(data) - window_size + 1):
        window = data[i:i+window_size]
        window_average = sum(window) / window_size
        moving_averages.append(window_average)
    return moving_averages, len(moving_averages)


def plot_graph(mu, tau, sigma, plot_index,p,q):
    x = np.linspace(1e-10, 1 - 1e-10, 20)
    w_values = []
    for l1_by_mu in x:
        l1 = l1_by_mu * mu
        l2 = mu-l1
        x0 = np.array([l1/l2,l1/l2,l1/l2,l1/l2,l1/l2,l1/l2])

        solution = solve_eqns(mu, tau, sigma,l1,l2,x0)
        w1_w2 = solution[0]-solution[1]

        w_values.append(w1_w2)



    #Percentile Filter
    lower_bound = np.percentile(w_values, 0)
    upper_bound = np.percentile(w_values, 100)
    w_values = [value for value in w_values if lower_bound <= value <= upper_bound]


    w_values,n = moving_average(w_values,2)
    x_ = np.linspace(0, 1, n)



    #Subplots
    ax = plt.subplot(p, q, plot_index)
    ax.plot(x_, w_values, label='w1-w2')
    ax.axhline(y=0, linestyle='dotted', color='grey')
    ax.axvline(x=0.5, linestyle='dotted', color='grey')
    ax.set_title('mu={},T={},sigma={}'.format(mu,1/tau,sigma))
    ax.set_xlabel('f = l1/mu')
    ax.set_ylabel('w1-w2')
    ax.legend()



#Defining MU and T
```

```
MU = [0.3,0.4,0.3,0.4,0.3,0.4]
T =  [1.1,1.3,1.7,2.1,2.2,4.1]

plt.figure(figsize=(15,15))
for i, (mu, t) in enumerate(zip(MU, T)):
    tau = 1/t
    sigma = 4
    plot_graph(mu, tau, sigma, i+1,3,2)

plt.tight_layout()
plt.show
```

Giving the following output:



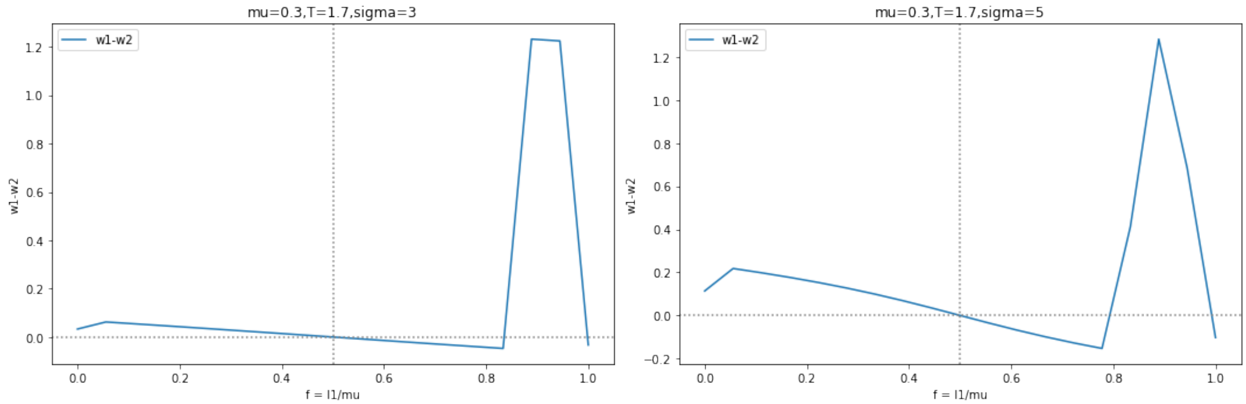
We can see the trend is correct (Similar to Long run) for all plots in the middle where the trend shows the increase or decrease of w1-w2 as f=L1/mu increases. But towards the ends(0 ans 1) and between 0.8 and 1, the solution is not optimized.

Question 3

Consider again the set of figures 2.4 in your textbook, write your own computer code to study workers' agglomeration and dispersal in the core-periphery model for nMF = 0.3 and T = 1.7. For this case, examine the role of s by studying s = 3 and =5; please comment on your results.

Answer:

```
MU = [0.3,0.3]
T =  [1.7,1.7]
sigma = [3,5]

plt.figure(figsize=(15,5))
for i, (mu, t,sig) in enumerate(zip(MU, T,sigma)):
    tau = 1/t
    sigma = 4
    plot_graph(mu, tau, sig, i+1,1,2)

plt.tight_layout()
plt.show
```



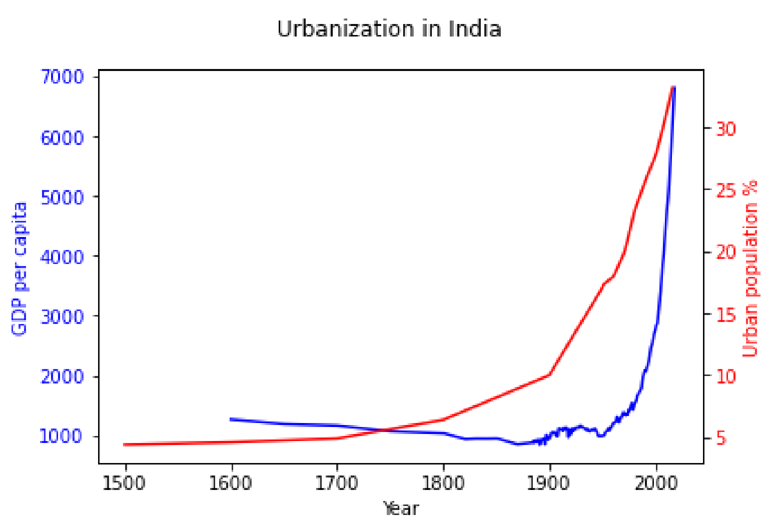
Assuming a similar optimization error between 0.8 and 1, we can observe the remaining graph having a declining trend and as sigma increases, we can see the value of w1-w2 decreases more sharply. In other words sigma increases and rate of change of w1-w2 increases(negatively)

Question 4

Pick your country of birth or any country you like from https://ourworldindata. org/urbanization in the section “Populations urbanize as they get richer.” After downloading the csv file, plot the two time-series of “GDP per capita” and “Share of population living in urban areas” for the entire range of available data.

Answer:

```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

urbanization = pd.read_csv('urbanization-vs-gdp.csv')
india_urbanization = urbanization[urbanization['Entity'] == 'India']
india_urbanization = india_urbanization.sort_values('Year')

# Create the plot with two lineplots and different colors
fig, ax1 = plt.subplots()

sns.lineplot(x = 'Year',y = 'GDP per capita',data=india_urbanization, color='blue', ax=ax1)
ax1.set_ylabel('GDP per capita', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')

ax2 = ax1.twinx()

sns.lineplot(x = 'Year',y = 'Urban population (%) long-run to 2016 (OWID)',data=india_urbanization, color='red', ax=ax2)
ax2.set_ylabel('Urban population %', color='red')
ax2.tick_params(axis='y', labelcolor='red')


plt.suptitle('Urbanization in India')

plt.show()
```



Question 5

Consider the model of urban economics in Section 2.2.5. Given a utility function of the form

$$U(c, af) = \sqrt{c}af \tag{4}$$

derive the functional form of the price of housing pf as a function of R that maximizes utility everywhere, equally in the city. Plot your expression for y = 10, 000, cT0 = 100, and pf (0) = 200
Answer: To maximize utility everywhere, we equate this to zero according to the model

$$\frac{\partial U}{\partial C} * \frac{\partial C}{\partial a_f} + \frac{\partial U}{\partial a_f} = 0 \tag{5}$$

That gives us

$$\frac{af}{2 * \sqrt{c}} * (-p_f) + \sqrt{c} = 0 \tag{6}$$

where c is

$$c = y - c_T o * R - p_f * a_f \tag{7}$$

Since we have pf(0) = 200, y =10000, cT0 = 100 - we can solve for a_f

```python
import sympy as sp
af, c = sp.symbols('a_f c')
eq1 = (af/(2*sp.sqrt(c)))*(-200) + sp.sqrt(c)
eq2 = c - (10000 - 200*af)
af_solved = sp.solve([eq1, eq2], [af, c])
sp.pprint(af_solved)
```

```
{a_f: 100/3, c: 10000/3}
```

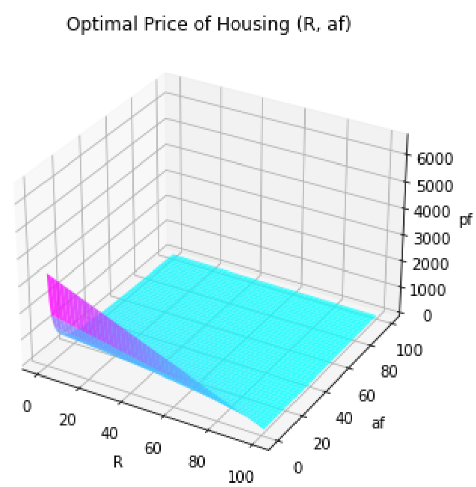
Substituitng and simplifying, we get

$$pf = 200 - 2 * R \tag{8}$$

or

$$pf = \frac{20000 - 200 * R}{3 * af} \tag{9}$$

Which gives a graph like



Question 6

Verify, if possible, equation 2.26 in your textbook in general and with respect to the problem above specifically.

Answer:

Equation 2.26:

$$\frac{\partial a_f}{\partial R} = \frac{\partial a_f}{\partial p_f} * \frac{\partial p_f}{\partial R} > 0 \tag{10}$$

$$a_f = frac20000 - 200 * R3 * p_f \tag{11}$$

$$\frac{\partial a_f}{\partial R} = \frac{200 * R - 20000}{3 * p_f^2} * \frac{-200}{3 * a_f} \tag{12}$$

Further equating to zero and applying inequality

$$(200R - 20000) * (-200) > 0 \tag{13}$$

Hence the equation hold true when

$$R < 100 \tag{14}$$

Question 7

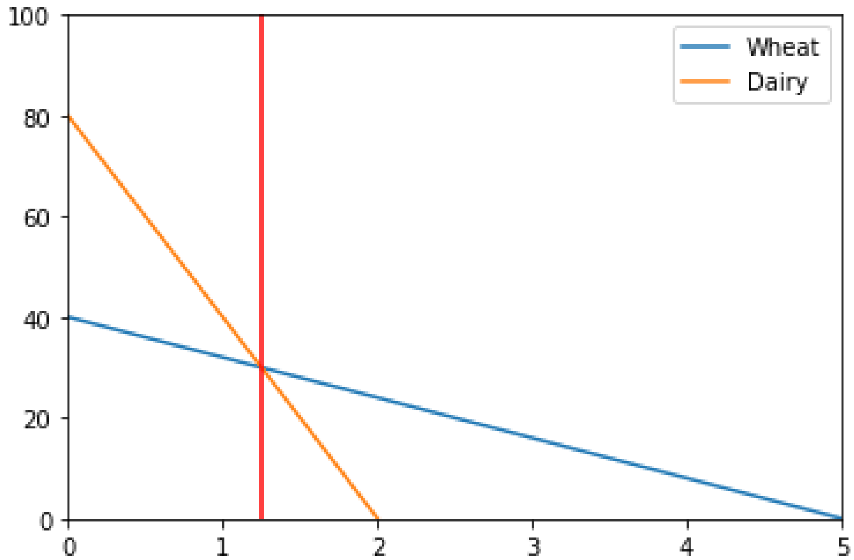
Consider the von Thunen's model for the case of two products only, wheat and dairy. Assume that: i) the yield of wheat is 80 bushels per acre and that of dairy is 40 gallons per acre, ii) the market price after production costs for a bushel of wheat is $0.5andforagallonofmilkis2$, and iii) the transportation costs per bushel/mile of wheat is $0.1andforgallon/milefordairyis1$. What is the rent income at the marketplace for the two products? What is the optimal range for each of them?

Answer: Von thunens equation:

$$Y_i = Q_i\left[(P_i - C_{Pi}) - C_T \cdot R\right] \tag{15}$$

Rent income for wheat =
= 0.5 x 80 - 80 * 0.1 x R = 40 - 8*R
Rent income for dairy =
= 2 x 40 - 40 * 1 x R = 80 - 40*R

```python
import matplotlib.pyplot as plt
import numpy as np
def eq1(R):
    return 40 - 8*R
def eq2(R):
    return 80 - 40*R
R_values = np.linspace(0, 5, 100)
plt.plot(R_values, eq1(R_values), label='Wheat')
plt.plot(R_values, eq2(R_values), label='Dairy')
plt.axvline(x=1.25, color='red')
plt.xlim([0, 5])
plt.ylim([0, 100])
plt.legend()
plt.show()
```

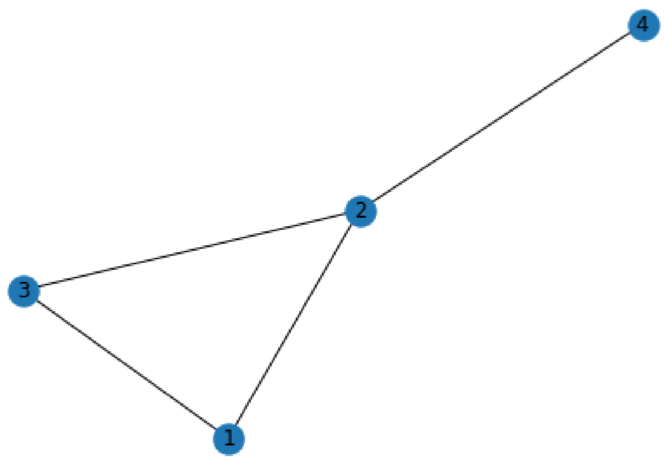


Solving both the equations, we can see that the range for wheat is above 1.25 miles while the range for dairy is below 1.25 miles.

Question 8

Consider the simple four-node network with adjacency matrix A Compute the degree of each node, the degree variance of the network, and the total connectivity. Is the network connected? How many paths from node 1 to 4, and what is the distance of the shortest one?

$A =$				
	0	1	1	0
	1	0	1	1
	1	1	0	0
	0	1	0	0



a. The degree of Node 1 is 2, Node 2 is 3 Node 3 is 2, Node 4 is 1. That is [2,3,2,1]

b. Mean degree is $\frac{2+3+2+1}{4} = 2$. So variance is $\frac{(2-2)^2+(2-3)^2+(2-2)^2+(2-1)^2}{4} = \frac{5}{4}$.

c. Total connectivity is 4

d. The Network is connected as all nodes are connected and no node is isolated.

e. There are 2 paths from 1 to 4, that is 1-3-2-4 and 1-2-4. 1-2-4 being the shortest with a distance of 2 edges.

Question 9

Read the paper: Oliveira, M. More crime in cities? On the scaling laws of crime and the inadequacy of per capita rankings—a cross-country study. Crime Sci 10, 27 (2021) https://doi.org/10.1186/s40163-021-00155-8. Write a 200-word summary of what you consider the major findings of the study.

Answer:

The study checked how crime and population are related in cities across 12 countries. They ranked cities for crime using per capita rates, and the results were very interesting. They found out that in most countries, theft grows faster than the population size, that is it moves superlinearly. In fact, they discovered that in 9 out of 11 countries, beta is above 1 where Y = N**beta. But when it comes to burglary, it increases in a linear fashion, and they couldn't reject this hypothesis in 7 out of 10 countries. The study also found that using per capita rates to rank cities for crime may not always show a real increase in crime with population size. This means that if we use linear assumptions, it could give wrong rankings, and we need to be careful when using such rankings. The study says we must check whether the linearity is plausible before using these rankings. Overall, the study shows us how crime and population are connected in cities.

Question 10

For the same paper above, download the US data from the FBI link in the paper and compute scaling exponent for violent crime or any other variable reported in the data that you like. Please comment on your results, anything interesting?

```python
df = pd.read_excel('Table_8_Offenses_Known_to_Law_Enforcement_by_State_by_City_2019.xls',
     header=3, usecols="C:D")
plt.scatter(df['Population'], df['Violent\ncrime'])
plt.xscale('log')
plt.yscale('log')
plt.xlabel('Population (log scale)')
plt.ylabel('Violent crime (log scale)')
plt.title('Scatter Plot of Population vs Violent Crime')
plt.grid(True)
plt.show()
```