Task1

Scelte delle misure

Le misure considerate per confrontare la rete net4 con il grafo generato dai differenti modelli sono le seguenti: numero di nodi, numero di edge, numero di triangoli, coefficiente di clustering, diametro, raggio, numero di comunità, distribuzione dei gradi.

Di seguito sono riportate le misurazioni ottenute dal modello assegnato.

- Numero di nodi: 20000
- Numero di archi: 227183
- Numero di triangoli: 1017162
- Coefficiente di clustering: 0.59
- Comunità: 40
- Raggio: 35
- Diametro: 68
- Numero di componenti connesse: 1

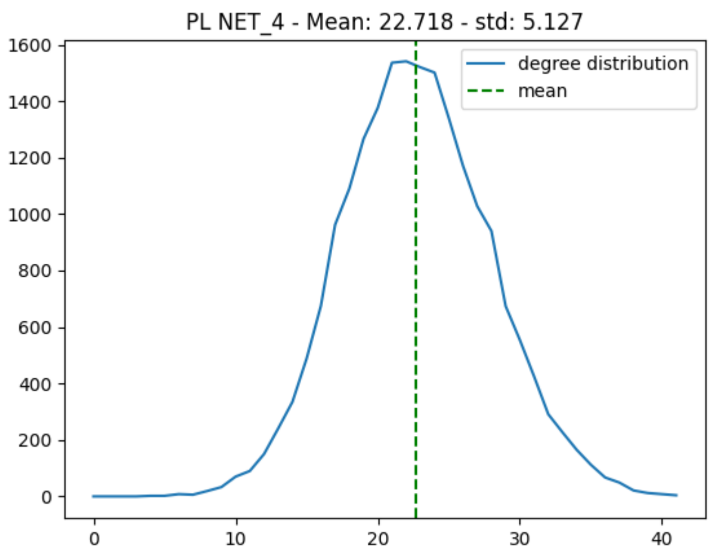

Figura 1. Distribuzione dei gradi di net4

La distribuzione dei gradi segue un andamento a campana, caratterizzata da una media di 22.718 e una deviazione standard di 5.127.

L'obiettivo proposto è quello di individuare l'algoritmo che ha generato il grafo net4 e l'individuazione dei suoi parametri, che ci avvicinano alle misurazioni elencate sopra.

Scelta del modello

Dalla distribuzione dei gradi della rete net4 si possono scartare gli algoritmi preferentialG e AffiliationG, in quanto entrambi seguono una legge di potenza. Inoltre, nei paragrafi successivi, saranno analizzati gli algoritmi restanti: ***randomG***, ***configuarationG***, e ***GenWS2DG***.

Random graph

L'algoritmo *random graph* genera un arco tra un nodo i e un nodo j con una certa probabilità p. Dalla distribuzione dei gradi di net4 si può evincere che ogni nodo in media ha 22.71 figli. Questa informazione può essere utilizzata per guidare l'algoritmo *random graph* per ottenere una distribuzione simile a quella di net4. Al fine di tale raggiungimento, viene settato p con il rapporto tra il numero in media di figli di net4 e il numero totale di nodi ($p = 0.0011$).

Di seguito vengono riportate le misurazioni ottenute con questo setting:

- Numero di nodi: 20000
- Numero di archi: 227240
- Numero di triangoli: 1919
- Coefficiente di clustering: 0.0011
- Comunità: 10
- Diametro: 5
- Numero di componenti connesse: 1

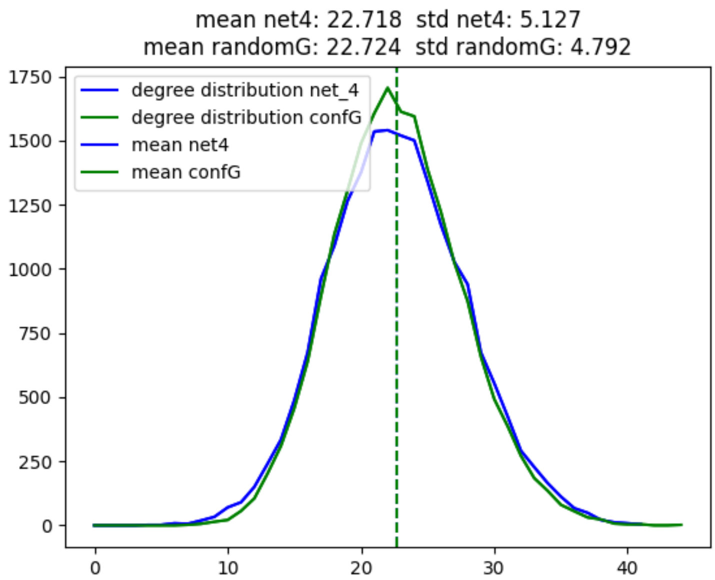

Figura 2. Confronto tra la distribuzione dei gradi di net4 del grafo generato dall'algoritmo randomG con p = 0.0011

Dalla figura mostrata sopra, si può notare che anche se si raggiunge una distribuzione "coerente" e un numero di archi dello stesso ordine con quello del modello in esame le altre misure sono ben lontane da quelle di net4; ad esempio, il numero di triangoli di net4 sono all'incirca un milione, mentre il numero di triangoli ottenuti con randomG sono nell'ordine delle migliaia. Per far sì che il numero di triangoli aumenti bisognerebbe aumentare la probabilità p in modo da creare più archi. Di seguito sono mostrati due casi p = 0.002 e p = 0.003.

Caso 1) $p = 0.002\ (40/2000)$

- Numero di archi: 400010
- Numero di triangoli: 10582
- Comunità: 8
- Coefficiente di clustering: 0.0019
- Componenti: 1
- Diametro: 4

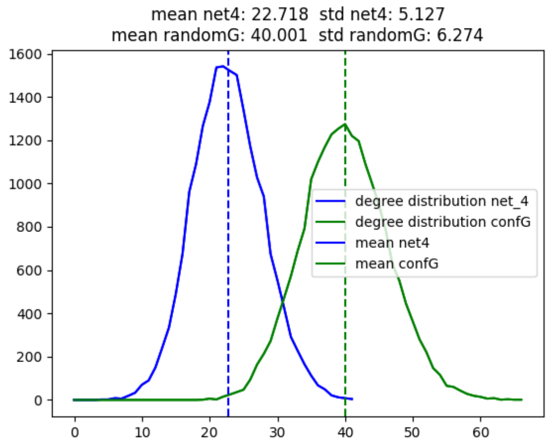

Figura 3. Confronto tra la distribuzione dei gradi di net4 del grafo generato dall'algoritmo randomG con p = 0.002

Caso 2) $p = 0.003\ (60/2000)$

- Numero di archi: 600422
- Numero di triangoli: 36084
- Comunità: 6
- Coefficiente di clustering: 0.003
- Componenti: 1
- Diametro: 4

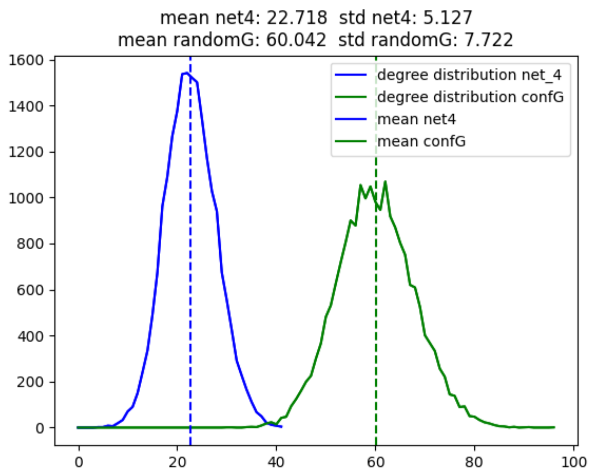

Figura 4. Confronto tra la distribuzione dei gradi di net4 del grafo generato dall'algoritmo randomG con p = 0.003

Si può osservare, che anche se il numero di triangoli sta aumentando, la distribuzione tende a traslare verso destra, perdendo una delle due caratteristiche in comune con il modello net4. Oltretutto il diametro e il numero di comunità si riduce in quanto si sta rendendo più denso il modello.

Tabella 1. Tabella riassuntiva del comportamento di randomG

NUMERO DI ARCHI	✓
NUMERO DI TRIANGOLI	
COMUNITA	
COEFFICIENTE DI CLUSTERING	
COMPONENTI	✓
DIAMETRO	
DISTRIBUZIONE DEI GRADI	✓

Configuration Graph

L'algoritmo configuration graph cerca di creare un grafo a partire da un elenco di gradi definito per ogni nodo. Analogamente a ciò che è stato fatto durante lo studio dell'algoritmo randomG, è stato proposto di individuare una lista di gradi che avrebbe permesso di ottenere una distribuzione che si avvicina a quella di net4. Visto che la distribuzione dei gradi è molto simile a quella di una gaussiana, si è deciso di campionare 20mila campioni (numero di nodi di net4) da una gaussiana con media e deviazione standard della distribuzione di net4.

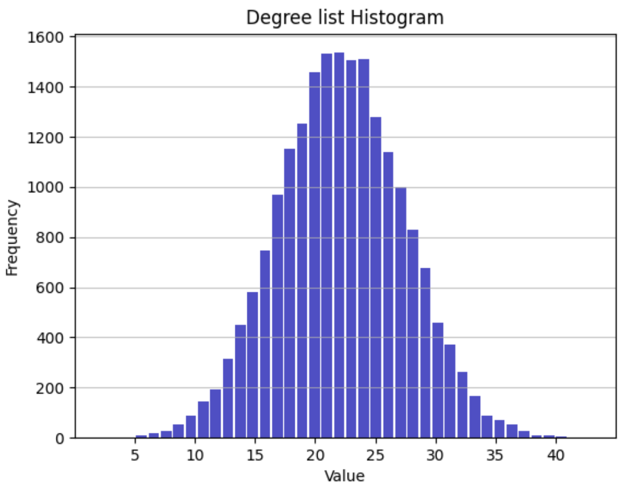

Figura 5. Istogramma costruito dai campioni di una gaussiana con media 22.718 e deviazione standard 5.127

Di seguito mostriamo i risultati ottenuti:

- Numero di archi: 221887
- Numero di triangoli: 2100
- Comunità: 8
- Coefficiente di clustering: 0.001
- Componenti: 1
- Diametro: 5

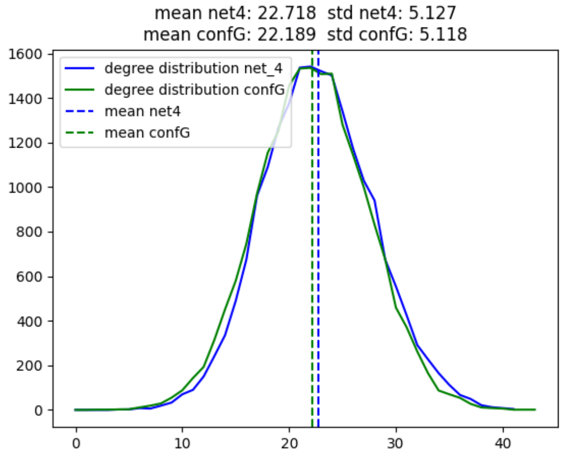

Figura 6. Confronto tra la distribuzione dei gradi di net4 e quella generato dall'algoritmo configurationG

Come nel caso precedente il numero di triangoli, il diametro, il numero di comunità e il coefficiente di clustering sono troppo bassi.

Tabella 2: tabella riassuntiva del comportamento di configurationG

NUMERO DI ARCHI	✓
NUMERO DI TRIANGOLI	
COMUNITA	
COEFFICIENTE DI CLUSTERING	
COMPONENTI	✓
DIAMETRO	
DISTRIBUZIONE DEI GRADI	✓

Watts-Strogatz Graph

L'algoritmo di Watts-Strogatz è diviso in due fasi: la prima si occupa della creazione degli strong-tie mentre la seconda dei weak-tie. I parametri utilizzati dall'algoritmo sono i seguenti:

- **r**: rappresenta il raggio, ad ogni nodo è assegnato una posizione casuale estratta da una distribuzione uniforme, se la distanza tra un generico nodo *i* e un generico nodo *j* è inferiore a *r* viene creato un arco.
- **k**: definisce il numero di weak-tie da creare dal nodo *i.*
- **q**: regola la probabilità di creazione di un weak-tie.

Studio dei parametri

Nei paragrafi successivi verranno spiegati i parametri e la loro influenza su alcune misure.

Parametro r

Il parametro r determina il numero di strong-tie da creare. Aumentando il raggio si può ottenere un considerevole aumento del numero di triangoli, questo perché aumenta la probabilità che i nodi vicini siano collegati.

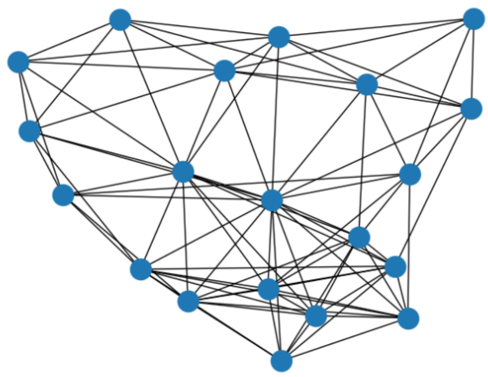

Figura 7. $r = 2, k = 2, q = 2$. *Numero di triangoli 163*

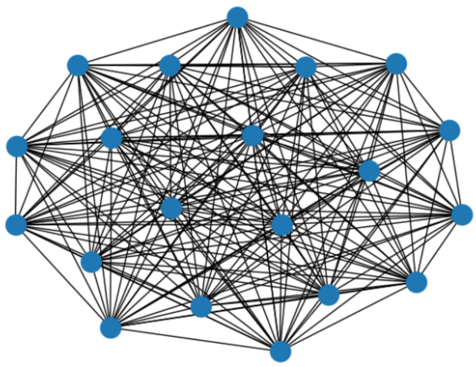

Figura 8. $r = 10, k = 2, q = 2$. *Numero di triangoli 1140*

Parametro k

Il parametro k è determinante sul diametro che avrà il grafo generato dall'algoritmo. Aumentando il numero di weak-tie più basso sarà il diametro, viceversa diminuendo il parametro k maggiore sarà il diametro.

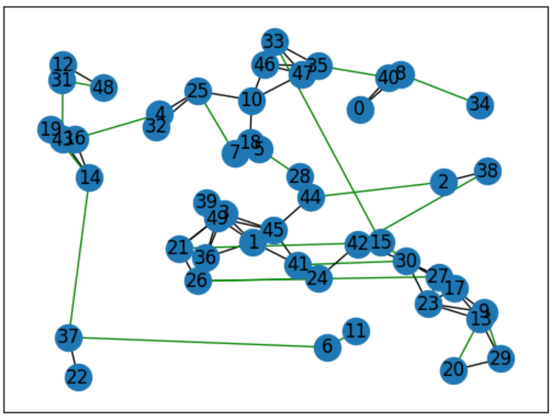

Figura 9. $k = 2$. *Diametro 7*

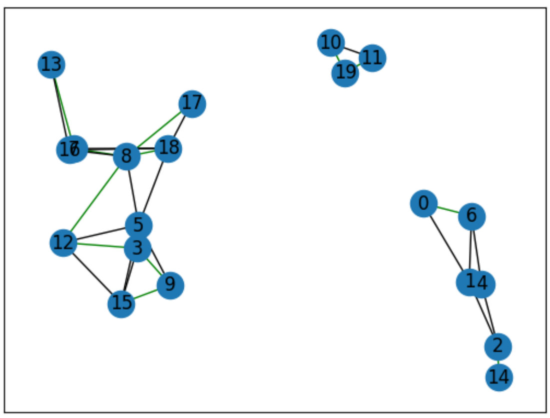

Figura 10. $k = 1$. *Diametro 13*

Parametro q

Il parametro q permette di regolare la creazione di weak-tie incidendo sulla distanza tra i nodi. In altre parole, aumentando q è come se si aumentasse la distanza tra ogni coppia nodi e visto che ogni nodo nella seconda fase dell'algoritmo viene estratto con probabilità $\frac{1}{d(i,j)^q}$, se la distanza tra i nodi era già molto grande questa probabilità all'aumentare di q diventa ancora più bassa.

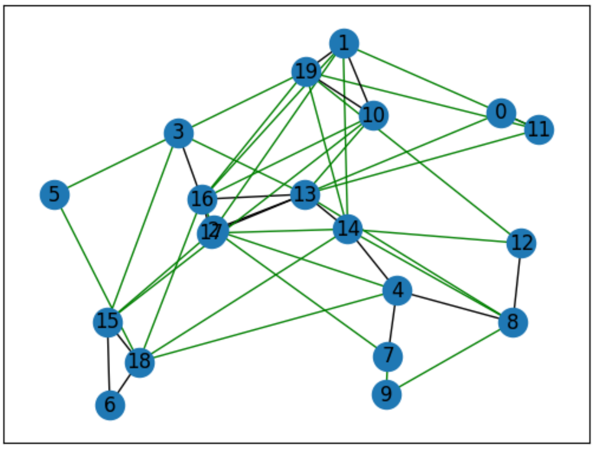

Figura 11. $q = 1$*. Numero di weak-tie 35*

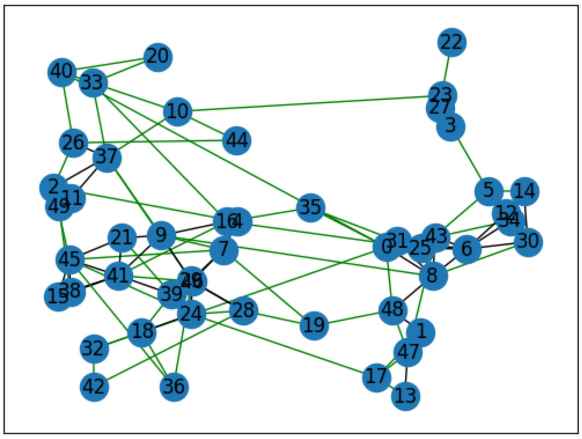

Figura 12. $q = 4$*. Numero di weak-tie 23*

Settaggio dei parametri ed esperimenti

Il primo parametro scelto da stimare è stato r. Per la stima di r è stato realizzato un algoritmo denominato nel progetto **approximation_r** (il file contenente l'algoritmo risiede nella directory *colab_ws*)**.** Nella fase iniziale l'algoritmo genera per ogni nodo una posizione casuale nello stesso modo di Watts-Strogatz, successivamente sceglie un nodo a caso i tra tutti i possibili nodi e calcola le distanze tra i e tutti gli altri nodi j *con* $i \neq j$*.* L'algoritmo, infine, ordina le distanze calcolate in ordine crescente e sceglie la 22-esima, in modo tale da poter rispettare la distribuzione dei gradi di net4. L'algoritmo è stato lanciato più volte ed ha generato valori compresi all'incirca tra 2.6 e 2.8.

Il secondo parametro k inizialmente è stato settato a zero, in quanto il diametro di net4 è molto alto.

Il parametro q inizialmente è stato settato a due per valutarne l'effetto.

Tabella 3. $k = 0, q = 2$

r	**archi**	**triangoli**	**diametro**	**componenti**	**comunità**	**Coefficiente di clustering**
2.6	209756	870206	88	1	43	0.593
2.7	225639	1004851	83	1	40	0.593
2.8	242314	1158450	82	1	38	0.592

Il numero di triangoli e il numero di archi per il raggio 2.6 è molto inferiore rispetto a quello di net4, mentre per 2.8 il numero di triangoli e il numero di archi è più grande di quello di net4.

Il numero di comunità per ogni raggio è molto vicino a quelle di net4. Per r = 2.7 il numero di comunità coincide con quello di net4.

In tutti e 3 i casi il diametro è più grande di quello di net4, per questo motivo è stato scelto di aumentare k portandolo ad 1.

Tabella 4. $k = 1, q = 2$.

r	archi	triangoli	diametro	componenti	comunità	Coefficiente di clustering
2.6	219022	867778	9	1	37	0.541
2.7	235899	1017951	10	1	33	0.545
2.8	252197	1167407	9	1	33	0.55

Dalla tabella 4 viene evidenziato che il diametro è diminuito drasticamente. Quindi, la sola cosa che resta da fare per provare ad aumentarlo è quello di sfruttare il parametro q.

Dalla tabella 3 si deduce, invece, che un raggio pari a 2.7 ci permette di ottenere un buon risultato, mentre dalla tabella 4 un valore di q pari a 2 non è purtroppo soddisfacente. Per i motivi appena citati è stato scelto di ripetere l'esperimento con un raggio compreso tra 2.7 e 2.72 e il parametro q pari a 4. Per ognuno di questi raggi è stato generato un modello dieci volte, sono state calcolate tutte le misure ed è stata infine effettuata una media.

Tabella 5. k=1, q=4.

r	archi	triangoli	diametro	componenti	comunità	Coefficiente di clustering
2.7	225794.8	1004118.4	64	1	39.3	0.617
2.71	227576.1	1020343.8	64.8	1	38.4	0.591
2.72	229367.9	1036579.5	64.1	1	38	0.591

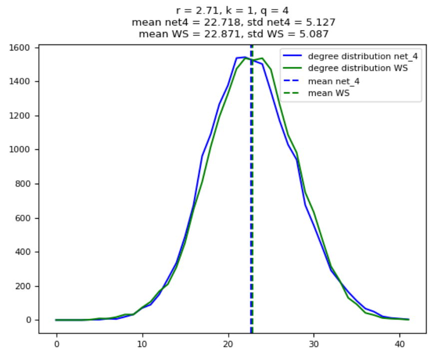

Figura 11. $r = 2.71, k = 1, q = 4$

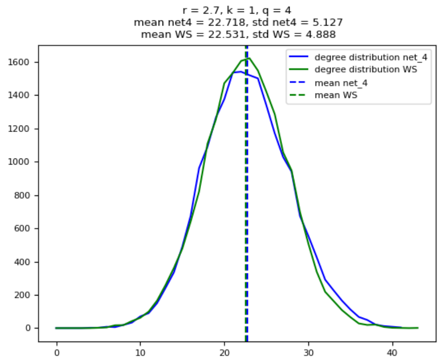

Figura 12. $r = 2.7, k = 1, q = 4$

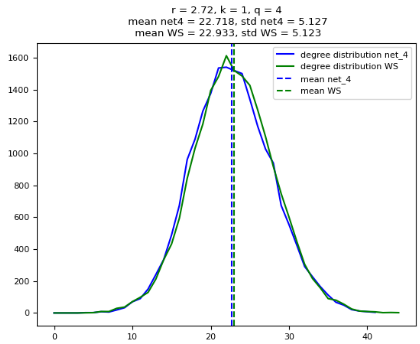

Figura 13. $r = 2.72, k = 1, q = 4$

La soluzione con r pari a 2.71 è quella che si avvicina di più alla rete net4 in termini di triangoli, coefficiente di clustering, numero di triangoli, diametro e distribuzione dei gradi. Da ciò si è successivamente calcolato il raggio e il vero diametro (non approssimato), ottenendo rispettivamente 36 e 70.

Tabella 6. Soluzione finale

modelli	archi	triangoli	componenti	comunità	Coefficiente di clustering
Net4	227183	1017162	1	40	0.590
Soluzione trovata	227576.1	1020343.8	1	38.4	0.591

Dalle analisi fatte abbiamo concluso che il modello utilizzato è **Watts-Strogatz** con i seguenti parametri:

- **r** = 2.71
- k = 1
- **q** = 4