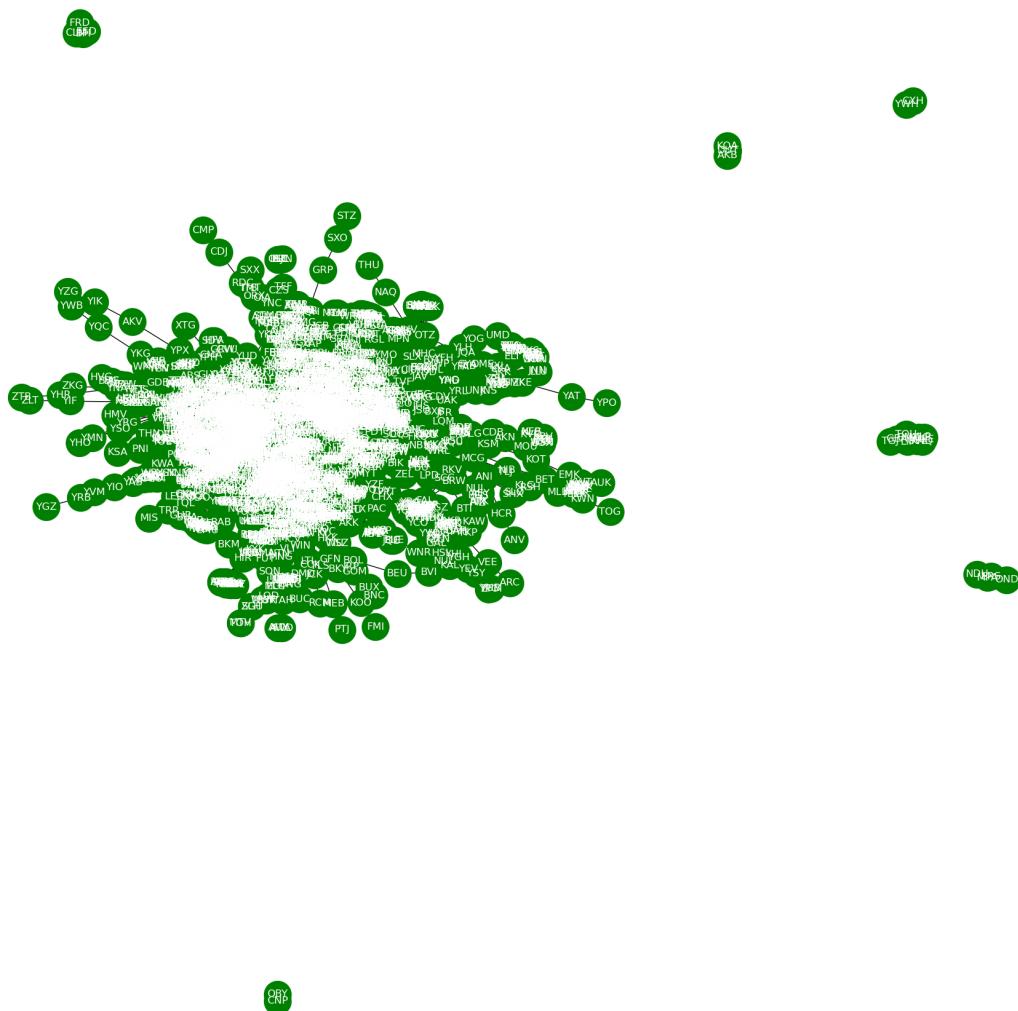


Progetto Network Science

Il progetto consiste nell'analisi di un dataset relativo alla rete aeroportuale mondiale

Struttura della rete



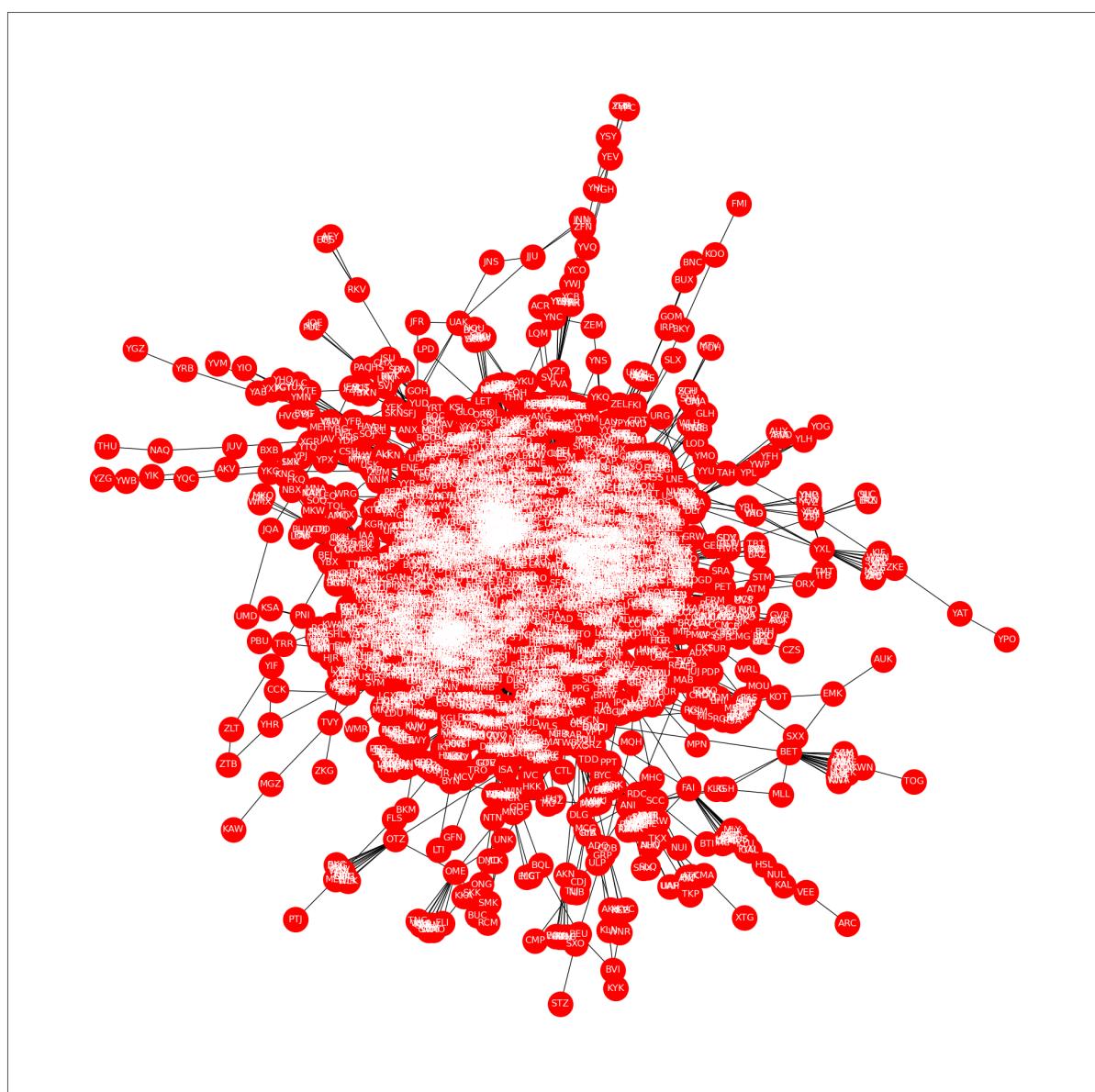
Anche se si capisce poco, questa è la struttura della rete aeroportuale mondiale, si nota decisamente la presenza di molteplici componenti connesse

Difatti se ne individuano 7

Le 6 più piccole sono:

- $\{\text{'CNP}', \text{'OBY'}\}$:
 - $\{\text{'BMY}', \text{'GEA}', \text{'ILP}', \text{'KNQ}', \text{'KOC}', \text{'LIF}', \text{'MEE}', \text{'TGJ}', \text{'TOU}', \text{'UVE'}\}$:
 - $\{\text{'BFI}', \text{'CLM}', \text{'ESD}', \text{'FRD'}\}$:
 - $\{\text{'AKB}', \text{'DUT}', \text{'KQA'}\}$:
 - $\{\text{'CXH}', \text{'YWH'}\}$:
 - $\{\text{'ERS}', \text{'MPA}', \text{'NDU}', \text{'OND'}\}$

E poi ce n'è una decisamente più grande (non riportata testualmente perché troppo grande, la si riporta graficamente qui sotto)



La rete ovviamente non è una rete diretta, quindi non ha senso chiedersi se sia fortemente o debolmente connessa

Densità della rete

La densità della rete si riferisce alla quantità di connessioni presenti nella rete rispetto al numero totale di connessioni possibili tra tutti i nodi della rete.

```
nx.density(G)
✓ 0.0s
0.0036854903095776227
```

Se la densità della rete è bassa, significa che solo una piccola percentuale delle connessioni possibili è effettivamente presente nella rete. In altre parole, ci sono molti nodi nella rete che non sono collegati tra di loro.

Coeff di clustering

Calcolo del coefficiente medio di clustering per la rete

```
nx.average_clustering(G)
✓ 0.5s
0.4905417456798235
```

Il coefficiente di clustering è la misura del grado in cui i nodi di un grafo tendono ad essere connessi fra loro

Un coefficiente di clustering medio pari circa a 0.5 indica che la rete aeroportuale mondiale ha un raggruppamento moderato.

Nel contesto dell'aeroporto, il coefficiente di clustering riflette il modo in cui gli aeroporti sono collegati tra loro. Un valore di 0.5 suggerisce che gli aeroporti sono organizzati in gruppi o comunità relativamente coese, ma non necessariamente completamente interconnesse. Ci possono essere sottoreti di aeroporti che sono ben collegati internamente, ma hanno connessioni meno forti con altri gruppi di aeroporti

Avere un coefficiente di clustering alto e una densità molto bassa è normale in una rete, questo indica che la rete è fatta a “grappoli” e internamente i grappoli sono molto densi

Per le due sotto-comunità USA-Europa il coefficiente di clustering è pari a

USA

Europa

```
nx.average_clustering(G)
0.49304538688224697
```

```
nx.average_clustering(european_graph)
✓ 0.1s
0.5740702290373797
```

Individuazione Hub

Grado

In network science, un hub è un nodo all'interno di una rete che ha un alto grado di connettività rispetto agli altri nodi

In pratica, un hub è un nodo che è connesso a molti altri nodi e che gioca un ruolo importante nel trasferimento di informazioni o nella diffusione di influenza all'interno della rete stessa

Nel caso in esame, il nodo con grado più elevato è: *AMS - Amsterdam-Schiphol, il principale aeroporto dei Paesi Bassi* con grado 246

```
high_deg_node = max(G.nodes, key=G.degree)
print(high_deg_node, "grado = ", G.degree(high_deg_node))
✓ 0.1s
AMS grado = 246
```

Betweenness

Il grado di un nodo in un grafo indica semplicemente il numero di archi che incidono su quel nodo, ovvero il numero di connessioni che ha con altri nodi della rete.

La centralità di un nodo invece è una misura più complessa che tiene conto della posizione del nodo all'interno della rete e del ruolo che esso svolge nella comunicazione tra gli altri nodi.

La betweenness, ad esempio, misura la frequenza con cui un nodo è coinvolto nei percorsi più brevi tra tutti gli altri nodi della rete.

Quindi un nodo con alta betweenness è un nodo che svolge un ruolo di ponte tra diverse parti della rete e che, se rimosso, potrebbe causare un notevole impatto sulla connettività della rete stessa.

Il grado di un nodo può essere utile per capire il numero di connessioni che esso ha con gli altri nodi della rete, ma non tiene conto della posizione e del ruolo del nodo nella rete.

La betweenness, invece, è una misura più sofisticata che tiene conto anche della posizione del nodo e del suo ruolo nella comunicazione tra gli altri nodi, risultando quindi più efficace per valutare la centralità dei nodi in una rete.

Infatti analizzando la betweenness la situazione cambia e si nota che l'aeroporto più centrale è *CDG - Aeroporto di Parigi-Charles de Gaulle*

```
betweenness = nx.centrality.betweenness_centrality(G)
high_betw_mode = max(G.nodes, key= betweenness.get)
print("Beetwenness massima -> ", high_betw_mode, betweenness[high_betw_mode])
✓ 1m 39.8s
Beetwenness massima -> CDG 0.06470661485997284
```

I valori di betweenness per i nodi possono assumere un'ampia gamma di valori, da 0 (nessun cammino più breve passa attraverso il nodo) a 1 (il nodo si trova su tutti i cammini più brevi tra tutte le coppie di nodi della rete)

```
print("Grado CDG = ", G.degree("CDG"))
✓ 0.0s
Grado CDG = 239
```

Questo è il grado dell'aeroporto Charles de Gaulle, si nota che è alto (quindi è un nodo abbastanza centrale), ma allora perchè la betweenness è bassa?

La risposta si può intuire ricordando che la rete ha una densità bassa

La bassa betweenness dell'aeroporto Charles de Gaulle potrebbe essere normale, poiché potrebbe non essere possibile raggiungere direttamente tutti gli aeroporti del mondo da questo aeroporto. La betweenness dipende dal numero di shortest path che passano attraverso un determinato nodo in una rete, e se non ci sono molti

shortest path che passano attraverso l'aeroporto di Charles de Gaulle, allora la sua betweenness sarà bassa

Se in una rete il nodo con la massima betweenness ha una betweenness tendente a 0 ma un grado alto, ciò potrebbe indicare che il nodo in questione ha molti collegamenti diretti con altri nodi nella rete, ma questi non sono posizionati in modo tale da fare passare la maggior parte dei cammini più corti attraverso di essi. In altre parole, il nodo potrebbe avere molti vicini, ma questi non sono centrali nella rete. Di conseguenza, nonostante abbia un alto grado, il nodo non è particolarmente centrale nella rete in termini di betweenness.

La presenza di un tale nodo nella rete suggerisce che potrebbe esserci una struttura gerarchica o modulare all'interno della rete, con gruppi di nodi altamente connessi che interagiscono principalmente tra loro. Cosa che rispecchia il ragionamento fatto precedentemente per densità e coefficiente di clustering

Grado medio e mediano

Il grado medio può fornire informazioni sulla densità della rete: se il grado medio è alto, significa che i nodi della rete sono connessi con molti altri nodi, il che indica una rete densa

I grado mediano può fornire informazioni sulla distribuzione dei gradi nella rete: se il grado mediano è alto, significa che molti nodi della rete hanno un alto grado, mentre se il grado mediano è basso, significa che la maggior parte dei nodi ha un basso grado.

```
degree_sequence = [G.degree(n) for n in G.nodes]
import statistics
print("Grado medio: ", statistics.mean(degree_sequence))
print("Grado mediano: ", statistics.median(degree_sequence))
✓ 0.1s
Grado medio: 11.712488203837685
Grado mediano: 3
```

Si nota che sia il grado medio che il grado mediano sono estremamente più piccoli rispetto al grado massimo trovato precedentemente (AMS: 246)

Grado medio basso \Rightarrow rete non densa

Grado mediano basso \Rightarrow tanti nodi della rete hanno grado basso

Questo vuol dire che ho tanti aeroporti con grado basso e pochi con grado alto o estremamente alto, in più il grado mediano mi dice che, prendendo la lista dei gradi e ordinandola in ordine crescente, al centro ho esattamente grado 3, il che vuol dire che almeno la metà degli aeroporti ha grado ≤ 3

Questi dati possono aiutare a spiegare ulteriormente la betweenness bassa del nodo con maggior betweenness. Infatti, se un aeroporto ha un grado basso significa che ha meno connessioni con altri aeroporti nella rete, il che può limitare le possibilità di raggiungere determinati aeroporti attraverso di esso. Quindi si necessiterà di cammini più lunghi per raggiungere determinati aeroporti

```
print("Numero di archi: ", G.number_of_edges())
print("Numero di nodi: ", G.number_of_nodes())
✓ 0.0s
Numero di archi: 18617
Numero di nodi: 3179
```

Il numero di nodi nella rete è 3179, 11.7 archi in media per ogni nodo è un valore che probabilmente fa aumentare la distanza dei cammini

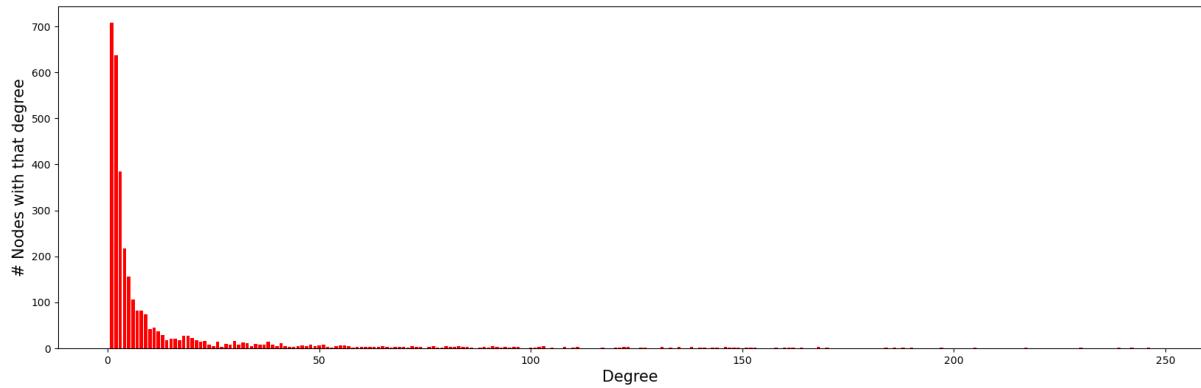
Betweenness media e mediana della rete

```
betweenness = nx.centrality.betweenness_centrality(G)
betweenness_sequence = list(betweenness.values())
print("Betweenness media: ", statistics.mean(betweenness_sequence))
print("Betweenness mediana: ", statistics.median(betweenness_sequence))
✓ 1m 59.0s
Betweenness media: 0.0009173778947573667
Betweenness mediana: 8.753059219980413e-08
```

Si nota che entrambe sono molto basse

Se i valori di betweenness media e mediana sono molto bassi nella rete, ciò potrebbe indicare che questa è relativamente poco connessa o che la struttura della rete è altamente centralizzata con pochi nodi che sono particolarmente importanti per la connessione degli altri nodi.

Plot della distribuzione del grado dei nodi

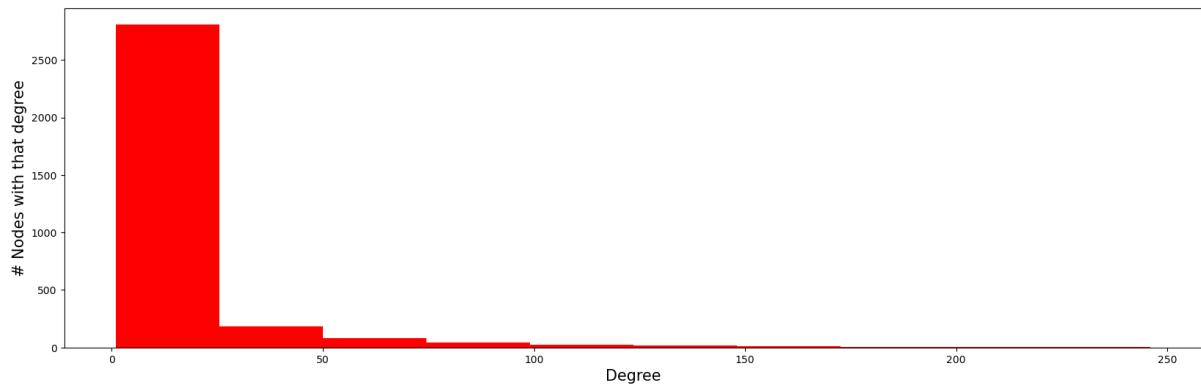


Si nota che ci sono tantissimi nodi con grado estremamente basso e pochissimi nodi con grado medio-alto-estremamente alto

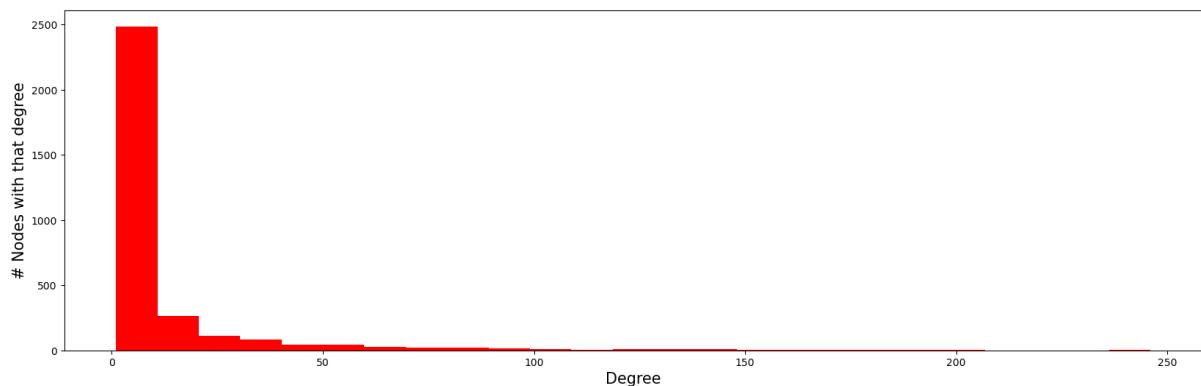
Questa è un'ulteriore conferma delle analisi fatte fino ad ora

Per una migliore visualizzazione si è fatto un plot tramite bins

10 bin, ogni bin contiene 25 gradi → si nota come i nodi con grado compreso fra 0 e 25 siano la maggioranza



25 bin, ogni bin contiene 10 gradi → si nota come i nodi con grado compreso fra 0 e 10 siano tantissimi rispetto agli altri



Closeness centrality

Un'altra importante misura di centralità è la closeness centrality

E' una misura di centralità in una rete calcolata come il reciproco della somma delle lunghezze dei percorsi più brevi tra il nodo e tutti gli altri nodi del grafo. In parole povere, più un nodo è vicino ad altri nodi, più è centrale.

Anche in questo caso la situazione cambia rispetto alle due misure precedenti

```
high_centr_node = max(G.nodes, key= centrality.get)
print("Max centrality -> ", high_centr_node, centrality[high_centr_node])
✓ 0.0s
Max centrality -> FRA 0.41090196565991216
```

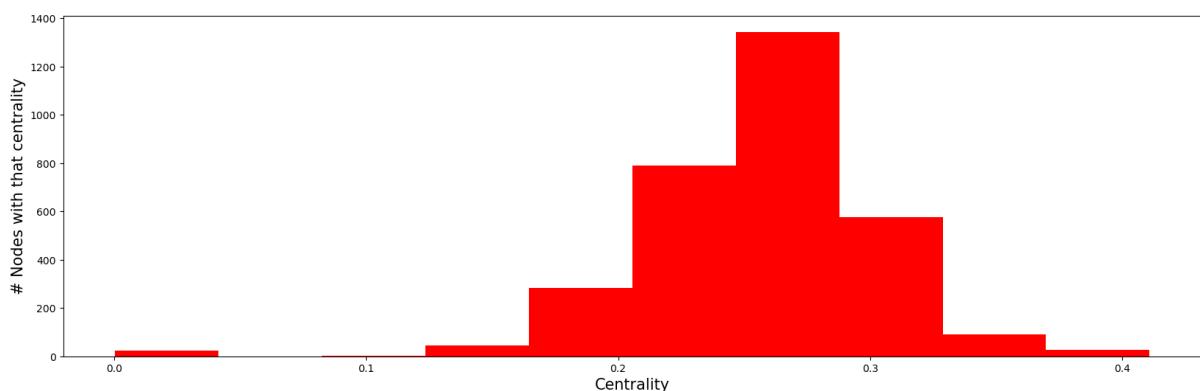
Il nodo più centrale secondo questa metrica è l'aeroporto di Francoforte

Andando a misurare la media e la mediana di tale misura otteniamo

```
print("Centralità media: ", statistics.mean(centr_sequence))
print("Centralità mediana: ", statistics.median(centr_sequence))
✓ 0.0s
Centralità media:  0.25540392153632985
Centralità mediana:  0.2588495378211759
```

Avendo ottenuto un valore medio basso, possiamo dire che i nodi del grafo sono generalmente lontani fra loro, ovvero che la distanza media tra i nodi è relativamente grande

Il che è coerente con quanto detto fino ad ora



Robustezza della rete

Un modo per pensare all'importanza dei nodi in una rete è quello di misurare quanto la loro rimozione danneggierebbe la struttura della rete

Consideriamo due tipi di danno alla rete:

- guasto casuale -> i nodi vengono scelti a caso per essere rimossi
- attacco mirato -> si rimuovono i nodi in base a un criterio, ad esempio rimuovendo i nodi in ordine decrescente rispetto alla loro centralità di grado.

Per misurare il "danno", si misurerà la dimensione della componente连通的 più grande della rete

Fallimento random

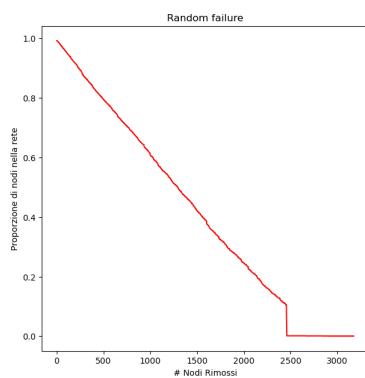
In sostanza, per simulare un fallimento casuale scelgo casualmente alcuni nodi e li rimuovo dal grafo

La simulazione completa sarà:

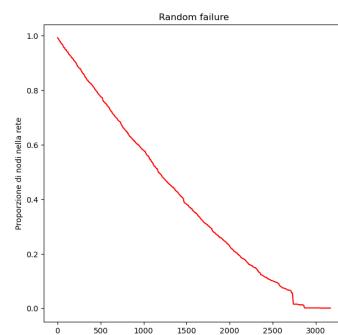
- Misuro la dimensione del core rispetto alla rete originale
- Seleziono M nodi random e li rimuovo
- Ripeto finchè la rete non ha meno di M nodi

Di seguito si riportano alcuni risultati ottenuti

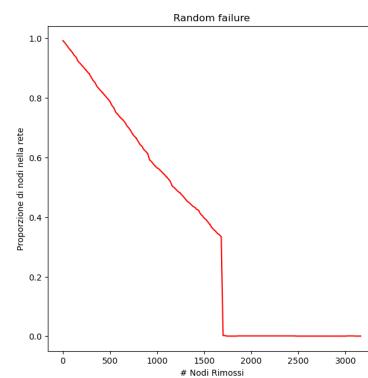
Rimuovendo 5 nodi alla volta:



Rimuovendo 10 nodi alla volta



Rimuovendo 20 nodi alla volta



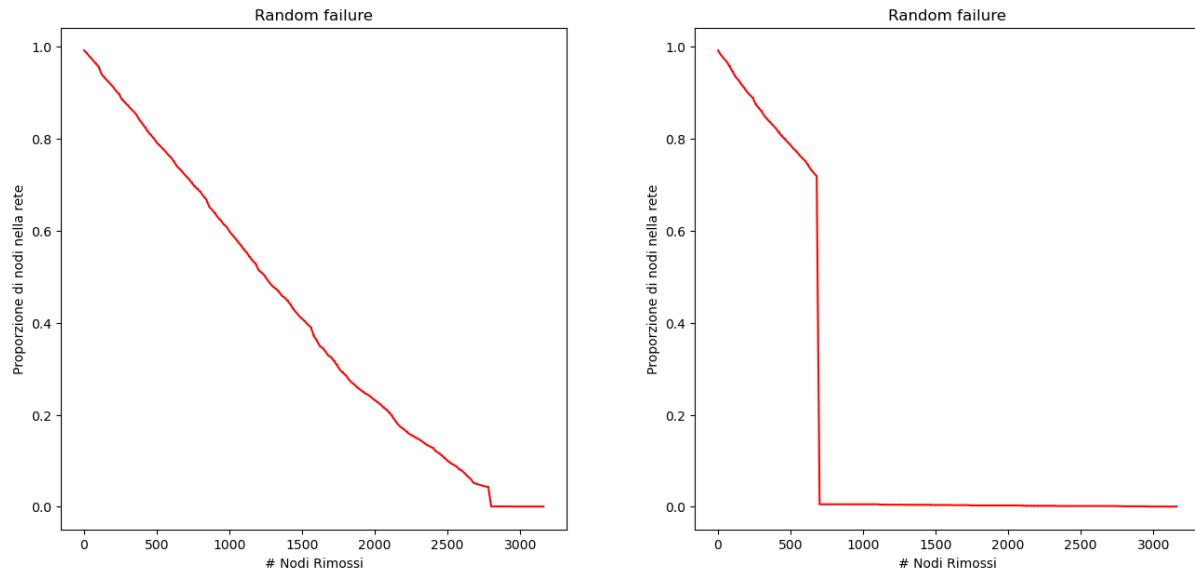
Ovviamente, essendo attacchi randomici, la robustezza della rete dipende da quali nodi vengono rimossi.

Se casualmente vengono rimossi nodi cruciali, allora la rete si distrugge molto rapidamente

In caso contrario la rete è in grado di “reggere” con una discesa più lineare

Con 20 nodi random rimossi

Sempre con 20 nodi random rimossi



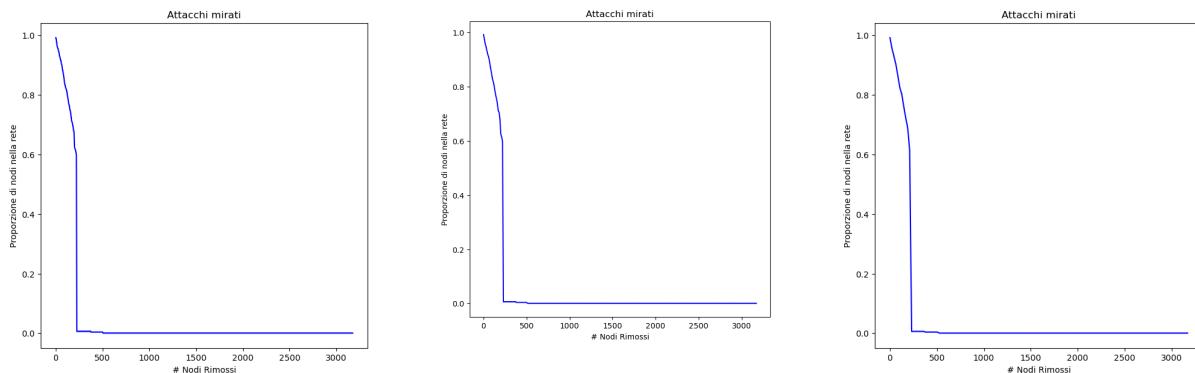
Attacco mirato

L'attacco mirato prevede di andare a scollegare i nodi cruciali della rete

Rimuovendo i 5 top degree nodes

Rimuovendo i 10 top degree nodes

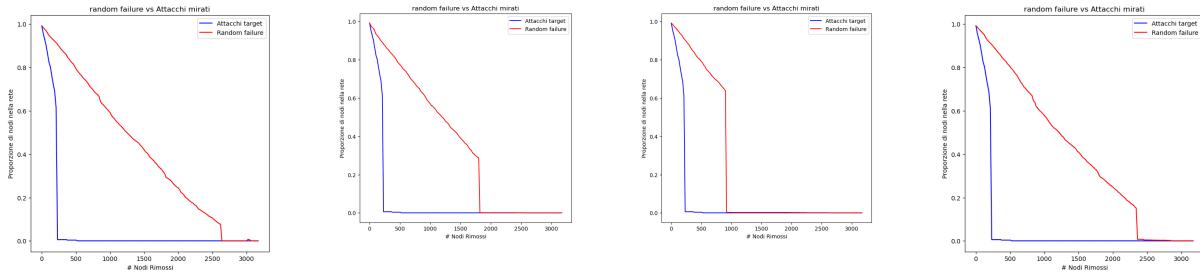
Rimuovendo i 20 top degree nodes



Si nota una sostanziale differenza rispetto agli attacchi randomici, questo è dovuto al fatto che la rete aeroportuale ha pochissimi nodi con grado elevatissimo che, una volta rimossi, fanno collassare completamente la rete

Plot delle differenze

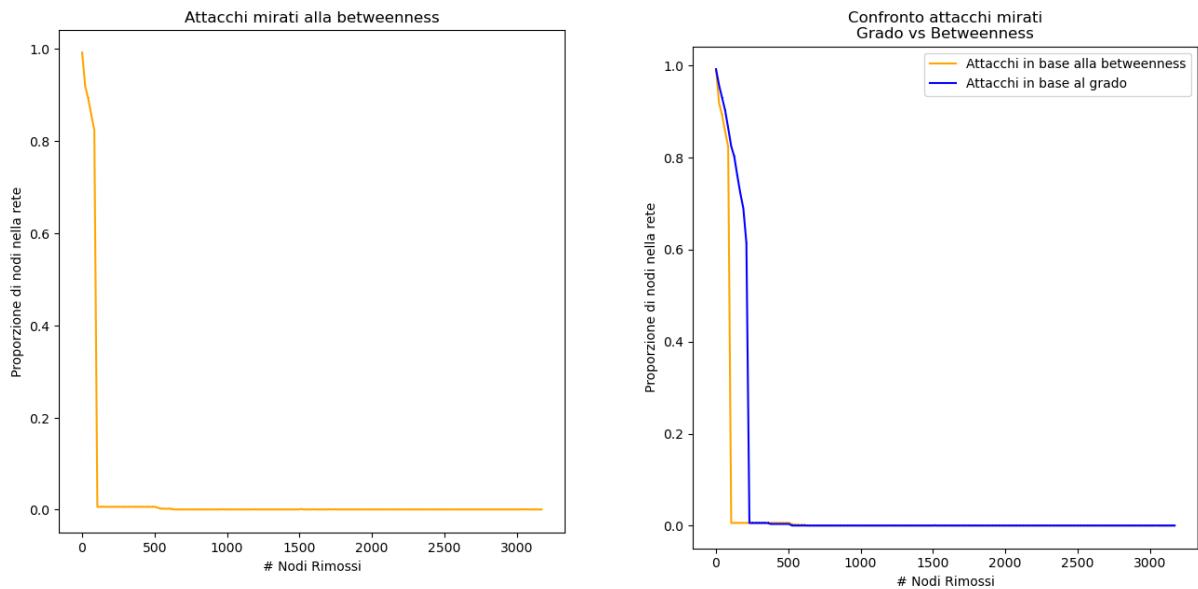
Si sono poi eseguiti diversi run facendo il plot in modo da mostrare la differenza di “drop” tra i due tipi di attacchi (si sono, in entrambi i casi, rimossi sempre 20 nodi ad ogni step di simulazione)



Come si diceva sopra, mentre la curva dell'attacco target sui 20 nodi più centrali rimane sempre uguale, la curva relativa agli attacchi random cambia a seconda dei nodi che vengono selezionati randomicamente nella singola simulazione

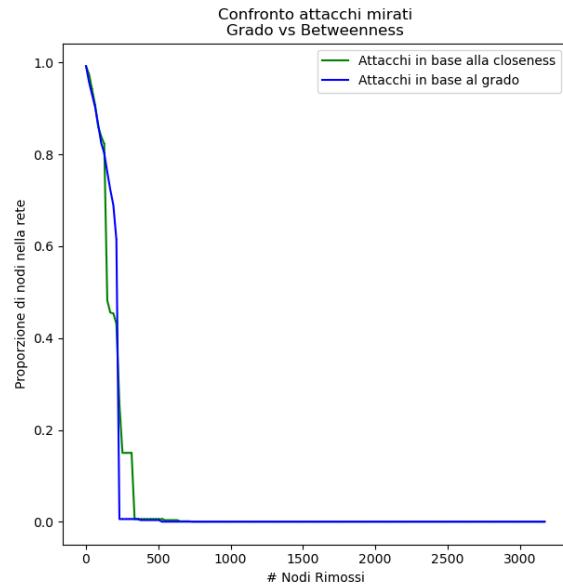
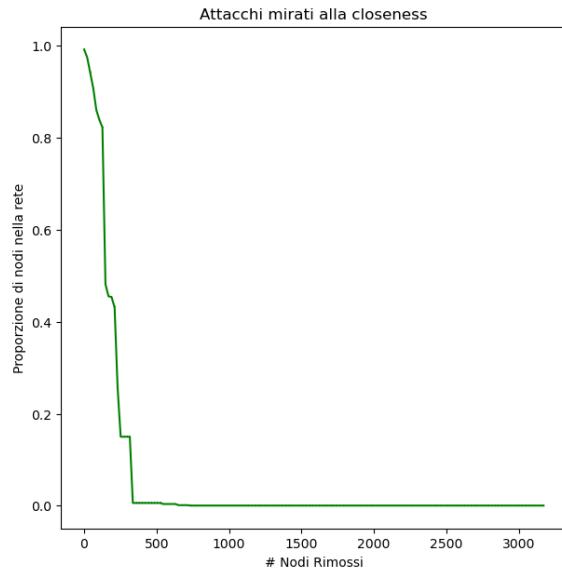
Attacchi mirati alla betweenness

Confronto con attacchi in base al grado

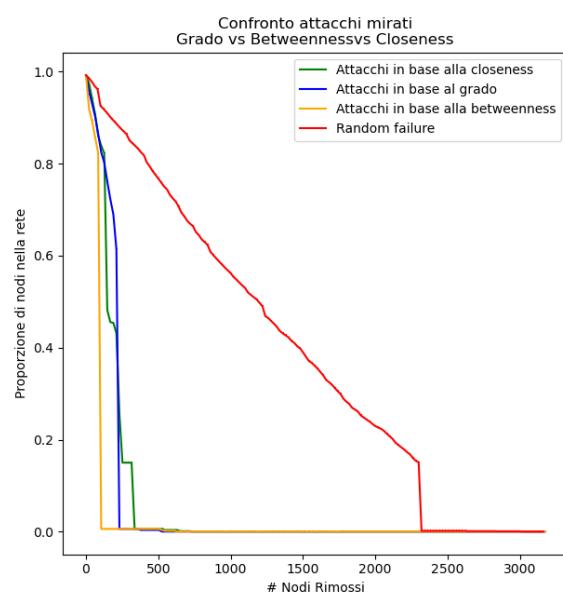
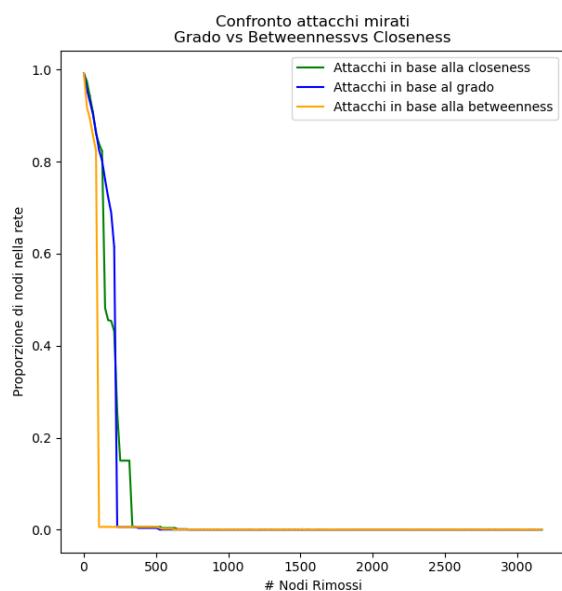


Attacchi mirati in base alla closeness

Confronto con attacchi in base al grado



Confrontototale dei 3 tipi di attacchi mirati



Si nota la differenza di “aggressività” dei 4 tipi di attacchi:

- Random failure: dipende molto dai nodi scelti randomicamente
- Attacchi mirati in base alla closeness: più aggressivo nelle fasi iniziali rispetto all’attacco basato sul grado, ma si raggiunge la disconnessione della rete più tardi rispetto a quest’ultimo
- Attacchi mirati in base al grado: simile, come aggressività, all’attacco basato su closeness, ma si arriva prima alla disconnessione della rete

- Attacchi mirati in base alla betweenness: l'attacco più aggressivo fra i 4

Individuazione comunità USA & Europa

Per motivi di efficienza è stata eseguita sul sottografo relativo alla rete aeroportuale degli USA e dell'Europa

La community detection è il problema di trovare regioni più dense all'interno di una rete, o meglio, dividere i nodi della rete in un numero di gruppi tali per cui il numero di link fra i gruppi sia minimo

Quindi il rilevamento delle comunità di rete consiste nel trovare una partizione che realizzi una buona separazione tra i gruppi di nodi

Community detection basta su network partitioning \neq Data clustering (quest'ultimo si basa sulla similarità fra i nodi)

Modularità

La modularità in network science è una misura della struttura modulare di una rete, ovvero la suddivisione della rete in gruppi di nodi (moduli o comunità) che sono più densamente interconnessi tra loro rispetto ad altri nodi all'interno della rete. In altre parole, la modularità quantifica la presenza di sottoreti o cluster di nodi altamente interconnessi all'interno della rete.

Una comunità può essere definita come un gruppo di nodi altamente connessi tra loro e meno connessi con i nodi esterni alla comunità.

Comunità USA

Tramite sia le funzioni networkX che tramite un'implementazione dell'algoritmo di Girvan-Newman si è ottenuto il seguente risultato, che coincide per i due metodi utilizzati

L'algoritmo di Girvan-Newman è principalmente utilizzato per il network partitioning, ossia la suddivisione di una rete in comunità o cluster. Si concentra sulla struttura della rete stessa e cerca di individuare gruppi di nodi che sono fortemente connessi tra di loro, ma meno connessi con i nodi esterni al gruppo.

La migliore partizione è composta da 132 partizioni

```

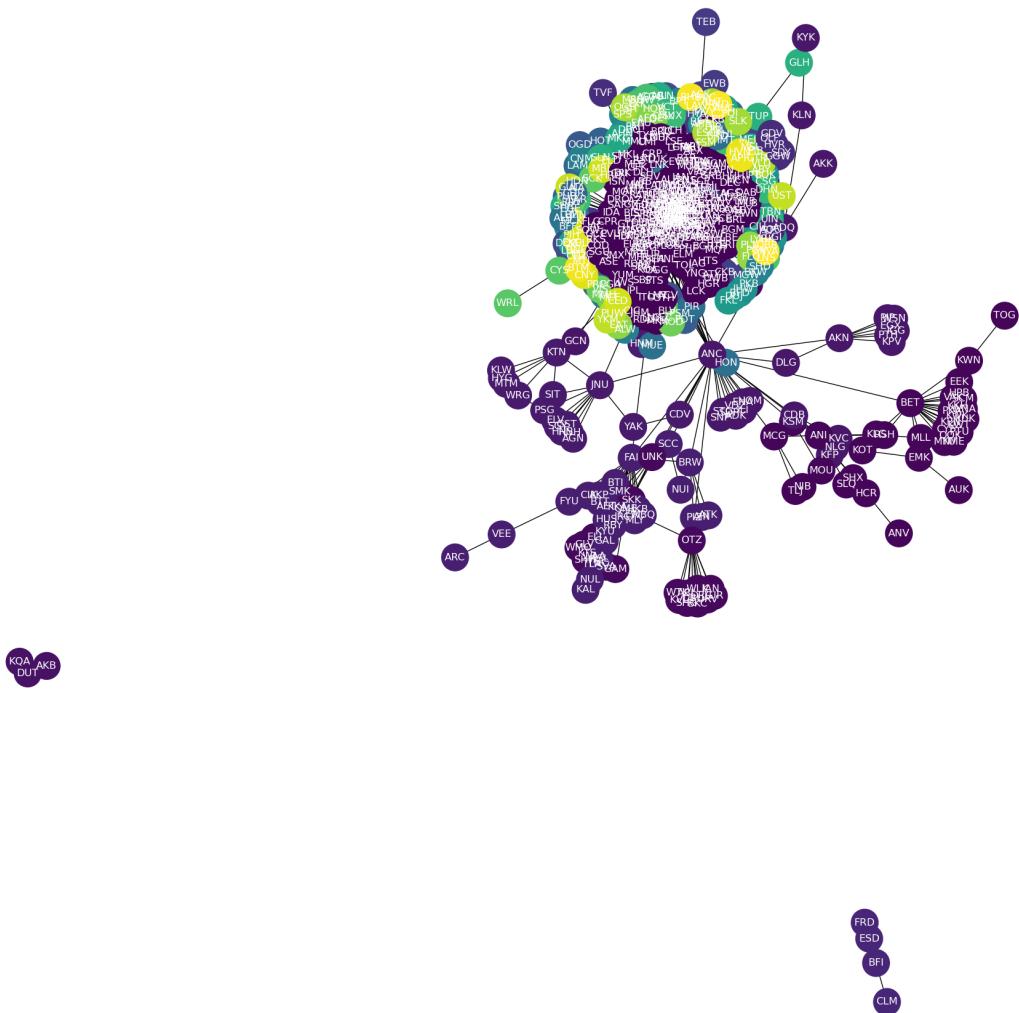
partition_sequence = nx.community.girvan_newman(G)
✓ 0.9s

best_partition = max(partition_sequence, key=my_modularity)
✓ 15m 6.4s

print(len(best_partition))
✓ 0.0s

132

```

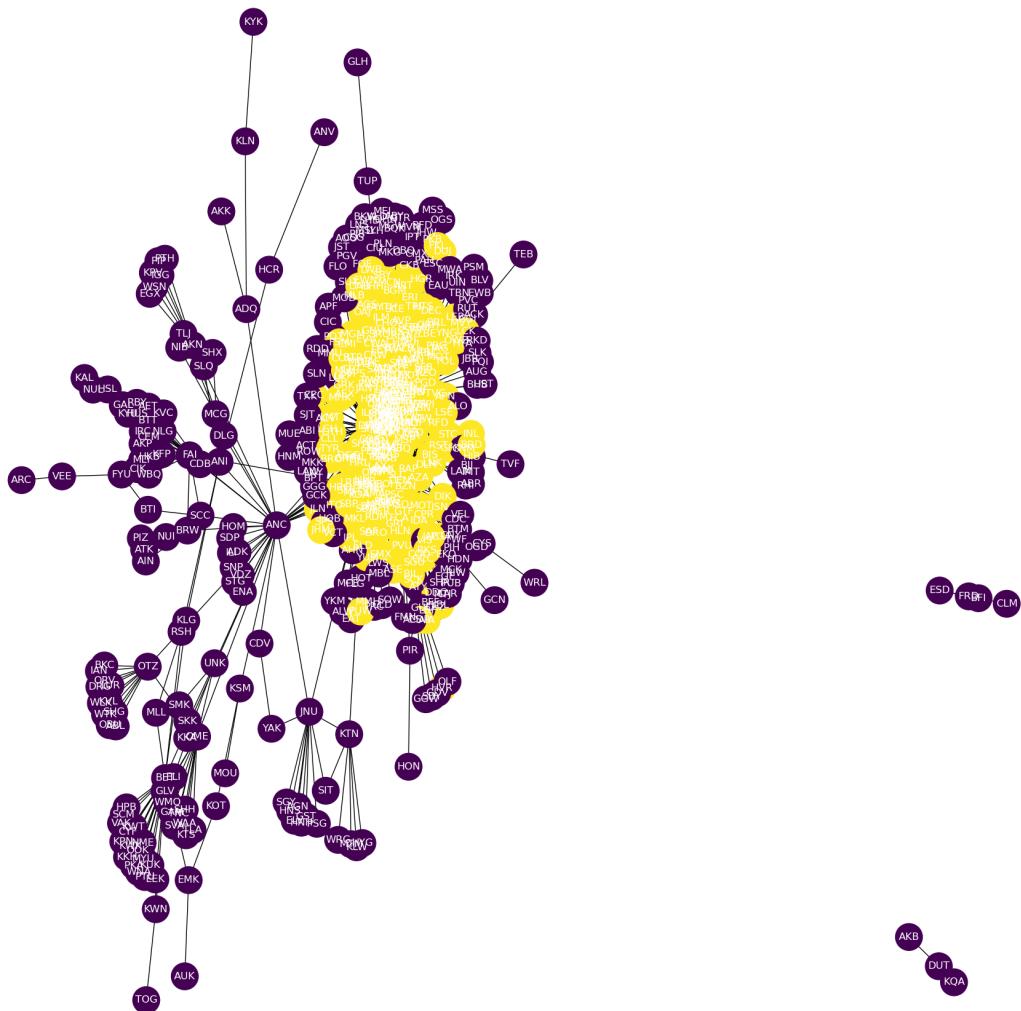


La modularità di tale partizione è stata calcolata tramite

```
nx.community.quality.modularity(G, best_partition)  
✓ 0.0s  
0.1451185880042576
```

Ed è relativamente bassa, una modularità bassa, indica che la rete non è molto divisa in comunità distinte

Partizionamento con kernighan Lin bisection



```
nx.community.quality.modularity(G, partition)  
✓ 0.0s  
0.1327221588006457
```

Comunità Europa

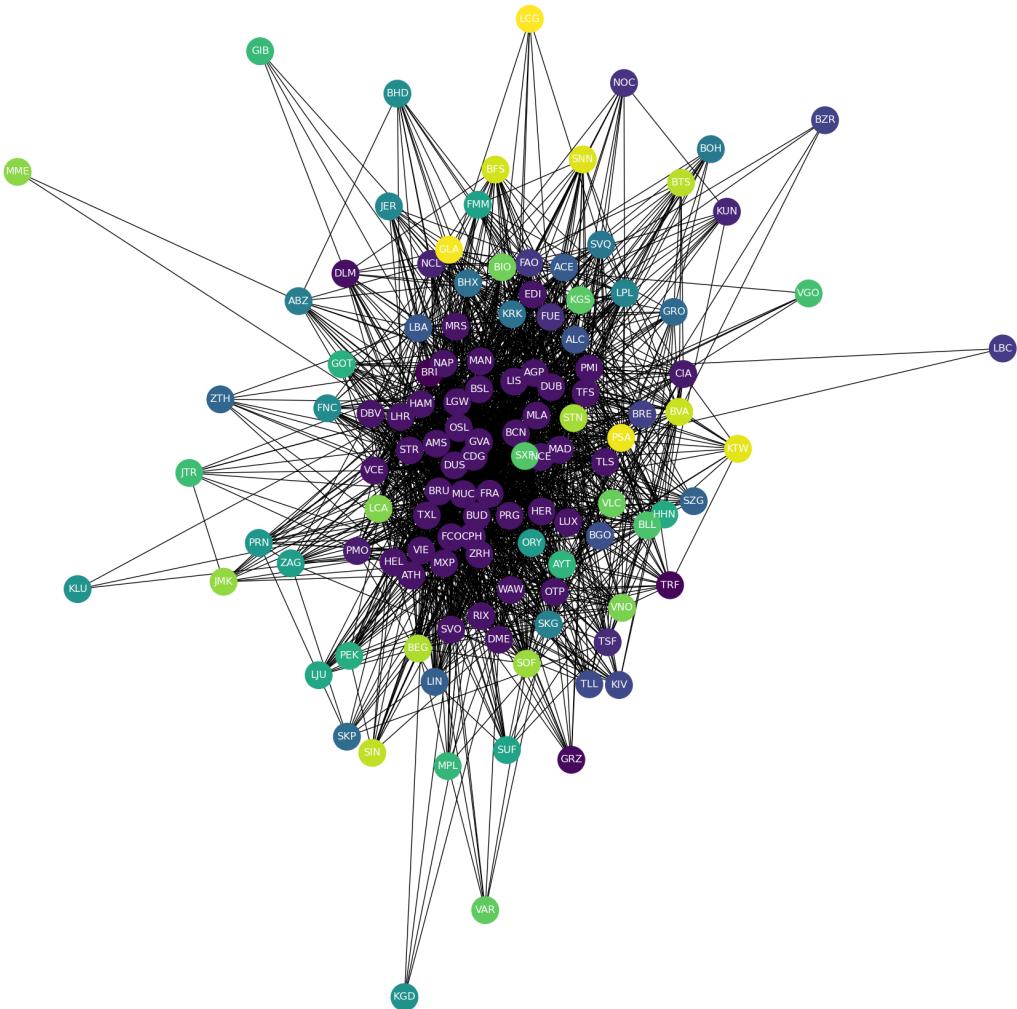
Per ottenere il grafico relativo alla rete europea si sono filtrati i codici IATA degli aeroporti europei dal grafico mondiale

Poi è stata eseguita la stessa analisi svolta precedentemente per gli USA, ottenendo:

La migliore partizione ha 72 partizioni

```
def my_modularity(partition):
    return nx.community.quality.modularity(G, partition)
best_partition = max(partition_sequence, key=my_modularity)
best_partition

print(len(best_partition))
✓ 0.0s
75
```

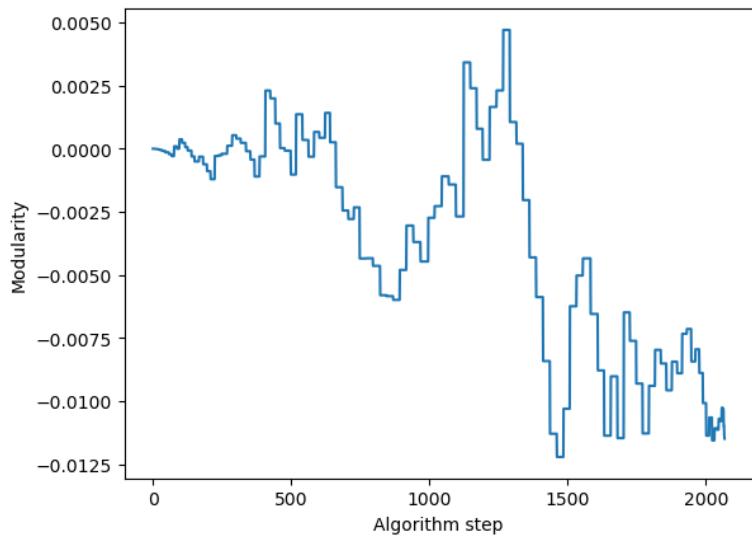


Per quanto riguarda la modularità di tale partizione si è ottenuto

```
nx.community.quality.modularity(G, best_partition)
✓ 0.0s
0.004704060960234013
```

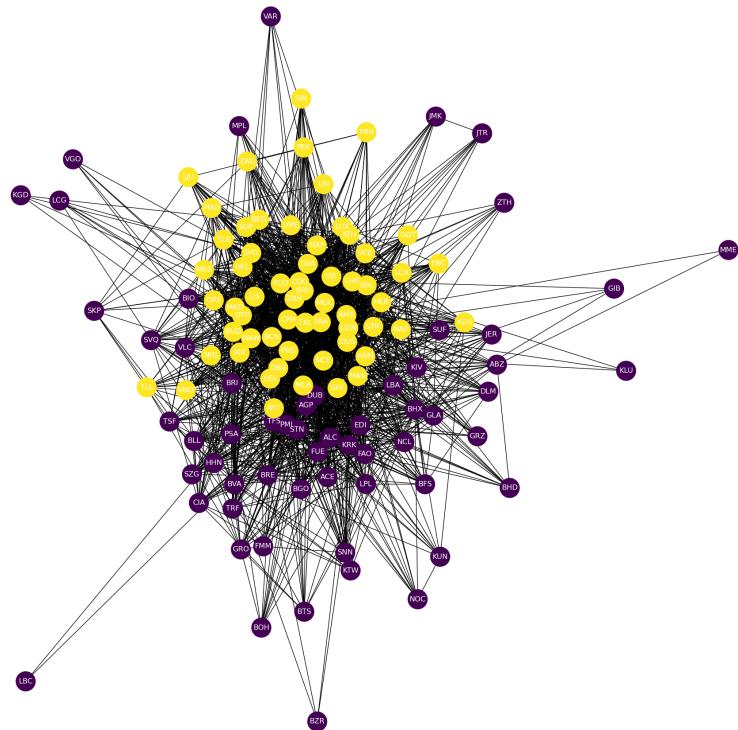
Che è un valore decisamente più basso rispetto a quello ottenuto per gli USA, ciò indica che la struttura aeroportuale europea è ancor meno suddivisa in comunità rispetto a quella statunitense

Andando a valutare le modularità delle partizioni ricavate si è ottenuto il seguente plot



In cui si nota che effettivamente la modularità più elevata è appena sotto il valore 0.0050, proprio come la modularità indicata precedentemente

Partizionamento con kernighan Lin bisection



```

nx.community.quality.modularity(G, partition)
✓ 0.0s
0.10234883871016012

```

Con questo algoritmo si è ottenuta una partizione con modularità più alta (seppur comunque molto bassa) del partizionamento eseguito con Girvan Newman

Relazione coefficiente di clustering & modularità

La modularità e il coefficiente di clustering sono entrambe misure utilizzate nell'analisi delle reti, ma descrivono concetti differenti.

Il coefficiente di clustering è una misura che valuta il grado di raggruppamento o clustering di un insieme di nodi in una rete. Rappresenta la tendenza dei nodi a formare gruppi o comunità interconnesse di nodi **simili** fra loro.

D'altra parte, la modularità è una misura utilizzata per valutare la struttura modulare di una rete. La modularità valuta quanto la rete sia suddivisa in sottoreti o moduli distinti. Un alto valore di modularità indica che la rete è divisa in gruppi distinti, o moduli, con un numero relativamente limitato di connessioni tra i moduli. In altre parole, i nodi all'interno di ciascun modulo sono più densamente collegati tra loro rispetto ai nodi in altri moduli.

La principale differenza tra le due misure è che il coefficiente di clustering si concentra sulle relazioni locali tra i nodi, valutando quanto i vicini di un nodo siano collegati tra loro. Invece, la modularità considera la struttura complessiva della rete e valuta la presenza di sottoreti distinte o moduli.

USA

```

nx.community.quality.modularity(G, best_partition)
✓ 0.0s
0.1451185880042576

```

Europa

```

nx.community.quality.modularity(G, best_partition)
✓ 0.0s
0.004704060960234013

```

```

nx.average_clustering(G)
0.49304538688224697

```

```

nx.average_clustering(european_graph)
✓ 0.1s
0.5740702290373797

```

Questa combinazione di un alto coefficiente di clustering e una bassa modularità potrebbe indicare una rete aeroportuale caratterizzata da gruppi localmente coesi di

aeroporti, ma con una connettività più uniforme e meno strutturata a livello globale. In altre parole, mentre esistono raggruppamenti locali di aeroporti ben collegati tra loro, non ci sono divisioni chiare o modelli distinti di connessione tra questi raggruppamenti. La rete potrebbe essere più uniformemente interconnessa su scala globale.

L'alto coefficiente di clustering indica che c'è un'alta similarità fra i nodi della rete, e ciò ha senso per la rete aeroportuale delle due comunità analizzate se consideriamo la similarità come il numero di vicini in comune fra due nodi. Ad esempio, posso raggiungere l'aeroporto di Barcellona da diversi altri aeroporti (Milano, Madrid, Parigi, ...). Quindi effettivamente molti nodi hanno dei vicini in comune e quindi sono simili, il che porta ad un coefficiente di clustering alto

Al contrario, la modularità della rete è bassa, quindi non si riesce ad individuare una netta separazione fra i nodi basandosi sulla struttura della rete, il che può essere una conseguenza del fatto che molti aeroporti hanno dei vicini in comune

SIS model

Il modello SIS (Susceptible-Infected-Susceptible) è un modello epidemiologico utilizzato in network science per studiare la diffusione di malattie infettive in una popolazione.

In questo modello, ogni individuo nella popolazione può essere in uno di due stati: suscettibile (S) o infetto (I). Inizialmente, tutti gli individui sono suscettibili alla malattia. Gli individui infetti hanno una certa probabilità di trasmettere la malattia agli individui suscettibili con cui entrano in contatto.

Gli individui suscettibili che vengono infettati passano nello stato infetto e rimangono infetti per un certo periodo di tempo, dopodiché possono guarire e tornare nello stato suscettibile o rimanere infetti e continuare a trasmettere la malattia.

Il modello SIS assume che la probabilità di trasmissione della malattia dipenda dal numero di contatti che un individuo ha con individui infetti nella rete sociale. Inoltre, il modello assume che gli individui infetti possano guarire naturalmente con una certa probabilità, ma possono anche essere reinfectati in seguito.

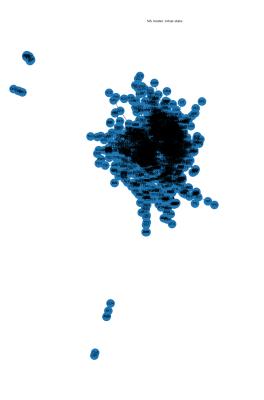
Può avere senso fare un'analisi di questo tipo sulla rete aeroportuale?

Potenzialmente sì, dato che gli aeroporti sono importanti centri di collegamento si può andare a studiare quanto velocemente potrebbe espandersi un contagio attraverso di essi

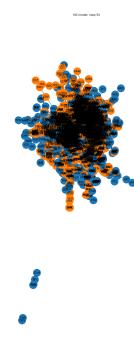
Probabilità di contagio = 0.1

Probabilità di guarigione = 0.1

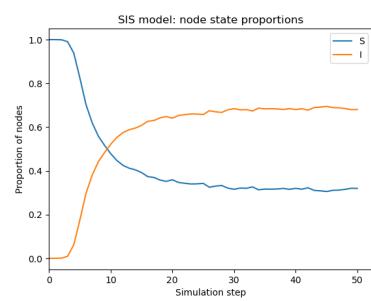
Step iniziale



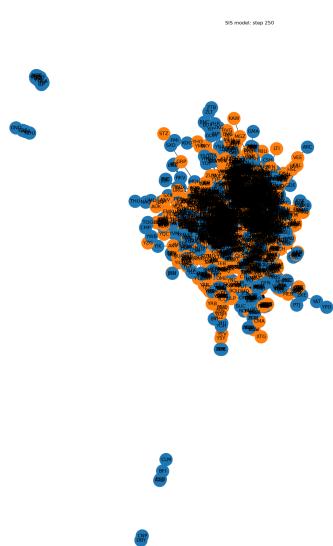
Step 50 della simulazione



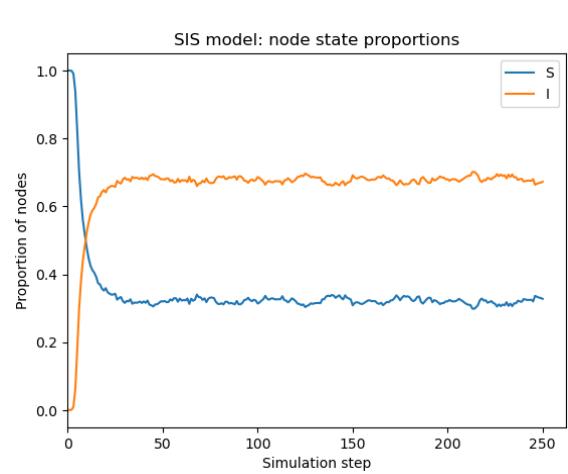
Plot della % di I e S fino al 50esimo step di simulazione



Step 250 della simulazione



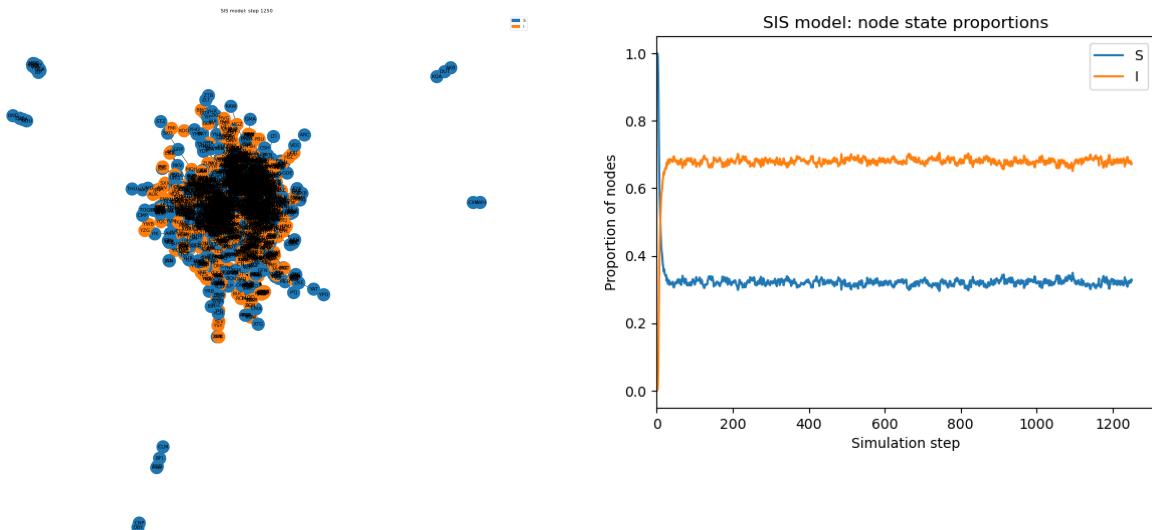
Plot della % di I e S fino al 250esimo step di simulazione



Step 1250 della simulazione



Plot della % di I e S fino al 1250esimo step di simulazione



Nella simulazione del modello SIS sulla rete aeroportuale mondiale, l'evoluzione della diffusione della malattia dipende dal numero di individui suscettibili (che non sono ancora stati infettati) e dal numero di individui infetti.

Inizialmente, il numero di individui suscettibili è molto maggiore rispetto al numero di individui infetti, quindi il numero di individui infetti tenderà ad aumentare rapidamente a causa della loro interazione con gli individui suscettibili.

Tuttavia, man mano che il numero di individui infetti aumenta, il numero di individui suscettibili diminuisce.

Quando il numero di individui suscettibili si avvicina a zero, l'aumento del numero di individui infetti diventa più lento, poiché il numero di individui suscettibili rimanenti è troppo piccolo per sostenere un'alta velocità di diffusione della malattia.

Allo stesso tempo, il numero di individui infetti che guariscono e tornano ad essere suscettibili aumenta, aumentando il numero di individui suscettibili nella rete.

In questo modo, le percentuali di infetti e di suscettibili tendono a bilanciarsi quando il numero di individui suscettibili e infetti si stabilizza in un equilibrio dinamico, ovvero quando il tasso di infezione e di guarigione della malattia si bilancia in modo da mantenere un numero costante di individui suscettibili e infetti.

Questo avviene perchè le probabilità di infezione e guarigione sono uguali

Ma un aeroporto è un'entità complessa:

- se un elemento di tale entità si infetta posso dire che l'aeroporto è infetto
- se lo stesso elemento guarisce non posso dire che l'intero aeroporto è guarito perchè potrebbero esserci stati dei contagi interni \Rightarrow può ever senso diminuire la

probabilità di guarigione

Dimezzando la probabilità di guarigione

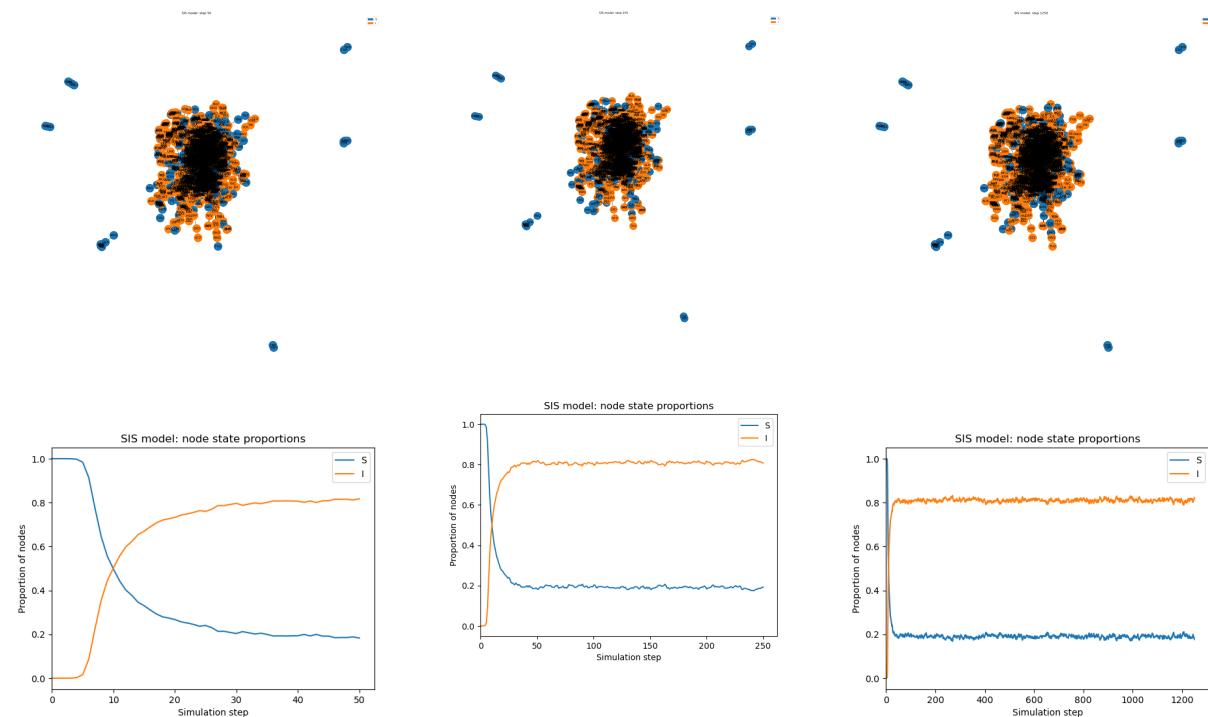
Probabilità di contagio = 0.1 (Beta)

Probabilità di guarigione = 0.05 (Mu)

Passo 50

Passo 250

Passo 1250



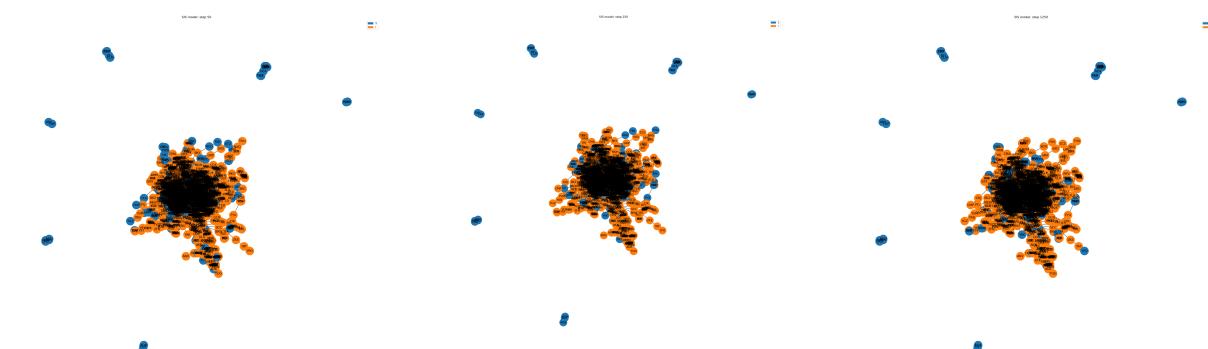
Probabilità di contagio = 0.1

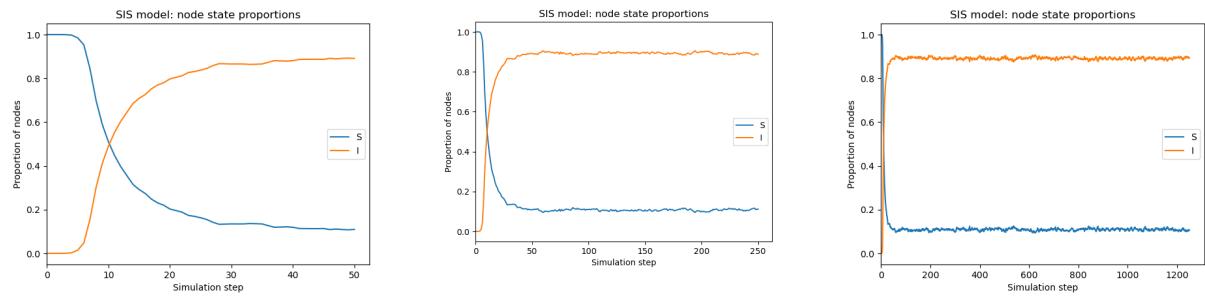
Probabilità di guarigione = 0.025

Passo 50

Passo 250

Passo 1250





Si nota come, al diminuire della probabilità di guarigione, la forbice si apra → i nodi suscettibili diminuiscono drasticamente e quelli infetti aumentano esponenzialmente. Il tutto raggiunge un equilibrio che è molto sbilanciato a favore dei nodi infetti