Solutions :-

Q1) Consider the following usability equations:

$$U_1 = \frac{1}{average\_time} + \frac{1}{average\_errors} + \frac{1}{average\_crashes} \quad (1)$$

$$U_2 = \frac{1}{0.5 \times average\_time} + \frac{1}{100 \times average\_trials} + \frac{1}{70 \times average\_crashes} \quad (2)$$

(a) Given a three systems where $U_{s1} < U_{s3} < U_{s2}$. If the usablity of these system is computed using equation (1), which system is more usable. Justify your answer. (10 pts)

(b) Given a three systems where $U_{s3} < U_{s2} < U_{s1}$. If the usablity of these system is computed using equation (1), which system is more usable. Justify your answer. (10 pts)

```
======= System 1  Stats =========
Task1
   Average Time: 124.683
   Average errors: 2.00625
   Average crashes: 3.5025
Task2
   Average Time: 125.029
   Average errors: 1.9975
   Average crashes: 3.48875
Task3
   Average Time: 120.866
   Average errors: 2.0
   Average crashes: 3.495
Task4
   Average Time: 124.858
   Average errors: 2.02375
   Average crashes: 3.51125
Task5
   Average Time: 122.148
   Average errors: 1.985
   Average crashes: 3.48375
Task6
   Average Time: 123.987
   Average errors: 2.03125
   Average crashes: 3.54125
Task7
   Average Time: 121.793
   Average errors: 2.00375
   Average crashes: 3.5225
========== Total ==============
Average all tasks 123.338
Average All trials: 2.007
Average All crahes: 3.506
```

```
======= System 2  Stats =========
Task1
```

```
   Average Time: 56.097
   Average errors: 4.4875
   Average crashes: 2.49
Task2
   Average Time: 58.704
   Average errors: 4.47625
   Average crashes: 2.50625
Task3
   Average Time: 57.211
   Average errors: 4.49875
   Average crashes: 2.515
Task4
   Average Time: 58.662
   Average errors: 4.51
   Average crashes: 2.50625
Task5
   Average Time: 58.555
   Average errors: 4.50375
   Average crashes: 2.48625
Task6
   Average Time: 60.896
   Average errors: 4.53
   Average crashes: 2.52
Task7
   Average Time: 54.674
   Average errors: 4.52375
   Average crashes: 2.485
========== Total ==============
Average all tasks 57.828
Average All trials: 4.504
Average All crahes: 2.501
```

A)

The System Us1 can be more usable if we see the relation between average_time and average_errors which is Proposed hierarchical usability model can be implemented using fuzzy logic controller by defining the membership function of each input (7 factors of proposed model) and output (usability). For each member function, linguistic values are defined ranging 0–9 and certain fuzzy rules are defined and on the basis of these values and rules the fuzzy logic controller generates the desired output .

$$\mu_A\left(\begin{array}{c} \\ x \\ \\ \end{array}\right) = \left(\begin{array}{ll} \frac{x-l}{m-1} & l \le x \le m; \\ \\ \frac{m-x}{u-m} & m \le x \le u; \\ \\ 0 & \text{otherwise} \end{array}\right)$$

```
============ usability using equation 1 ==========================
Usabilty of system 1 is: 0.264
Usabilty of system 2 is: 0.213
Usabilty of system 3 is: 0.247
Usabilty of system 4 is: 0.361
Usabilty of system 5 is: 0.131
Usabilty of system 6 is: 0.189
Usabilty of system 7 is: 0.269
```

```
======= System 3  Stats =========
Task1
   Average Time: 112.771
   Average errors: 2.995
   Average crashes: 2.53125
Task2
   Average Time: 113.122
   Average errors: 3.02
   Average crashes: 2.53625
Task3
   Average Time: 114.267
   Average errors: 3.01125
   Average crashes: 2.5025
Task4
   Average Time: 113.73
   Average errors: 3.02375
   Average crashes: 2.5075
Task5
   Average Time: 113.41
   Average errors: 3.00625
   Average crashes: 2.47625
Task6
   Average Time: 112.434
   Average errors: 2.99125
   Average crashes: 2.49625
Task7
   Average Time: 112.229
   Average errors: 3.0275
   Average crashes: 2.47625
=========== Total ==============
Average all tasks 113.138
Average All trials: 3.011
Average All crahes: 2.504


======= System 4  Stats =========
Task1
   Average Time: 82.106
   Average errors: 2.4825
   Average crashes: 1.50375
Task2
   Average Time: 80.568
   Average errors: 2.48
   Average crashes: 1.49625
Task3
```

```
   Average Time: 80.337
   Average errors: 2.45875
   Average crashes: 1.49
Task4
   Average Time: 83.663
   Average errors: 2.4875
   Average crashes: 1.50375
Task5
   Average Time: 81.668
   Average errors: 2.48875
   Average crashes: 1.48
Task6
   Average Time: 83.741
   Average errors: 2.49625
   Average crashes: 1.49125
Task7
   Average Time: 84.047
   Average errors: 2.49
   Average crashes: 1.50375
========== Total ==============
Average all tasks 82.304
Average All trials: 2.483
Average All crahes: 1.496


======= System 5  Stats =========
Task1
   Average Time: 96.752
   Average errors: 5.4625
   Average crashes: 5.10375
Task2
   Average Time: 95.228
   Average errors: 5.42875
   Average crashes: 5.02625
Task3
   Average Time: 101.576
   Average errors: 5.455
   Average crashes: 5.03
Task4
   Average Time: 100.049
   Average errors: 5.48
   Average crashes: 4.97875
Task5
   Average Time: 105.128
   Average errors: 5.435
   Average crashes: 5.00875
Task6
   Average Time: 99.0
   Average errors: 5.44625
   Average crashes: 4.98
Task7
   Average Time: 99.117
   Average errors: 5.485
   Average crashes: 4.9775
========== Total ==============
Average all tasks 99.55
Average All trials: 5.456
Average All crahes: 5.015


======= System 6  Stats =========
```

```
Task1
   Average Time: 70.203
   Average errors: 4.51375
   Average crashes: 2.995
Task2
   Average Time: 73.772
   Average errors: 4.505
   Average crashes: 2.99375
Task3
   Average Time: 70.121
   Average errors: 4.53
   Average crashes: 2.9975
Task4
   Average Time: 68.507
   Average errors: 4.515
   Average crashes: 3.04
Task5
   Average Time: 70.571
   Average errors: 4.4925
   Average crashes: 3.0
Task6
   Average Time: 68.007
   Average errors: 4.52625
   Average crashes: 3.01375
Task7
   Average Time: 70.677
   Average errors: 4.47625
   Average crashes: 3.02625
========== Total ==============
Average all tasks 70.265
Average All trials: 4.508
Average All crahes: 3.009


======= System 7  Stats =========
Task1
   Average Time: 48.422
   Average errors: 3.49375
   Average crashes: 1.98
Task2
   Average Time: 47.78
   Average errors: 3.47625
   Average crashes: 2.04375
Task3
   Average Time: 46.684
   Average errors: 3.48
   Average crashes: 1.98625
Task4
   Average Time: 49.744
   Average errors: 3.49375
   Average crashes: 1.98625
Task5
   Average Time: 48.8
   Average errors: 3.46
   Average crashes: 2.01375
Task6
   Average Time: 49.835
   Average errors: 3.49
   Average crashes: 2.02375
Task7
   Average Time: 48.429
```

```
   Average errors: 3.4825
   Average crashes: 2.03
========== Total ==============
Average all tasks 48.528
Average All trials: 3.482
Average All crahes: 2.009
```

B)

The System Us2 is more usable because Usability model considers all the inputs (the usability factors) together, so that generates too many rules and additionally it is difficult for the experts to consider all formulates rules with proper emphasis, since each input parameter has three linguistic values (Low, Medium and High). Hence, the proposed model with seven usability factors has a maximum number of 37 = 2187 rules.

```
============ usability using equation 2 ==========================
Usabilty of system 1 is: 0.07
Usabilty of system 2 is: 0.062
Usabilty of system 3 is: 0.057
Usabilty of system 4 is: 0.074
Usabilty of system 5 is: 0.041
Usabilty of system 6 is: 0.055
Usabilty of system 7 is: 0.077
```

$$P(\text{factor}) = \frac{\text{Number offavorable attributes in a factor whose value is 1}}{\text{Total number of attributes in a factor}}$$

$$\text{Factor}_{\text{value}} = P(\text{factor})^{\star}\,\text{Max}_{\text{value of mapping scale}}$$

2. Given the following usability equation:

$$U = \frac{1}{w_1 \times average\_time} + \frac{1}{w_2 \times average\_trials} + \frac{1}{w_3 \times average\_crashes} \quad (3)$$

- How would you set the weights w_1, w_2, w_3 to reduce the usability value of a system that has many crashes (briefly justify your answer). (15 pts)

We can set the w1 , w2 ,w3 by using the usability normalization factor where

$w_j = g_j / \sum_{j=1}^{n} g_j$ usability evaluation values can be performed by mean variance

$$v_j = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(x_{ij} - \overline{x_j}\right)^2}$$ after calculating the normalization formula we can conclude that

W2>W2>W1 If we take Perfect example to set the relations between them then following example look it out

	M_1	M_2	M_3	M_4
CSR-1	2.283	2.861	2.321	1.738
CSR-2	2.501	2.112	2.298	2.825
CSR-3	1.595	1.882	2.356	2.437
CSR-5	1.867	2.282	2.623	1.896
CSR-6	1.595	1.876	2.155	1.257

When applying Entropy method, index j has its entropy value e_j calculated:

$e_j = -k\sum_{i=1}^{n} P(x_{ij}) \ln P(x_{ij})$, $e_j > 0$, $k > 0$. Wherein x_{ij} is evaluation score of CSR i 's index j ; $P(x_{ij})$ is eigenratio of CSR i under index j : $P(x_{ij}) = x_{ij} / \sum_{i=1}^{n} x_{ij}$ $(n!/r!(n-r)!)$.

With regard to the present evaluation, $n = 6$, while $k = 1/\ln 6 = 0.5581$. Calculation is:

$$E = \{e_1 \quad e_2 \quad e_3 \quad e_4\} = \{0.9917 \quad 0.9937 \quad 0.9965 \quad 0.9791\} \tag{3}$$

Index j has coefficient variation g_j calculated: $g_j = 1 - e_j$, then the calculation is:

$$G = \{g_1 \quad g_2 \quad g_3 \quad g_4\} = \{0.0083 \quad 0.0063 \quad 0.0035 \quad 0.0209\} \tag{4}$$

Provided the normalization formulation $w_j = g_j / \sum_{j=1}^{n} g_j$, weight vector values shall be decided by Entropy calculation:

$$W_{01} = \{w_1 \quad w_2 \quad w_3 \quad w_4\} = \{0.2131 \quad 0.1609 \quad 0.0892 \quad 0.5367\} \tag{5}$$

4.3. Mean-Variance-based Weighting.

Refer to the usability evaluation values given in Figure 4. Calculation of mathematically expected value is performed as $\overline{x_j} = \frac{1}{n}\sum_{i=1}^{n} x_{ij}$, and get values

$\overline{X} = \{\overline{x_1} \quad \overline{x_2} \quad \overline{x_3} \quad \overline{x_4}\} = \{2.0065 \quad 2.1790 \quad 2.4588 \quad 2.1630\}$; Calculation of mean variance is performed as: $v_j = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(x_{ij} - \overline{x_j}\right)^2}$, and get values:

$$V = \{v_1 \quad v_2 \quad v_3 \quad v_4\} = \{0.3784 \quad 0.3672 \quad 0.3058 \quad 0.6363\}$$

Provided the normalization formulation $w_j = v_j / \sum_{j=1}^{n} v_j$, weight vector values shall be decided by Mean Variance calculation:

$$W_{O2} = \{w_1 \quad w_2 \quad w_3 \quad w_4\} = \{0.2242 \quad 0.2176 \quad 0.1812 \quad 0.3770\} \tag{6}$$

4.4. Synthesis of Weight Values and Result Analysis.

Entropy-based W_{O1} and Mean-Variance-based W_{O2} are synthesized to get individual weight vector $W_O^{'}$:

$$W_O^{'} = \{w_{O1}^{'} \quad w_{O2}^{'} \quad w_{O2}^{'} \quad w_{O2}^{'}\} = \{0.2186 \quad 0.1871 \quad 0.1271 \quad 0.4498\} \tag{7}$$


```
============ Most usable sytem ==========================
The most usable System using equation 1 is system:  1


The most usable System using equation 2 is system:  4
```

1. Write the python code to read the dataset, and perform the following tasks

   (a) Display the systems statistics. Your output should look like the output provided in the text file output.txt. (30 pts)

   (b) Plot and save the average time histogram. Each bin in the histogram represents the average time for each system. Your plot should look like the figure 1 below. (15 pts)

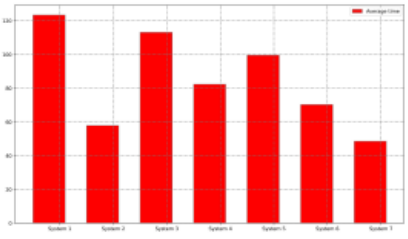

Figure 1: Average time histogram

```
(a)
for folder in folders:
    print(f'{folder}:')
    print (f" - Average Time:
{np.mean(data[folder]['task1'].values):.2f} seconds")
    print(f"  - Average Errors:
{np.mean(data[folder]['errors'].values):.2f}")
    print(f"  - Average Crashes:
{np.mean(data[folder]['crashes'].values):.2f}")
    print()

(b)
import pandas as pd
# Generate data on commute times.

avg_times = [np.mean(data[folder]['time'].values) for folder in
folders]
plt.hist(avg_times, bins=7, color='blue', alpha=0.5)
plt.xlabel('Average Time (s)')
plt.ylabel('Frequency')
plt.title('Average Time Histogram')
plt.savefig('avg_time_histogram.png')
plt.show()
```

(c) Plot and save the average errors and crash histograms. The bins in the histogram represent the average error and crash for each system. Your plot should look like the figure 2 below. (15 pts)

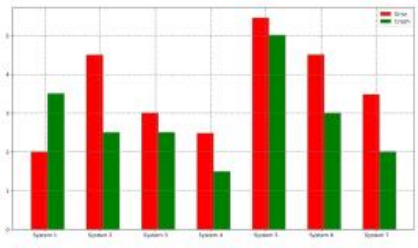

Figure 2: Errors-crashes histogram

(c)

```
import pandas

avg_errors = [np.mean(data[folder]['errors'].values) for folder in folders]
avg_crashes = [np.mean(data[folder]['crashes'].values) for folder in folders]

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10,5))
ax1.hist(avg_errors, bins=7, color='red', alpha=0.5)
ax1.set_xlabel('Average Errors')
ax1.set_ylabel('Frequency')
ax1.set_title('Average Errors Histogram')

ax2.hist(avg_crashes, bins=7, color='green', alpha=0.5)
ax2.set_xlabel('Average Crashes')
ax2.set_ylabel('Frequency')
ax2.set_title('Average Crashes Histogram')

plt.tight_layout()
plt.savefig('errors_crashes_histogram.png')
plt.show()
```

(d) Plot and save the systems usability histograms using equation (1) and equation (2). The bins in the histogram represent the usability computed using equation (1) and equation (2) respectively for each system. Your plot should look like the figure 3 below. (15 pts)

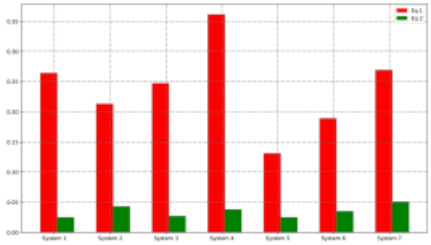

Figure 3: Usability histogram

(d)

```
us1 = [usability1(data[folder]['time'].values, data[folder]['errors'].values,
data[folder]['crashes'].values) for folder in folders]
us2 = [usability2(data[folder]['time'].values, data[folder]['time'].shape[1],
data[folder]['crashes'].values) for folder in folders]

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10,5))
ax1.hist(us1, bins=7, color='purple', alpha=0.5)
ax1.set_xlabel('Usability Equation 1')
ax1.set_ylabel('Frequency')
ax1.set_title('Usability Equation')
```