Part I
Analyzing a Real-World Graph

Coding results

Task 1 : Basic graph statistics extraction

```
{'total_edges': 25998,
 'total_nodes': 9877}
max number of edges 48772626
```

Task 2 : Extract the connected components of the graph

```
Graph has 429 connected components
Largest connected component:
{'total_edges': 24827,
 'total_nodes': 8638}
8638 total_nodes represent 87.46% of the graph
24827 total_edges represent 95.50% of the graph
```

Task 3 : Statistics on the degrees of the nodes of the graph

```
{'degree_of_nodes': {'max': 65,
                     'mean': 5.264351523742027,
                     'median': 3.0,
                     'min': 1}}
```

Task 4 : Degree histogram plot

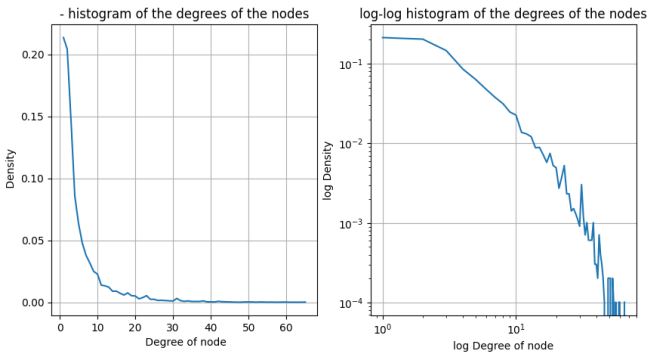

Figure 1: Histogram of degrees of the nodes

Task 5 : Global clustering coefficient

```
Graph clustering coefficient: 0.284
```

1 Question 1:

Assume $G = (V, E)$ is an undirected graph of n nodes without self-loops. $|V| = n$.

Number of edges

The maximum number of edges is the cardinal of the set of possible combinations of 2 nodes chosen from n nodes. which is equal to the binomial coefficient $\binom{n}{2} = \frac{n!}{2!(n-2)!}$

$$|E| \leq \frac{n * (n-1)}{2} \tag{1}$$

This can also be viewed when writing the adjacency matrix of a complete graph.

- A matrix full of 1 has n^2 elements
- set the diagonal to 0 to remove the self loops. $n * (n-1)$ elements
- Divide by two since we consider an undirected graph.

In the code the property is verified through an assert.

Number of triangles

The maximum number of triangles is $\binom{n}{3} = \frac{n!}{3!*(n-3)!} = \frac{n*(n-1)(n-2)}{6}$ when choosing the combination of 3 nodes from n nodes.

2 Question 2 : 2 graphs having the same degree distribution $\not\Longleftrightarrow$ isomorphic

If two graphs have the same degree distribution, it does not imply that they are isomorphic to each other. The simplest counter-example with degrees of nodes of 2 is presented in 2, it's a bit of a degenerate case as we're using the fact that a loop adds two to the degree in case of undirected graphs. Below are four cases illustrating the counter example. 4 shows an hexagon versus 2 triangles which is the most pleasant example to visualize.

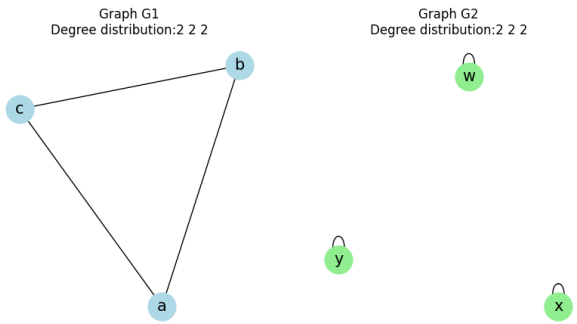

Figure 2: Graphs G1 is a triangle and G2 is made of 3 isolated nodes with a self loop.They have the same degree distribution (every node has a degree of 2)but are not isomorphic to each other.

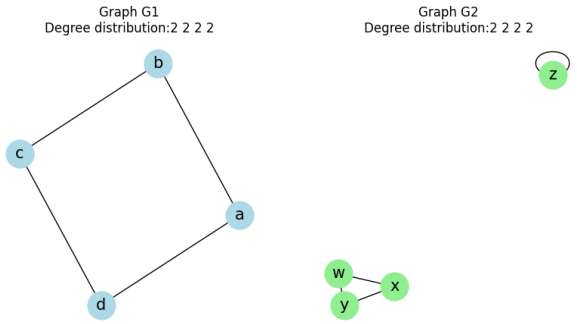

Figure 3: G1 is a rectangle, G2 is made of a triangle and a single node with a self loop.

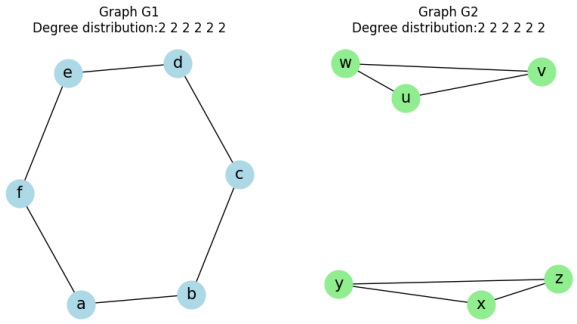

Figure 4: G1 is an hexagon, it has 6 edges. G2 has 2 separate triangles.All nodes have a degree of 2, G1 and G2 have the same degree histograms.But they are not isomorphic to each other

3 Question 3 : n-cycle graphs

- Clustering coefficient of the triangle C_3 is $1 = \frac{1}{1+0}$
- Clustering coefficient of cycle graphs C_n where $n \geq 4$ is $0 = \frac{0}{0+n}$ as there are no closed triplets but only (n) open triplets.

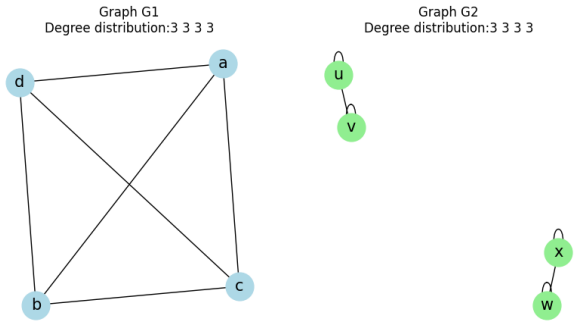

Figure 5: Counter example where all nodes have a degree of 3.G1=(a-b, b-c, c-a, b-d, d-c, d-a) is a rectangle with its diagonalsG2=(u-v, u-u, v-v, w-x, w-w, x-x) are 2 segment where the end nodes have self loops

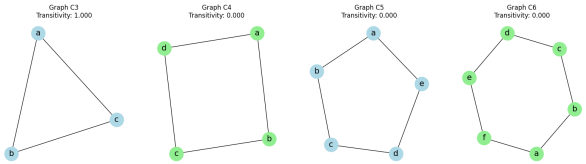

Figure 6: Transitivity of C_n cycle graphs becomes 0 if $n \geq 4$

Part II
Community detection

Task 6 : Toy example for spectral graph clustering

```
ADJACENCY MATRIX:
[[0 1 1 0 0 0 0 0 0 0 0 0 0]
 [1 0 1 0 0 0 0 0 0 0 0 0 0]
 [1 1 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 1 0 1 0 0 0 0 0 0]
 [0 0 0 1 0 1 0 0 0 0 0 0 0]
 [0 0 0 0 1 0 1 0 0 0 0 0 0]
 [0 0 0 1 0 1 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 1 1 1 1 1 1]
 [0 0 0 0 0 0 0 1 0 0 0 0 0]
 [0 0 0 0 0 0 0 1 0 0 0 0 0]
 [0 0 0 0 0 0 0 1 0 0 0 0 0]
 [0 0 0 0 0 0 0 1 0 0 0 0 0]
 [0 0 0 0 0 0 0 1 0 0 0 0 0]]
```

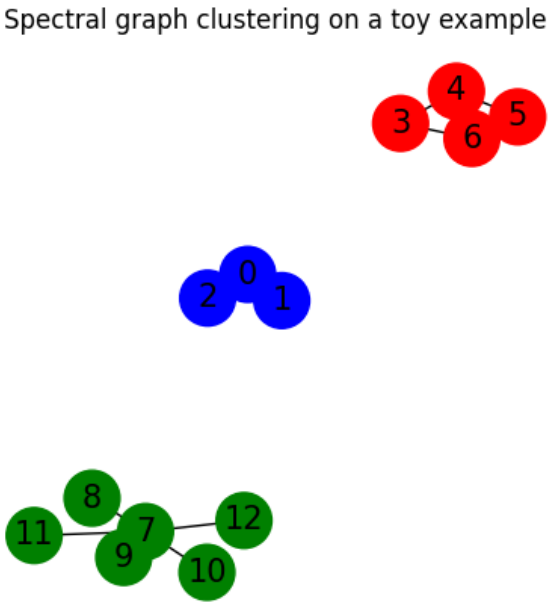

Figure 7: G =union of 3 disjoint subgraphs (complete, cycle, star). Spectral graph clustering, k=3 clusters found correctly

```
LAPLACIAN:
[[ 1.  -0.5 -0.5  0.   0.   0.   0.   0.   0.   0.   0.   0.   0. ]
 [-0.5  1.  -0.5  0.   0.   0.   0.   0.   0.   0.   0.   0.   0. ]
 [-0.5 -0.5  1.   0.   0.   0.   0.   0.   0.   0.   0.   0.   0. ]
 [ 0.   0.   0.   1.  -0.5  0.  -0.5  0.   0.   0.   0.   0.   0. ]
 [ 0.   0.   0.  -0.5  1.  -0.5  0.   0.   0.   0.   0.   0.   0. ]
 [ 0.   0.   0.   0.  -0.5  1.  -0.5  0.   0.   0.   0.   0.   0. ]
 [ 0.   0.   0.  -0.5  0.  -0.5  1.   0.   0.   0.   0.   0.   0. ]
 [ 0.   0.   0.   0.   0.   0.   0.   1.  -0.2 -0.2 -0.2 -0.2 -0.2]
 [ 0.   0.   0.   0.   0.   0.   0.  -1.   1.   0.   0.   0.   0. ]
 [ 0.   0.   0.   0.   0.   0.   0.  -1.   0.   1.   0.   0.   0. ]
 [ 0.   0.   0.   0.   0.   0.   0.  -1.   0.   0.   1.   0.   0. ]
 [ 0.   0.   0.   0.   0.   0.   0.  -1.   0.   0.   0.   1.   0. ]
 [ 0.   0.   0.   0.   0.   0.   0.  -1.   0.   0.   0.   0.   1. ]]
```

We clearly observe the 3 blocks on the diagonal. When we perform the eigen decomposition and keep the $d = 3$ eigen vectors associated with the 3 lowest eigen values (0, 0, 0).
$eigenvalues = [0, 0, 0, 1, 1, 1, 1, 1, 1.5, 1.5, 2, 2]$. The first 3 zeros correspond to the 3 disjoint subgraphs.

```
MATRIX OF EIGEN VECTORS ASSOCIATED WITH THE THREE LOWEST EIGEN VALUES
[[ 0.          0.         -0.57735027]
 [ 0.          0.         -0.57735027]
 [ 0.          0.         -0.57735027]
 [ 0.5         0.          0.        ]
 [ 0.5         0.          0.        ]
 [ 0.5         0.          0.        ]
 [ 0.5         0.          0.        ]
 [ 0.         -0.40824829  0.        ]
 [ 0.         -0.40824829  0.        ]
 [ 0.         -0.40824829  0.        ]
 [ 0.         -0.40824829  0.        ]
 [ 0.         -0.40824829  0.        ]
 [ 0.         -0.40824829  0.        ]]
```

4 Question 4 : Spectral clustering

From the trials on the code, I see that the lowest eigen value is equal to zero (see the numerical example from previous section). I don't know how to prove the property shown for the symmetric L Laplacian matrix from the class (slide 69 on community detection class) on the L^{RW} matrix.
I will assume this result. $L^{RW} * u_1 = 0 \implies u_1 = D^{-1}A * u_1 \implies D * u_1 = A * u_1$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}([u_1]_i - [u_1]_j)^2 = \sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij} * [u_1]_i^2 + \sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij} * [u_1]_j^2 - 2 * \sum_{i=1}^{n}\sum_{j=1}^{n} [u_1]_i[u_1]_j \tag{2}$$

By identifying a quadratic form.

$$\sum_{i=1}^{n}\sum_{j=1}^{n} [u_1]_i[u_1]_j = u_1^T.A.u_1 \tag{3}$$

And using the property that the row and columns of the adjacency matrix A sum to the diagonal elements of D.

- $\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij} * [u_1]_i^2 = \sum_{i=1}^{n}([u_1]_i^2 \sum_{j=1}^{n} A_{ij}) = \sum_{i=1}^{n}[u_1]_i^2 * D_{i,i} = u_1^T D u_1$
- $\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij} * [u_1]_j^2 = \sum_{j=1}^{n}([u_1]_j^2 \sum_{i=1}^{n} A_{ij}) = u_1^T D u_1$

We get

$$\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}([u_1]_i - [u_1]_j)^2 = 2. * (u_1^T.(D - A).u_1) = 0 \tag{4}$$

According to the "assumption" on the null eigen value associated with u_1, we have $(D - A).u_1 = 0$, therfore the expression is equal to zero.

Task 9

```
---------------------------------------- task_9 ----------------------------------------
Modularity computation of the giant connected component of the CA-HepTh dataset.
----------------------------------------------------------------------------------------
Modularity of the spectral k=50 clustering 0.037
Modularity of the random graph partition -0.000
```

5 Question 5 : Modularity

Left graph

Let's have 2 classes denoted by B for blue and Y for yellow. $c \in [B, Y]$

- $n_c = 2$ (number of communities)
- $m = 14$ (total number of edges)
- $l_B = l_Y = 6$ (6 edges inside the blue or yellow community)
- $d_B = d_Y = 2 * 3 + 2 * 4 = 14$ (degree of v1 and v2 =3 , degree of v3 and v4 = 4)

Finally we have $Q_{left} = 2 * \left[\frac{6}{14} - \frac{14}{2*14}^2\right] = 2 * \frac{6*4-14}{4*14} = \frac{5}{14} \approx 0.357$

Right graph

- $n_c = 2$ (number of communities)
- $m = 14$ (total number of edges)
- $l_B = 5$ (3 edges inside the blue triangle v1, v2, v4 + 2 blue segments (v4-v6 and v6-v8))
- $l_Y = 2$ (2 edges inside the yellow community v3-v5 and v5-v7)
- $d_B = 3 * 3 + 2 * 4 = 17$ (degree of v1, v2, v8 = 3, degree of v4 and v6 =4)
- $d_Y = 1 * 3 + 2 * 4 = 11$ (degree of v7=3 , degree of v3 and v5 = 4)

Finally we have $Q_{right} = \left[\frac{5}{14} - (\frac{17}{2*14})^2\right] + \left[\frac{2}{14} - (\frac{11}{2*14})^2\right] = \frac{5*4*14-17^2+2*4*14-11^2}{4*14^2} = -\frac{9}{2*14^2} \approx -0.023$
In 8, we test the python modularity function. We get the same numerical results indeed.

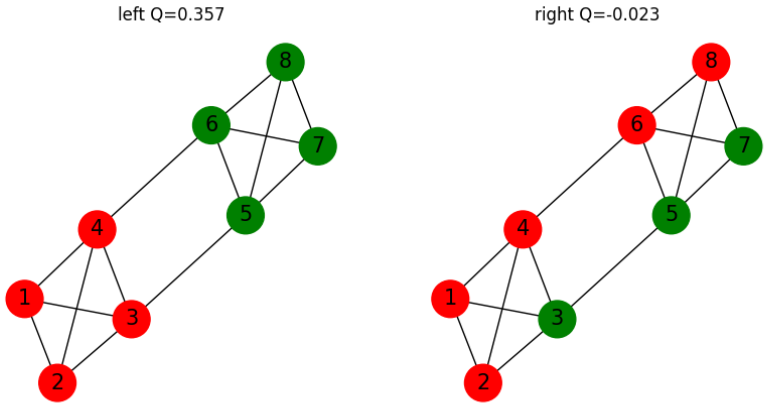

Figure 8: modularity comparison (computation using the python implementation)

Part III
Graph classification

Task 10 : Cycle and graph dataset creation

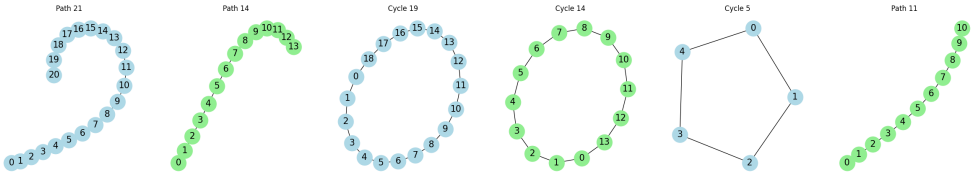

Figure 9: Dataset is made of cycles C_n and paths P_n

6 Question 6 : Shortest path kernel

In 10 and 11, we show how we progressively build the shortest paths graph for the path P_4 and the start S_4. Then we simply compute the distribution of the edges weights (=length) to get the so called "feature map" ϕ. We get:

- $\phi(P_4) = [3, 2, 1, 0...]^T$
- $\phi(S_4) = [3, 3, 0, 0...]^T$
- $k(P_4, P_4) = 3^2 + 2^2 + 1^2 = 14$
- $k(P_4, S_4) = K(S_4, P_4) = 3 * 3 + 2 * 3 + 1 * 0 = 15$
- $k(S_4, S_4) = 3^2 + 3^2 = 18$

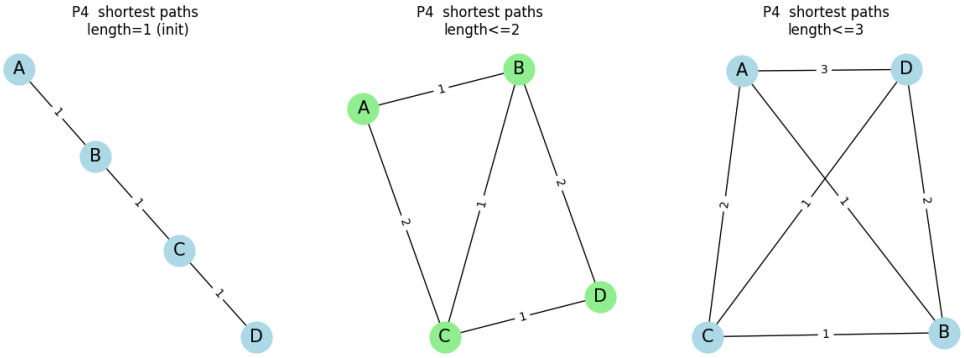

Figure 10: Path graph P_4, $\phi(P_4) = [3, 2, 1, 0...]^T$

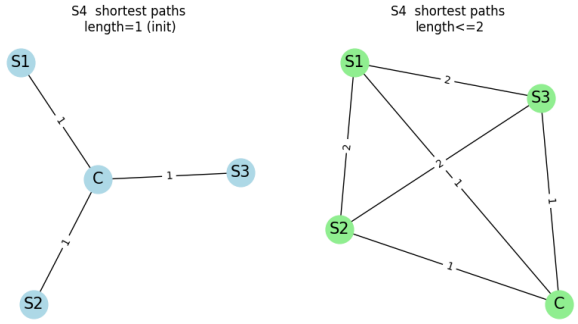

Figure 11: Start graph S_4 , $\phi(S_4) = [3, 3, 0, 0...]^T$

Task 11: Finding isomorphic graphlets in subgraphs

We show in 12 and 13 how to compute the feature maps for each sampled triplet subgraph. We repeat this sampling process $N = 200$ times for each graph.

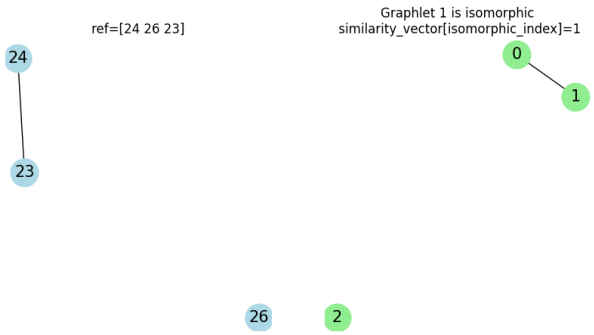

Figure 12: Finding graphlet "G_1" (notation of the code) which is isomorphic with a randomly sampled subgraph made of 3 nodes. Feature vector is $[0, 1, 0, 0]^T$ for this subgraph

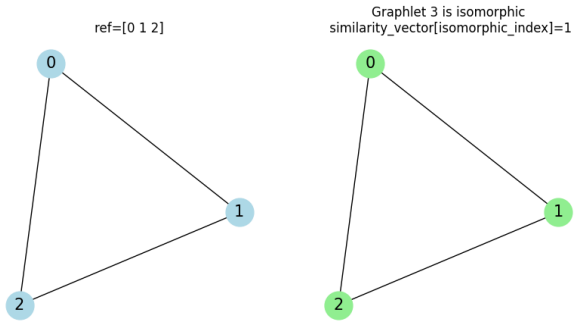

Figure 13: Finding triangle graphlet "G_3" (notation of the code) which is isomorphic with a randomly sampled subgraph made of 3 nodes? Feature vector is $[0,0,0,1]^T$ for this subgraph

Task 13 : SVM classifier comparison of kernels (graphlet vs shortest path)

```
Graphlet based kernel, SVM classifier (trained with accuracy=50.56%) Test accuracy: 45.00%
Shortest path based kernel, SVM classifier (trained with accuracy=100.00%) Test accuracy: 85.00%
```

Shortest path kernel based classifier is perfectly accurate on this dataset. Graphlet accuracy is much lower. This is explained by the fact that when we use Graphlet feature, the discriminative triplets choice involve choosing the endpoints of the Path P_n which makes a difference with cycle graphs.

7 Question 7 : "Orthogonal graphs" with the graphlet kernel

If G and G' verify $k(G,G') = 0$, it means that they do not share the same graphlet types. We can have $f_G = [m,n,k,0]^T$ and $f_{G'} = [0,0,0,p]$ for instance, G contains m * G_1, n*G_2, k*G_3. G' only contains p*G_4 graphlets for instance.
A trivial example is for instance constructed with

- $G = G_1$ and $G' = G_4$.
- $k(G,G') = [1,0,0,0] * [0,0,0,1]^T = 0$

A more elegant example involing 4 nodes.

- Path $G = P_4$ has a graphlet feature vector $f_G = [0,2,2,0]^T$ (graphlets 2*G_2 and 2*G_3)
- Dense graph G' with 4 nodes is only made of G_1 graphlets, it has a $f_{G'} = [4,0,0,0]^T$.
- Inner product $k(G,G') = f_{g'}^T . f_G = 0$