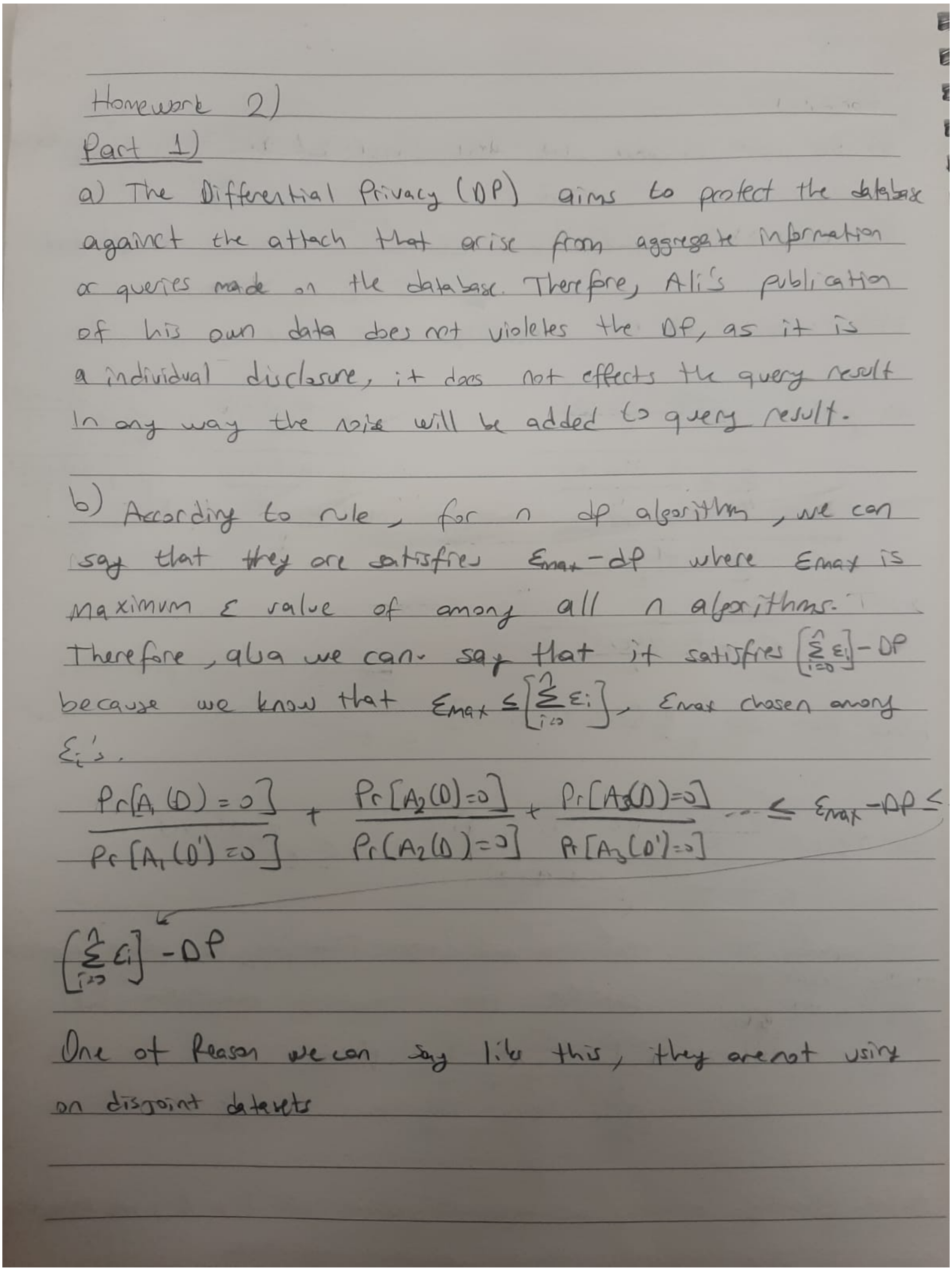
Homework 2 Report



Homework 2)

Part 1)

a) The Differential Privacy (DP) aims to protect the database against the attach that arise from aggregate information or queries made on the database. Therefore, Ali's publication of his own data does not violetes the DP, as it is a individual disclosure, it does not effects the query result In any way the noise will be added to query result.

b) According to rule, for n dp algorithm, we can say that they are satisfies ε_{max}-dp where ε_{max} is maximum ε value of among all n algorithms. Therefore, also we can say that it satisfies $\left[\sum_{i=0}^{n} \varepsilon_i\right]$-DP because we know that $\varepsilon_{max} \leq \left[\sum_{i=0}^{n} \varepsilon_i\right]$, ε_{max} chosen among ε_i's.

$$\frac{Pr[A_1(D)=0]}{Pr[A_1(D')=0]} + \frac{Pr[A_2(D)=0]}{Pr(A_2(D)=0]} + \frac{Pr[A_3(D)=0]}{Pr[A_3(D')=0]} \dots \leq \varepsilon_{max}-DP \leq$$

$$\left[\sum_{i=0}^{n} \varepsilon_i\right]-DP$$

One of Reason we can say like this, they are not using on disjoint datasets

Part 2)

a) We want to show that Φ satisfies $\frac{Pr[\Phi(v_1)=y]}{A[\Phi(v_2)=y]} \leq e^{a\, d(v_1,v_2)}$

We know that $Pr[\Phi(v)=y] = \frac{e^{\frac{-a \cdot d(v,y)}{2}}}{\sum_{z\in V} e^{\frac{-a \cdot d(v,z)}{2}}}$ and we plugged this output to the unequality that above.

We get

$$\frac{\dfrac{e^{\frac{-a \cdot d(v_1,y)}{2}}}{\sum_{z\in V} e^{\frac{-a \cdot d(v_1,z)}{2}}}}{\dfrac{e^{\frac{-a \cdot d(v_2,y)}{2}}}{\sum_{z\in V} e^{\frac{-a \cdot d(v_2,z)}{2}}}} = \underbrace{\frac{e^{\frac{-a \cdot d(v_1,y)}{2}}}{e^{\frac{-a \cdot d(v_2,y)}{2}}}}_{1} \cdot \underbrace{\frac{\sum_{z\in V} e^{\frac{-a\, d(v_2,z)}{2}}}{\sum_{z\in V} e^{\frac{-a\, d(v_1,z)}{2}}}}_{2}$$

By (3) and (4) ~~properties~~;

Now for 1: $e^{\frac{a}{2}[d(v_2,y) - d(v_1,y)]} \leq e^{a/2\, d(v_1,v_2)}$

for 2: $\frac{e^{\frac{-a\, d(v_2,z_1)}{2}} + e^{\frac{-a\, d(v_2,z_2)}{2}} + \cdots}{e^{\frac{-a\, d(v_1,z_1)}{2}} + e^{\frac{-a\, d(v_2,z_2)}{2}} + \cdots} \leq e^{a/2\, d(v_1,v_2)}$

This package is smaller or equal to $e^{a/2\, d(v_1,v_2)}$ like in 1

This also and so on. Therefore, the whole expression satisfies the inequality condition. i.e.

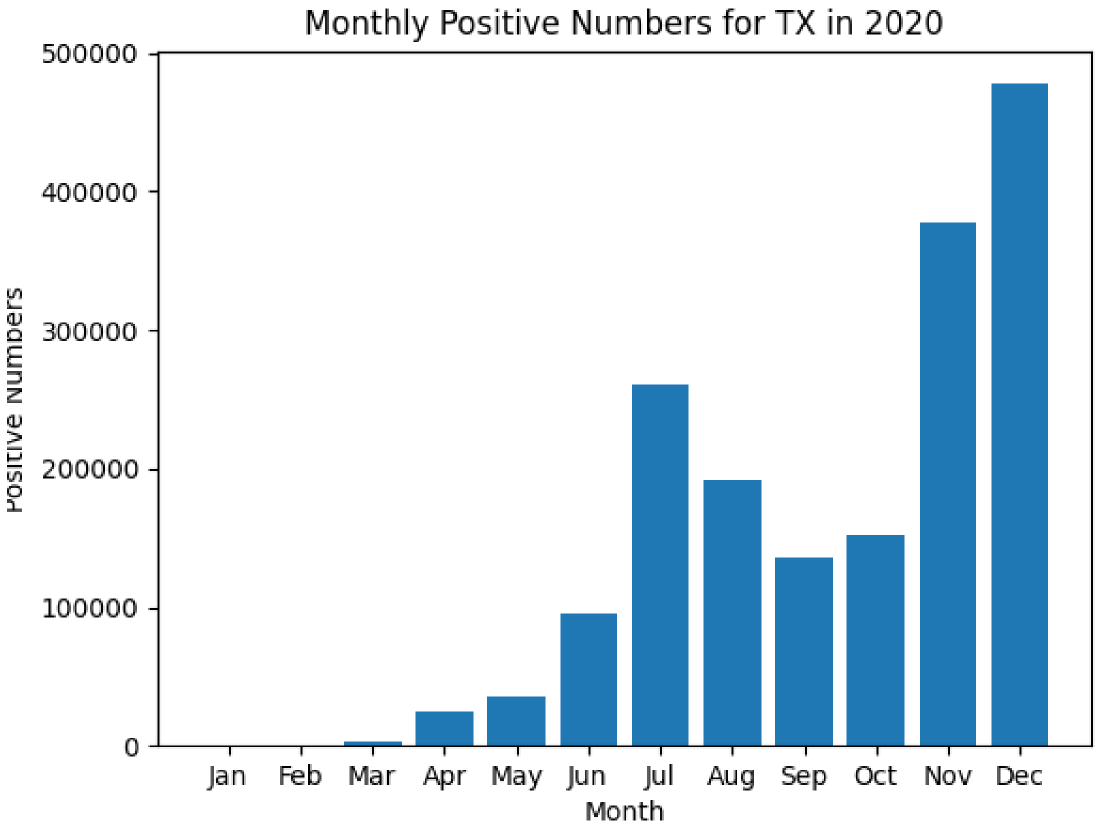
$$\frac{\boxed{1} + \boxed{2} + \boxed{3}}{\boxed{6} + \boxed{7} + \boxed{8}} = \boxed{\frac{6}{21}}$$

$\leq 1/2$ $\quad \leq 1/2$ $\quad \leq 1/2$ $\qquad \leq 1/2$

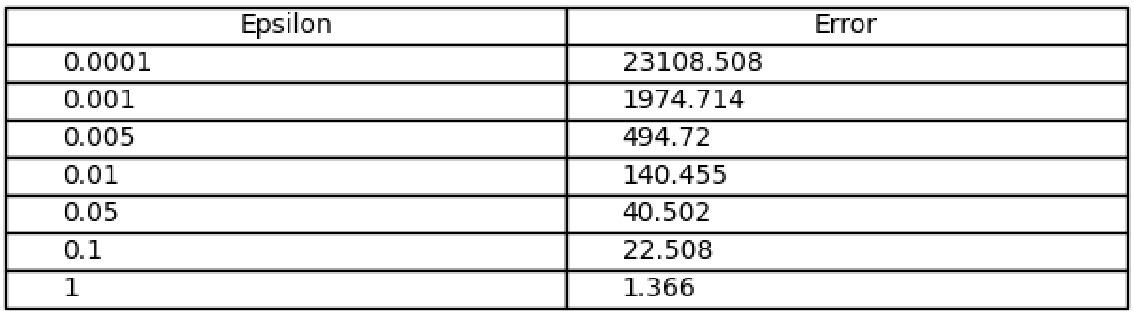
And finally because we take the product of 1 and 2, we take the sum of the powers. Hence we can say that

$$\frac{Pr[\Phi(v_1)=y]}{Pr[\Phi(v_2)=y]} \leq e^{a \cdot d(v_1,v_2)}$$

Part 3)



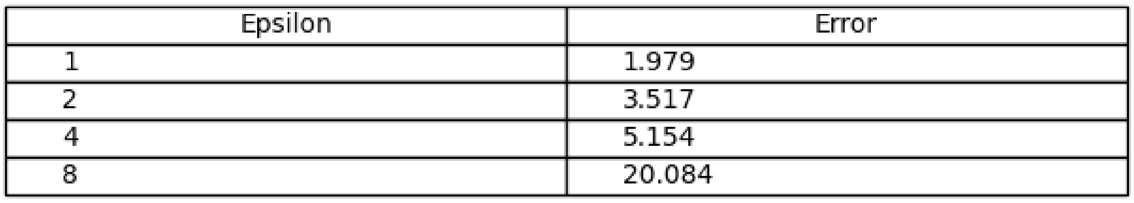
Laplace Experiment Result:



Epsilon	Error
0.0001	23108.508
0.001	1974.714
0.005	494.72
0.01	140.455
0.05	40.502
0.1	22.508
1	1.366

We use Laplace distribution to add noise to data and we decide the value that we choose from Laplace distribution with b value which is equal to Sensitivity(N) / Epsilon. As Epsilon gets bigger the b value will be smaller. Therefore, the added noise to data will be smaller, and so is the Error.

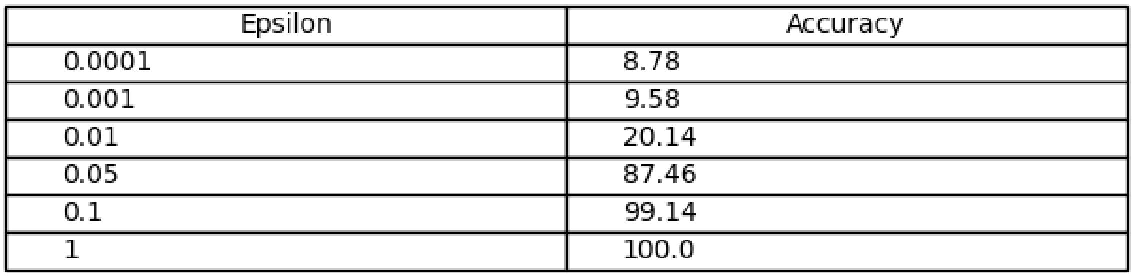
N Value Experiment:



Epsilon	Error
1	1.979
2	3.517
4	5.154
8	20.084

Again, we use Laplace distribution to add noise to data and we decide the value that we choose from Laplace distribution with b value which is equal to Sensitivity(N) / Epsilon. As N gets bigger the b value will be bigger. Therefore, the added noise to data will be bigger, and so is the Error.

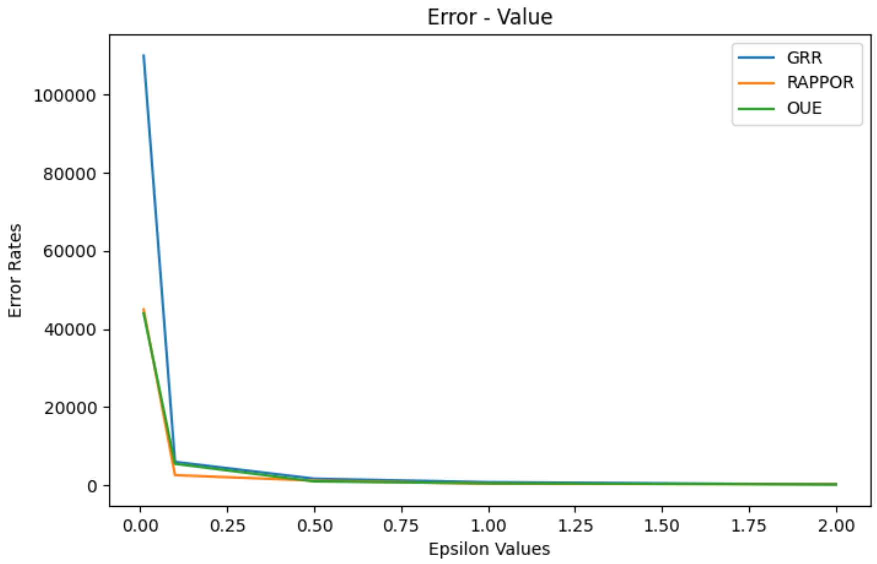
Exponential Experiment Result:



Epsilon	Accuracy
0.0001	8.78
0.001	9.58
0.01	20.14
0.05	87.46
0.1	99.14
1	100.0

While Epsilon gets bigger, the noise that we add will be smaller again. Therefore, we can see like a hundred percent accuracy in the result.

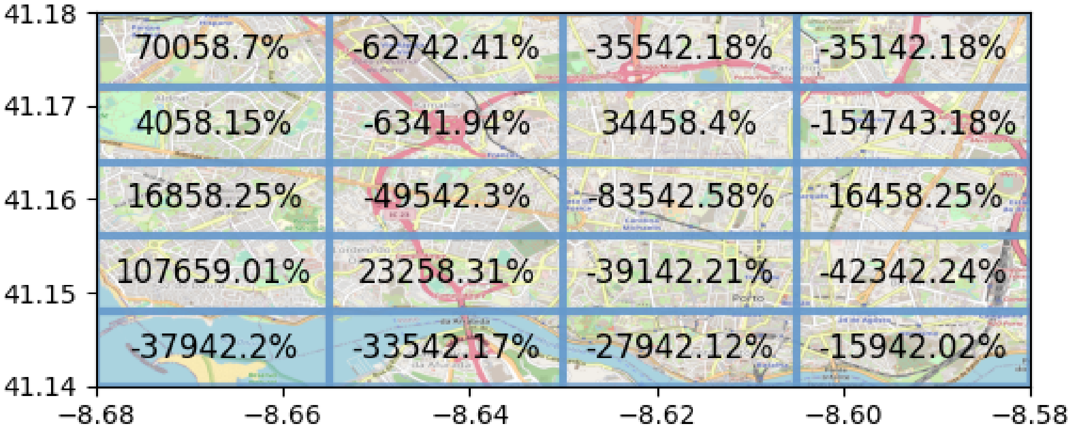
Part 4)



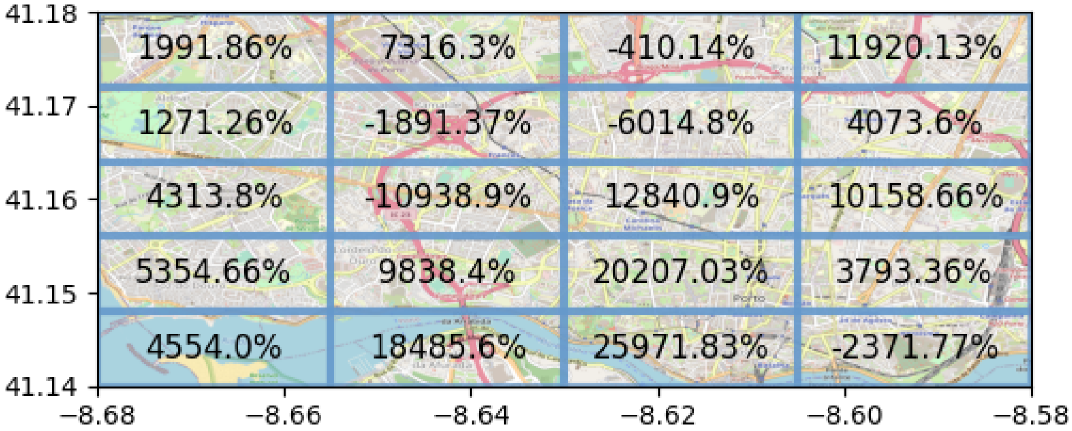
According to this result, we cannot say there is a better protocol in all values of epsilon. Also, errors of all of them get smaller, while epsilon value is getting bigger because the added noise is getting smaller, while epsilon value is getting bigger.

Visual Analysis:

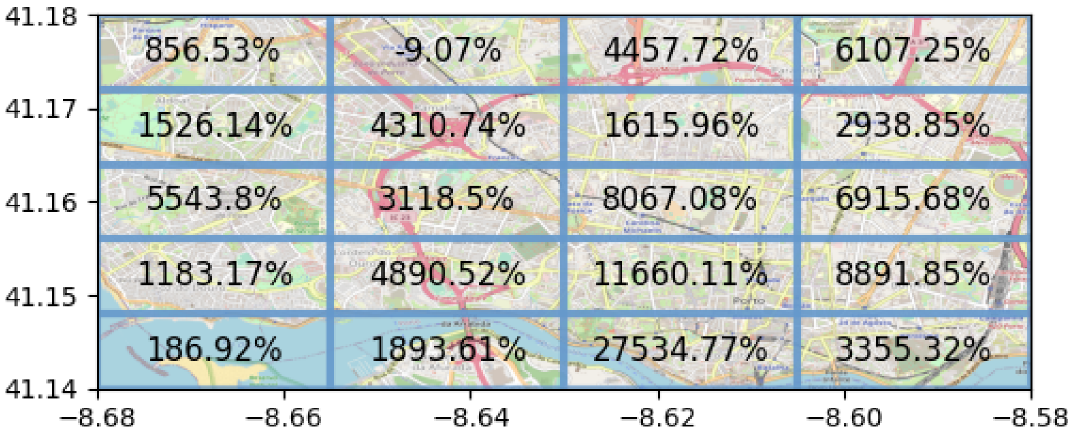
0.01:



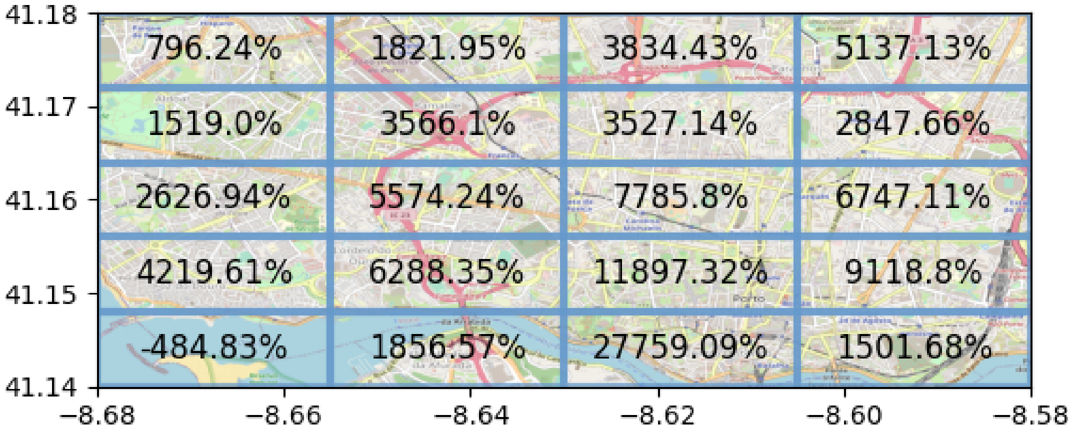
0.1:



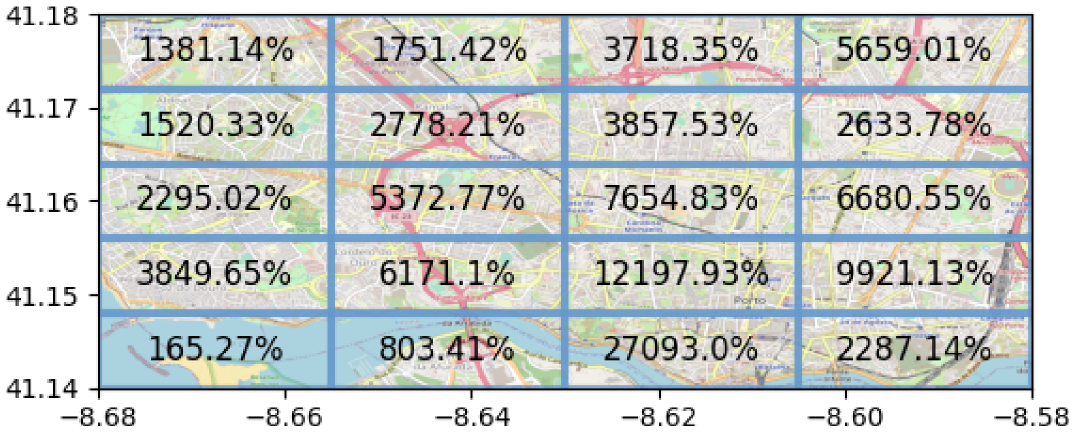
0.5:


41.18
41.17
41.16
41.15
41.14
−8.68
−8.66
−8.64
−8.62
−8.60
−8.58
856.53%
-9.07%
4457.72%
6107.25%
1526.14%
4310.74%
1615.96%
2938.85%
5543.8%
3118.5%
8067.08%
6915.68%
1183.17%
4890.52%
11660.11%
8891.85%
186.92%
1893.61%
27534.77%
3355.32%

1:


41.18
41.17
41.16
41.15
41.14
−8.68
−8.66
−8.64
−8.62
−8.60
−8.58
796.24%
1821.95%
3834.43%
5137.13%
1519.0%
3566.1%
3527.14%
2847.66%
2626.94%
5574.24%
7785.8%
6747.11%
4219.61%
6288.35%
11897.32%
9118.8%
-484.83%
1856.57%
27759.09%
1501.68%

2:


41.18
41.17
41.16
41.15
41.14
−8.68
−8.66
−8.64
−8.62
−8.60
−8.58
1381.14%
1751.42%
3718.35%
5659.01%
1520.33%
2778.21%
3857.53%
2633.78%
2295.02%
5372.77%
7654.83%
6680.55%
3849.65%
6171.1%
12197.93%
9921.13%
165.27%
803.41%
27093.0%
2287.14%

I chose the OUE protocol to plot the grid. I think because of the highness of the error we get these awkward results. It seems very unrealistic. Even if I put the plot_grid function in the our_experiment function and use the estimated frequency. On the other hand, while we make the epsilon values bigger, we get less errors in our result as we can see from the plots.