Università degli studi di Padova
Dipartimento di Scienze Statistiche
Corso di Laurea Magistrale in Scienze Statistiche


# Modelli per dati di rete e regole associative: applicazione a una rete di prodotti Amazon

Relatore Prof.ssa Mariangela Guidolin
Dipartimento di Scienze Statistiche

Alessandro Viel
Matricola N. 2026569

Anno Accademico 2022/2023

# Indice

# Introduzione

I dati di rete sono nella maggior parte dei casi caratterizzati da strutture complesse, e se analizzati con tecniche appropriate, possono fornire informazioni riguardanti le relazioni che costituiscono la loro struttura di base. In questo contesto le metodologie di analisi rivestono sicuramente un ruolo chiave, e nell'arco del tempo si sono sviluppate e adattate a diversi campi di applicazione, partendo dalle relazioni sociali tra gli individui, e arrivando poi ad estendersi in vari ambiti, tra cui quello tecnologico, biologico, dell'informazione e del *marketing*.
In questa tesi è stata proposta un'applicazione della *Social Network Analysis (SNA)* in un contesto strettamente legato al *marketing*, con lo scopo di studiare e comprendere il funzionamento delle raccomandazioni personalizzate utilizzate da Amazon. Nello specifico si è deciso di analizzare un dataset di articoli appartenenti alla stessa categoria principale, ovvero *"Movies &TV"*, cercando si comprendere quali siano i principali fattori che influenzano l'acquisto di due o più articoli.
Per comprendere al meglio le relazioni che legano i prodotti, e scoprire quali possano essere le più interessanti, ovvero quelle che riescono ad apportare più utilità commerciale o efficacia previsiva, si è deciso di applicare, oltre ad un approccio moderno come quello della *Social Network Analysis*, anche una metodologia più tradizionale come quella della *Market Basket Analysis*. A questo scopo è stato utilizzato un dataset contenente le recensioni degli utenti, nel quale è presente l'identificativo di ogni acquirente e il codice univoco di ogni articolo acquistato, mentre per quanto riguarda l'approccio moderno della *SNA*, è stato utilizzato un dataset parallelo, nel quale per ogni articolo è presente una variabile riassuntiva dei prodotti più acquistati congiuntamente.
L'elaborato è stato organizzato in 3 capitoli principali. Nel Capitolo 1 è stato inizialmente approfondito il tema generale della *Market Basket Analysis*, per poi approfondire nel dettaglio le metodologie utilizzate nell'applicazione delle regole associative e dei filtri collaborativi. Nel Capitolo 2 si è invece trattato l'approccio moderno per l'analisi dei dati di rete, partendo dai temi più semplici e definendo gli elementi che la caratterizzano. In seguito si sono specificati i principali indici descrittivi, sia

a livello di nodo che a livello di rete, e si sono proposti alcuni modelli d'analisi, partendo da quelli più semplificativi e arrivando a definire quelli più utili e complessi *(ANOVA*, *Social Relations Model (SRM)*, *Social Relations Regression Model (SRRM)* e *Additive and Multiplicative Effects model (AME))*. Nel Capitolo 3 sono stati illustrati i dati Amazon, e sono state effettuate le principali analisi d'interesse. Inizialmente si sono svolte alcune operazioni di ristrutturazione, in modo tale da pulire i dataset e renderli idonei alle fasi successive. In seguito si sono applicate le regole associative e si è svolta l'analisi per i dati di rete, nella quale inizialmente sono state effettuate le principali analisi descrittive e successivamente si sono applicati i modelli proposti precedentemente. Infine nelle conclusioni viene proposto un confronto tra le due metodologie utilizzate, sottolineando le caratteristiche comuni e le divergenze nei risultati ottenuti.

# Capitolo 1

# Market Basket Analysis

La *Market Basket Analysis* è una tecnica di *data mining* utilizzata da molte aziende, con il principale scopo di far aumentare le proprie vendite e comprendere il comportamento dei consumatori. Si basa sull'analisi di grandi quantità di dati, come la cronologia degli acquisti, al fine di rilevare una regolarità, nonché di individuare i prodotti che potrebbero essere acquistati insieme.
La scoperta di queste regolarità risulta di grande utilità, in quanto potrebbero aiutare i venditori nello sviluppo o nell'ottimizzazione di specifiche strategie di *marketing*, attraverso le quali si riuscirebbe a ridurre i costi e ad aumentare i profitti, ma anche a migliorare la fidelizzazione della clientela, creando una proposta di prodotti e un'esperienza d'acquisto coerente alle aspettative. Ad esempio se in un negozio il prodotto "A" viene spesso acquistato inseme al prodotto "B", ha senso per il venditore mettere i due prodotti vicini, in modo tale da facilitare il comportamento d'acquisto da parte del cliente, o in alcuni casi addirittura indurlo. Ovviamente quello appena esposto è un esempio semplicistico, si possono infatti trovare applicazioni molto più complesse e che riescano a portare maggiori vantaggi, come ad esempio quelle relative alla pubblicità mirata o delle offerte selezionate per uno specifico sottogruppo di clienti.
Due rami della *Market Basket Analysis* sono per l'appunto le regole associative e i filtri collaborativi, nel primo caso l'obiettivo è quello di identificare quali gruppi di prodotti tendono ad essere acquistati insieme, mentre nel secondo caso lo scopo è quello di fornire una raccomandazione personalizzata basata sugli acquisti effettuati da utenti classificati come simili.
Questi due approcci rientrano nei metodi di apprendimento non supervisionato, ovvero tecniche che consistono nel fornire al sistema una serie di *input*, che poi verranno classificati e riorganizzati secondo determinate caratteristiche comuni, in modo tale che il sistema possa effettuare ragionamenti e previsioni sugli *input* suc-

cessivi.

## 1.1 Regole associative

Le regole associative, come anticipato, si collocano tra i metodi di apprendimento non supervisionato, e hanno lo scopo di individuare ricorrenze e regolarità tra i dati (Azzalini e Scarpa, 2009). Nella maggior parte dei casi questo metodo trova applicazione nelle analisi di transazioni commerciali, ovvero liste di prodotti acquistati da uno specifico cliente in un'unica visita. Per riuscire ad immagazzinare al meglio le informazioni riguardanti il cliente e i prodotti acquistati, risulta di fondamentale importanza l'utilizzo delle carte fedeltà, infatti queste carte riescono a dare un grande vantaggio nel raccogliere informazioni riguardanti le transazioni, che in seguito saranno utilizzate per classificare il cliente secondo le finalità d'interesse; con l'introduzione dello *shopping online* questo passaggio è diventato quasi automatico, in quanto, nella maggior parte dei casi, prima di un acquisto l'utente deve crearsi un *account*, che ovviamente fungerà da "carta fedeltà".
Ogni volta che un utente effettua una transazione commerciale vengono registrati alcuni dati, come:

- identificativo dei prodotti acquistati e rispettiva numerosità;
- caratteristiche dei prodotti;
- prezzo di ogni prodotto;
- prezzo complessivo;
- modalità di pagamento;
- identificativo del cliente se in possesso di carta fedeltà o se l'acquisto avviene in rete;
- data e orario della transazione.

In questo ambito risulta fondamentale l'insieme di dati che vengono registrati ed archiviati ogni volta che un utente effettua una transazione, che stanno alla base delle applicazioni tramite regole associative. Attraverso appropriate analisi, questi dati possono essere utilizzati per creare del valore aggiunto grazie ad apposite strategie di *marketing*. Localizzando per esempio gruppi di prodotti che vengono spesso acquistati insieme, si può pianificare una distribuzione diversa degli articoli all'interno dello *store*, oppure indagando sulle abitudini d'acquisto dei consumatori

si potrebbe pensare ad una segmentazione della clientela, al fine di introdurre specifiche promozioni differenziate per gruppi.
Un concetto fondamentale delle regole associative è quello della regolarità d'acquisto, che può essere espresso tramite regole probabilistiche. Per esempio se un cliente acquista il prodotto "A", allora acquisterà anche il prodotto "B" con probabilità $p$. Le regole possono trovare motivazione attraverso fattori esogeni, come mode o azioni dei concorrenti, oppure attraverso fattori endogeni, come sconti e promozioni. Per essere utili però devono soddisfare due principali requisiti, in primo luogo devono essere non banali, e in secondo luogo devono essere coerenti e facilmente comprensibili, in modo tale da riuscire ad estrapolare un'informazione sensata e non scontata.
L'idea di base è comunque quella di contrapporre due *item* o due *itemset* (insieme di prodotti) con una relazione del tipo: se acquisto "A" allora acquisterò anche "B". Per fare ciò però risulterebbe necessario studiare la legge di probabilità di ogni *itemset* rispetto a tutti gli altri, e ciò creerebbe un costo computazionale non indifferente e in molti casi inutile. Per evitare tale costo l'opzione migliore è quella di considerare solamente gli *itemset* più frequenti, e per fare ciò è necessario introdurre il concetto di supporto.

$$supporto = \frac{(numero\,di\,transazioni\,contenenti\,un\,determinato\,itemset)}{(numero\,totale\,di\,transazioni)}$$

Un'altra misura importante nelle regole associative è la fiducia, che può essere espressa come:

$$fiducia = \frac{(transazione\,con\,antecedente\,e\,conseguente)}{(transazione\,con\,antecedente)}$$

Oppure come probabilità condizionata:

$$fiducia = \frac{p(conseguente \bigcap antecedente)}{p(antecedente)} = p(cons|ant)$$

Questa misura è sicuramente di grande interesse nell'analisi delle regole associative, ed esprime la probabilità che un determinato articolo venga acquistato dato l'acquisto di un altro.
È però importante prestare la giusta attenzione nella lettura dei risultati, in quanto un alto livello di fiducia tra due *itemset* potrebbe suggerire erroneamente una regola forte, che potrebbe essere dettata solamente dalla elevata presenza

del conseguente all'interno del dataset. Ad esempio se in un negozio l'articolo "A" ha un supporto molto elevato, e quindi viene registrato in molte transazioni, potrebbero esserci un livello di fiducia molto elevato della regola "se acquisto B allora acquisterò anche $A$", per un generico *itemset* "B". Bisogna quindi prestare la giusta attenzione ai risultati forniti da questa misura, e garantirne un senso logico.

Una possibile soluzione ai problemi descritti precedentemente risiede nell'utilizzo di un indice più affidabile e completo, basato sull'integrazione della precedente misura. Si confronta dunque la fiducia di una regola con un *benchmark*, nel quale assumiamo indipendenza tra antecedente e conseguente. Sotto questa ultima ipotesi si ha:

$$fiducia = p(conseguente)$$

Mettendo a confronto la fiducia di una regola con il suo *benchmark*, si può definire il lift come:

$$lift = \frac{p(conseguente|antecedente)}{p(conseguente)}$$

Questa nuova misura risulta di estrema utilità nell'ambito delle regole associative, in quanto risulta sia attendibile che facilmente interpretabile. Nello specifico una regola utile è definita da un *lift* maggiore di 1, e sta ad indicare che la regola posta in essere ha una capacità predittiva superiore alla semplice conoscenza della probabilità del conseguente.
Mentre la fiducia di una regola serve a determinarne l'utilità commerciale, e dunque a garantirne lo sforzo economico per metterla in atto, il *lift* si concentra maggiormente nel determinarne l'efficacia previsiva rispetto ad una selezione casuale. Sebbene una regola efficiente sia sempre preferibile, è importante che abbia anche un supporto adeguato che ne giustifichi l'attuazione commerciale, risulta dunque di fondamentale importanza trovare un giusto compromesso tra queste misure.
Nella pratica il compito di scovare le migliori regole associative si può suddividere in due parti; la prima parte si concentra nell'individuare tutte le regole da tenere in considerazione, ovvero quelle con un supporto abbastanza elevato, mentre la seconda si occupa di fissare una soglia di fiducia minima per garantirne l'utilità. Per fare ciò vengono utilizzati appositi algoritmi e nel seguito verrà illustrato quello Apriori, che è basato sull'individuazione degli *itemset* più frequenti, ed è fondato su due concetti chiave:

- Se un insieme di oggetti è frequente, allora anche tutti i suoi sottoinsiemi sono frequenti;
- Ogni sovrainsieme di un *itemset* non frequente non può essere frequente.

L'algoritmo Apriori ha la seguente struttura:

1. Si assegnano delle soglie di supporto $s$ e fiducia $p$;
2. Si selezionano gli *itemset* grandi con un solo elemento e che garantiscono un supporto maggiore di $s$;
3. Ad ogni passo si costruiscono gli *itemset* di dimensione $k$, prendendo come rifermento gli *itemset* grandi di dimensione $k-1$, in questo modo si considerano i loro soprainsiemi con un elemento in più (gli *itemset* non grandi di dimensioni $k-1$ non vengono considerati in quanto i loro soprainsiemi non possono essere grandi);
4. Si calcola il supporto degli *itemset* di dimensione $k$ e si considerano solo quelli con supporto adeguato;
5. Per ogni *itemset* si calcolano le regole e si scelgono quelle con fiducia maggiore di $p$.

Attraverso questo processo possono essere estratte numerose regole associative, e quindi risulta necessaria una fase di selezione, che riveste un ruolo di fondamentale importanza in questo ambito. Infine è importante specificare che l'esito delle regole associative non è un modello globale, bensì un insieme di risultati interessanti che possono essere utilizzati in svariati contesti per apportare un guadagno, non necessariamente in ambito economico o di marketing.

## 1.2 Filtri collaborativi

Un secondo concetto fondamentale che riguarda la *Market Basket Analysis* è quello dei filtri collaborativi, che si differenziano dalle regole associative soprattutto per il *focus*, infatti se prima ci si concentrava sul prodotto, ora ci si focalizza maggiormente sul cliente e sulle caratteristiche che lo identificano. Questo processo sta alla base dei sistemi di raccomandazione, ovvero *software* di filtraggio dei contenuti atti alla raccomandazione personalizzata per l'utente, che trovano sempre più spazio nei siti *web* orientati alla vendita *online*, come Amazon, Spotify e Netflix.

Al giorno d'oggi infatti, l'identificazione delle caratteristiche principali che differenziano un cliente dagli altri risulta di assoluta importanza per le aziende, in quanto può permettere di creare un'offerta personalizzata per ogni individuo, e quindi di indurre o suggerire l'acquisto di prodotti selezionati.
L'efficacia dei filtri collaborativi dipende soprattutto da quanto si comprendono gli interessi e le preferenze degli utenti, per questo se si hanno poche informazioni a riguardo di un cliente il meccanismo sarà di difficile attuazione. L'idea di base è che se un utente $x$ condivide dei prodotti acquistati con un altro utente $y$, allora l'utente $x$ sarà interessato agli articoli che non ha acquistato e che sono contemporaneamente apprezzati dall'utente $y$, questo apprezzamento può essere rilevato semplicemente tramite un *click* o un *like*. In conclusione un filtro collaborativo si basa sostanzialmente su due principi:

- Identificare gli utenti simili (vicini);
- Considerare gli *item* che l'utente di interesse non ha acquistato e che risultano i preferiti dei vicini.

Gli approcci più adatti per creare delle misure di prossimità tra gli utenti sono sostanzialmente due, ovvero i metodi basati su *clustering*, oppure se il costo computazionale non risulta eccessivo i KNN (*K-Nearest Neighbors*). Infine è importante specificare che un grosso limite dei filtri collaborativi risiede nel *cold start*, che si può generalmente riassumere nella mancanza di informazioni riguardanti i nuovi utenti. Questo problema è sicuramente di grande rilevanza, in quanto con un numero relativamente basso di informazioni il sistema farà fatica a differenziare l'utente e a identificare quelli simili.
Si è deciso di dedicare una breve sezione a questa metodologia sebbene non verrà poi applicata nella parte di analisi, con lo scopo di illustrare entrambi gli approcci che caratterizzano la *Market Basket Analysis*.

# Capitolo 2

# Analisi per dati di rete

## 2.1 Introduzione

Una rete sociale è un sistema complesso di elementi interconnessi tra loro, ed è costituito da archi che hanno il ruolo di identificare e misurare le relazioni tra gruppi di unità dette nodi. Le reti sociali possono essere applicate in molteplici ambiti, infatti possono misurare qualunque relazione, come ad esempio di prossimità, interazione o vicinanza tra svariate unità, come persone, luoghi o oggetti.
La *Social Network Analysis* (SNA), è una tecnica utilizzata per misurare e rappresentare le relazioni che intercorrono tra individui o gruppi di individui, e la sua nascita si riconduce a innumerevoli studiosi provenienti da ambiti disciplinari differenti. Le applicazione della SNA hanno avuto 3 principali influenze a partire dagli anni '30. La prima da parte dello psichiatra Jacob Levi Moreno, fondatore della sociometria, branca delle scienze sociali che studia la costruzione di gruppi e l'individuazione delle relazioni interpersonali al loro interno. La seconda dallo psicologo Kurt Lewin che diede per primo un approccio matematico, poi integrato da alcuni ricercatori di Harvard, che cambiò la natura prettamente descrittiva della *Social Network Analysis*. La terza influenza deriva invece dagli antropologi di Manchester che studiarono le relazioni nella struttura comunitaria dei villaggi. Successivamente negli anni '50 Dorwin Cartwright e Frank Harary, proseguendo gli studi di Jacob Moreno, ideatore del sociogramma[1], lavorarono alla costruzione della teoria dei grafi, unendo formule matematiche al sociogramma. Nelle prime teorie dei grafi le connessioni iniziavano ad avere valore positivo o negativo in base alla relazione e all'influenza, e a contenere punte di freccia che ne indicavano la direzione, creando così le prime nozioni di asimmetria ed equilibrio nella teoria e analisi delle reti. La

[1]Metodo di osservazione indiretta usato principalmente nelle analisi sociali, col principale scopo di individuare la posizione degli individui nel gruppo

teoria dei grafi era basata su algoritmi molto complessi e difficilmente processabili a causa della loro complessità, e venivano quindi applicati a dataset relativamente ridotti che permettessero di ottenere risultati in archi di tempo ragionevoli. L'interesse e lo sviluppo della SNA riemerse successivamente negli anni '70 con il progresso delle tecniche di analisi tramite computer, che aumentò di conseguenza in numero considerevole la quantità di studi e pubblicazioni, tra i più importanti si ricordano Burt (1982), Freeman et al. (1989), Wasserman e Faust (1994), Scott (2000). Alla fine degli anni '90 vennero poi introdotte due teorie di grande rilevanza per l'analisi delle reti; la prima è quella del mondo piccolo di Duncan Watts e Steven Strogatz (1998), che partendo dalla teoria dei 6 gradi di separazione di Milgram (1967), per cui qualsiasi persona può essere collegata a qualsiasi altra attraverso una catena di non più di 5 intermediari, impone che qualsiasi elemento di una rete possa essere collegato ad un altro con un numero limitato di connessioni; La seconda teoria è quella dell'invarianza di scala di Albert-László Barabási, per cui un nuovo elemento in un una rete tende a collegarsi con quelli con più connessioni. Dagli anni '90 in poi la *Social Network Analysis* ha esteso i suoi orizzonti di ricerca trovando applicazione in svariati campi, come ad esempio in quello biologico, tecnologico e economico.

## 2.2 Definizione di rete

Con il termine rete, come anticipato nella sezione precedente, si vuole fare riferimento ad un sistema composto da due elementi principali, ovvero i nodi, che sono elementi o unità in grado di comunicare fra loro, e le connessioni o archi, il cui scopo è appunto quello di collegare i nodi in base ad alcune relazioni, che possono essere differenti in base alla rete di riferimento e all'oggetto di studio. Al giorno d'oggi il concetto di rete è applicabile ad una vasta gamma di campi (sociologia, matematica, economica, biologia, informatica, ecc.) e per comprendere al meglio la sua definizione si riportano nella Tabella 2.1 alcuni semplici esempi di reti, con i rispettivi nodi e le rispettive connessioni.

| Rete | Nodi | Archi |
|---|---|---|
| Sociale | Persone | Amicizie su Facebook |
| Informazione | Riviste | Citazioni da una rivista all'altra |
| Economica | Stati | Import o export |
| Biologica | Neuroni | Connessioni neurali |
| Aeroportuale | Aeroporti | Voli da un aeroporto all'altro |
| Internet | Pagine web | Link tra le pagine |

Tab. 2.1: Esempi di reti con rispettivi nodi e archi.

Sebbene le reti vengano applicate in una grande vastità di campi, ci sono alcune proprietà e caratteristiche ricorrenti che risultano di estrema utilità per la loro comprensione. Le più importanti sono le seguenti:

- Mondo piccolo (*Small world*): la maggior parte dei nodi ha poche connessioni, ma quasi ogni nodo può essere collegato a qualsiasi altro con un numero limitato di collegamenti;
- Invarianza di scala (*Scale free*): tendenza di un nuovo nodo a connettersi con nodi aventi più connessioni;
- *Hub*: nodi con molte connessioni che fungono da ponti;
- Omofilia: tendenza di un nodo a connettersi con nodi aventi caratteristiche simili ad esso.

In linea generale una rete può essere rappresentata tramite un grafo e una matrice di adiacenza. Un grafo $G = (N, A)$ è costituito da un insieme di nodi $N = \{1, ..., V\}$ e da un insieme di archi $A \subseteq \{\{i, j\} : i, j \in N\}$ definiti come coppie di nodi. Una matrice di adiacenza $Y$ nel caso più semplice, ovvero quello in cui si rileva la presenza o meno di una connessione tra le unità, senza considerarne l'intensità e la direzionalità, è definita come una matrice simmetrica $V \times V$ in cui i nodi vengono disposti in riga e colonna e i valori al suo interno possono assumere valore 0 o 1, rispettivamente $Y_{ij} = Y_{ji} = 1$ se $\{i, j\} \in A$, ovvero se $i$ e $j$ sono connessi, e 0 altrimenti.

$$Y_{ij} = Y_{ji} \begin{cases} = 1 & \text{se c'è relazione tra i e j} \\ = 0 & \text{se non c'è relazione tra i e j} \end{cases}$$

Nel caso più generale la matrice di adiacenza è invece definita come una matrice quadrata $V \times V$ in cui i nodi vengono disposti in riga e colonna, e il generico elemento $Y_{ij}$ rappresenta la relazione che intercorre da $i$ a $j$.

$$Y_{ij} = \begin{cases} \neq 0 & \text{se c'è relazione da i a j} \\ = 0 & \text{se non c'è relazione da i a j} \end{cases}$$

Nello specifico le reti si classificano in 4 principali categorie:

- Rete diretta;
- Rete indiretta;

- Rete binaria;
- Rete pesata.

In Figura 2.1 viene proposto un esempio grafico di rete diretta e indiretta.

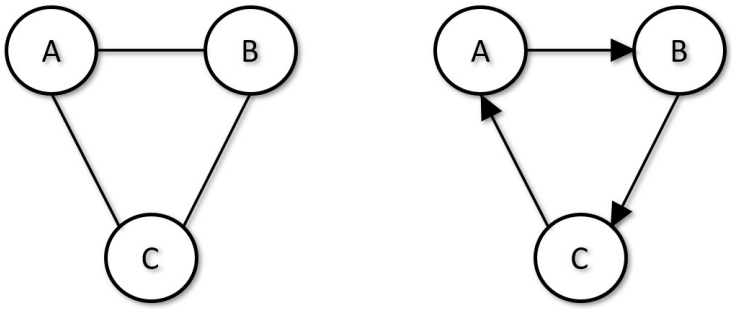


Fig. 2.1: Esempio di rete indiretta (a sinistra) e diretta (a destra).

In riferimento alle matrici di adiacenza specificate precedentemente, si può osservare che il primo caso, ovvero quello più semplice, è definito da una rete indiretta e binaria, in quanto gli elementi della matrice possono assumere solamente 2 valori, che si riconducono al caso di presenza o meno di connessione, e non si osserva direzionalità nelle relazioni. Al contrario il secondo caso risulta più generale e lascia spazio alla possibilità di avere reti dirette e pesate, ovvero con connessioni che misurano l'intensità del legame e presenza di direzionalità nel collegamento. Per completezza di seguito si riporta un esempio di una matrice di adiacenza binaria e di una sociomatrice, ovvero una matrice di adiacenza pesata.

$$Y = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \qquad Y = \begin{pmatrix} 0 & y_{ab} & 0 \\ y_{ba} & 0 & y_{bc} \\ 0 & y_{cb} & 0 \end{pmatrix} \tag{2.1}$$

## 2.3 Indici descrittivi

Una volta organizzati i dati, è possibile iniziare a svolgere alcune analisi in base agli obiettivi prefissati dal ricercatore. Per un primo approfondimento risulta essenziale la visualizzazione degli indici descrittivi, in quanto la sola rappresentazione grafica può non bastare per ricavare le informazioni d'interesse, soprattutto nel caso di una rete con un numero di nodi molto elevato. Gli indici descrittivi risultano quindi molto importanti per comprendere al meglio la struttura e le proprietà della rete

che si vuole analizzare, e possono essere suddivisi in due macro aree, ovvero gli indici a livello di nodo e gli indici a livello di rete.

### 2.3.1 Indici a livello di nodo

La misura più semplice a livello di nodo è il suo grado, e in poche parole definisce il numero di connessioni che esso possiede. Nello specifico dato il nodo $i$, e il numero di nodi nella rete $V$, il grado di un nodo ha la seguente espressione:

$$d_i = \sum_{j=1}^{V} y_{i,j} \tag{2.2}$$

Questa espressione non tiene però conto della direzione dei collegamenti, e risulta quindi più utile nel caso di reti indirette. In presenza di reti dirette risulta invece più opportuno misurare l'*in-degree* e l'*out-degree*, intesi come il numero di collegamenti che arrivano e partono da uno specifico nodo, calcolabili rispettivamente come il numero di valori per riga e colonna diversi da 0 in una matrice di adiacenza.
Altri indici più complessi che forniscono indicazioni sulla posizione del nodo, oltre che al numero di connessioni, sono i seguenti:

- *Betweenness*: indice che misura la media del numero di volte in cui un nodo si trova nel percorso più breve tra altri due nodi. Nella pratica, il livello di *betweenness* di un nodo $i$ è la somma del rapporto tra il numero degli *shortest paths* tra i nodi $u$ e $v$ che passano per $i$, ed il totale degli *shortest paths* tra $u$ e $v$, ovvero $n_{uv}$, con $u$ e $v$ coppie di nodi diversi da $i$. Per *shortest paths* si intende il cammino più breve tra due nodi interconnessi, si misura come il numero di archi di cui si compone e possono essere molteplici.

$$b_i = \sum_{u \neq i \neq v} \frac{n_{uv}(i)}{n_{uv}} \tag{2.3}$$

- *Closeness*: indice che misura la media delle distanze di un nodo da tutti gli altri, e quindi si può dire che fornisce la velocità di propagazione dell'informazione da uno specifico nodo. La *closeness* del nodo $i$ si calcola come rapporto tra $V - 1$, ovvero il totale dei nodi presenti nella rete tranne quello di interesse, e la somma delle distanze tra il nodo $i$ e tutti gli altri nodi.

$$c_i = \frac{V - 1}{\sum_{i \neq j} d(i, j)} \tag{2.4}$$

Infine si può considerare un ultimo indice a livello di coppie di nodi, ovvero la lunghezza minima di *path* di $(i, j)$, $s_{ij}$, che indica il minor numero di archi da attraversare partendo da un nodo $i$ e arrivando ad un nodo $j$.

### 2.3.2 Indici a livello di rete

A livello di rete possiamo trovare i seguenti indici:

- Densità: misura quanto una rete è connessa, e si calcola come il rapporto tra il numero di archi diretti osservati sul totale degli archi diretti possibili, tenendo conto che un nodo non può essere connesso a se stesso:

$$D = \frac{1}{V(V-1)} \sum y_{i,j} \tag{2.5}$$

  L'indice può assumere valori che variano da 0 a 1, può raggiungere l'estremo inferiore se non ci sono connessioni nella rete o l'estremo superiore nel caso di una rete totalmente connessa, in cui ogni nodo è collegato a tutti gli altri.

- Diametro: in una rete connessa è la lunghezza della geodetica più lunga. Si definisce distanza geodetica la lunghezza del percorso più breve che unisce due nodi, quindi il valore assunto dal diametro può dare un'idea della grandezza della rete.

- Lunghezza media di *shortest path*: descrive in media il numero minimo di connessioni per connettere un nodo ad un altro. Si calcola come la media delle lunghezze minime di path:

$$S = \frac{1}{V(V-1)} \sum s_{ij} \tag{2.6}$$

- Transitività: innanzitutto risulta importante importante definire il concetto di triade, ovvero un sottografo formato da tre nodi connessi tra loro; considerando 3 nodi $a$, $b$ e $c$ il concetto su cui è fondato questo indice è che se $a$ conosce $b$, e $b$ conosce $c$ allora è probabile che $a$ conosca anche $c$. Si calcola come:

$$T = \frac{3\tau_{\triangle}(G)}{\tau_3(G)} \tag{2.7}$$

  In cui $\tau_{\triangle}(G)$ indica il numero totale di triple chiuse, ovvero con tre connessioni, e $\tau_3(G)$ indica il numero complessivo di triple aperte e chiuse (con due o tre connessioni). L'indice varia da 0 a 1, e ad un elevato valore corrisponde un'alta propensione a formare triple chiuse.

- Reciprocità: in una rete diretta rappresenta la tendenza dei nodi nel formare archi reciproci. Si calcola come il rapporto tra il numero degli archi reciproci e il totale degli archi presenti:

$$R = \frac{\sum_{i \neq j} (YY')_{ij}}{\sum_{i \neq j} Y_{ij}} \tag{2.8}$$

- Modularità: rappresenta una misura di omofilia, ovvero la tendenza dei nuovi nodi a connettersi con nodi che possiedono caratteristiche simili. Per calcolarla definiamo le seguenti due misure:

$$e_{ij} = \frac{\text{numero di archi che hanno inizio nel gruppo } i \text{ e fine nel gruppo } j}{\text{numero totale di archi nella rete}}$$
$$a_i = \frac{\text{numero di archi che hanno inizio (o fine) nel gruppo } i}{\text{numero totale di archi nella rete}}$$

  La modularità è la frazione di archi che connette nodi dello stesso tipo meno il valore atteso della stessa quantità in una rete con connessioni casuali:

$$Q = \sum_{K=1}^{K} e_{kk} - \sum_{K=1}^{K} a_k^2 \tag{2.9}$$

  Nella formula $K$ rappresenta il numero di gruppi ipotizzati a priori basandosi sul grafo.

- Assortatività: modularità normalizzata:

$$A = \frac{\sum_{K=1}^{K} e_{kk} - \sum_{K=1}^{K} a_k^2}{1 - \sum_{K=1}^{K} a_k^2} \tag{2.10}$$

  L'indice varia tra -1 e 1, un valore molto vicino all'estremo superiore indica una forte coesione tra i gruppi, ovvero che i nodi nella rete sono collegati con nodi simili; empiricamente un valore superiore a 0.3 indica che i gruppi non sono banali.

## 2.4 Modelli per dati di rete

L'analisi delle statistiche descrittive è sicuramente un processo essenziale e necessario per comprendere al meglio la rete di interesse, ma per un'analisi più approfondita risulta utile applicare alcuni modelli per dati di rete. Per dati di rete si intendono i

dati che forniscono informazioni sulla relazione tra nodi, e generalmente si tratta di relazioni tra coppie di nodi. A questo punto è importante introdurre il concetto di diade, ovvero coppia di nodi, e di variabile diadica, ovvero una quantità misurata o osservata per molte diadi. Per misurare una variabile diadica viene utilizzata una matrice di adiacenza pesata (sociomatrice), in cui $y_{ij}$ misura la relazione tra i nodi $i$ e $j$ nella direzione da $i$ a $j$; Si tenga infatti presente che $y_{i,j} \neq y_{j,i}$, perché sebbene i nodi presi in causa siano i medesimi, cambia la direzionalità della relazione, che nel primo caso implica $i$ come mittente e $j$ come ricevente, e nel secondo caso il contrario.
Allo scopo di valutare gli effetti delle altre variabili sulle relazioni diadiche, oppure più in generale per individuare specifici *patterns*, risulta fondamentale l'analisi della matrice di adiacenza. Per svolgere queste analisi è necessario l'adattamento di modelli statistici basati sulla struttura della sociomatrice, e a tal fine di seguito ne vengono proposti alcuni, partendo dal più semplice e aumentando via via la complessità.

### 2.4.1 Modello ANOVA

Innanzitutto, prima di spiegare il funzionamento del modello, è necessario definire 3 misure fondamentali, ovvero la media generale, la media di riga e la media di colonna:

$$\bar{y}_{..} = \frac{y_{..}}{n(n-1)} = \frac{1}{n(n-1)} \sum_{i \neq j} y_{i,j} \tag{2.11}$$

$$\bar{y}_{i.} = \frac{y_{i.}}{n(n-1)} = \frac{1}{n-1} \sum_{j:j \neq i} y_{i,j} \tag{2.12}$$

$$\bar{y}_{.j} = \frac{y_{.j}}{n(n-1)} = \frac{1}{n-1} \sum_{i:i \neq j} y_{i,j} \tag{2.13}$$

Nello specifico, l'equazione 2.11 rappresenta la media generale della matrice, e viene definita come il rapporto tra la somma delle osservazioni, e il numero di nodi moltiplicati per lo stesso numero meno uno, in quanto evitando la presenza di nodi riflessivi, ovvero nodi collegati con se stessi, la diagonale principale della sociomatrice presenta valori sempre pari a 0.
L'equazione 2.12 rappresenta una media condizionata di riga, infatti mantenendo fissa una specifica riga $i$, e dunque condizionandosi ad essa, si procede variando l'indice di colonna $j$. In questo modo si ottiene una media per riga, ovvero una media per uno specifico nodo, che riesca a tenere in considerazione tutti gli altri nodi presenti sulla rete.

Infine l'equazione 2.13 rappresenta una media condizionata di colonna, e dunque riconducendosi alla logica enunciata precedentemente, si mantiene fissa la colonna $j$ e si procede variando l'indice di riga $i$. In questo modo si ottiene una media per colonna che riesca a tenere in considerazione tutti i nodi di riga, ad eccezione dello stesso nodo corrispondente a quello di colonna, in quanto nella sommatoria viene imposto $i \neq j$.
Questi ultimi due indici risultano molto utili e forniscono delle informazioni interessanti sulla rete, in primo luogo, la media condizionata di riga fornisce una misura di "socialità" di un certo nodo, infatti valuta la propensione nel stabilire connessioni con altri nodi, basandosi sulla media dei collegamenti in uscita; in secondo luogo, la media condizionata di colonna fornisce una misura di "popolarità" di uno specifico nodo, che al contrario di prima esprime la propensione del nodo nel ricevere collegamenti da altri, ed è quindi basata sul numero di collegamenti in entrata.
Spesso in una sociomatrice si hanno valori di una certa riga correlati tra loro, in quanto caratterizzati dallo stesso "mittente" $i$, infatti se il nodo $i_1$ risulta più socievole del nodo $i_2$ ci si aspetta relativamente dei valori più elevati nella riga riferita al nodo $i_1$, e questo comporta eterogeneità tra le medie di riga, nonché eterogeneità nella socialità. Discorso analogo si può fare con la correlazione tra i valori di una stessa colonna, in quanto caratterizzati dallo stesso "ricevente" $j$; infatti se il nodo $j_1$ risulta più popolare del nodo $j_2$ ci sia aspetta relativamente dei valori più elevati nella colonna riferita al nodo $j_1$, e questo comporta eterogeneità tra le medie di colonna, nonché eterogeneità nella popolarità.
Il fatto che un nodo possa assumere valori di "popolarità" o "socialità" più elevati, porta inevitabilmente ad un eterogeneità tra le medie di riga e le medie di colonna, e per tenere in considerazione questo aspetto, un modello classico che viene utilizzato in questo ambito è il modello ANOVA.
L'ANOVA è un modello basato sulla scomposizione della varianza, ed ipotizza appunto che la variabilità di $y_{i,j}$, attorno alla media generale $\mu$, sia data da effetti additivi di riga e colonna, $a_i$ e $b_j$.

$$y_{i,j} = \mu + a_i + b_j + \epsilon_{i,j} \tag{2.14}$$

Tramite l'equazione 2.14 si riesce a tenere in considerazione sia il comportamento medio della sociomatrice, racchiuso in $\mu$, sia gli effetti di riga e di colonna, tramite $a_i$ e $b_j$, che riescono a cogliere l'eterogeneità nella matrice.
Questo modello presenta però alcuni limiti, infatti non tiene in considerazione un aspetto molto importante, ovvero che ogni nodo è allo stesso tempo mittente e ricevente, e che dunque è coinvolto in due effetti additivi, di riga $a_i$ e di colonna $b_j$.

Questo ovviamente comporta la presenza di ogni nodo in una coppia di effetti $(a_i, b_i)$, e quindi è lecito aspettarsi una correlazione tra i vettori $(a_1, ..., a_n)$ e $(b_1, ..., b_n)$; questa relazione aiuta inoltre a capire se un nodo "popolare" è anche "sociale". In aggiunta per ogni coppia di nodi $i, j$, nella matrice di adiacenza $Y$ sono presenti due osservazioni, ovvero $y_{i,j}$ e $y_{j,i}$, ed è dunque ragionevole aspettarsi una correlazione tra le due.

### 2.4.2 Social Relations Model (SRM)

Risulta dunque necessaria l'introduzione di un metodo che non tenga in considerazione solo gli effetti di riga e di colonna, ma anche le correlazioni citate precedentemente. Questo metodo è stato introdotto dai sociologi Warner, Kenny e Stoto (1979) con il termine *Social Relations Model* (SRM). Il modello proposto presenta una versione più completa rispetto al modello ANOVA esplicitato precedentemente, e necessita l'aggiunta di alcune importanti assunzioni:

$$\begin{aligned} y_{i,j} &= \mu + a_i + b_j + \epsilon_{i,j} \\ \{(a_1, b_1), ..., (a_n, b_n)\} &\sim N(0, \Sigma_{ab}) \quad i.i.d. \\ \{(\epsilon_{i,j}, \epsilon_{j,i})\} &\sim N(0, \Sigma_\epsilon) \quad i.i.d. \end{aligned} \tag{2.15}$$

dove

$$\Sigma_{ab} = \begin{pmatrix} \sigma_a^2 & \sigma_{ab} \\ \sigma_{ba} & \sigma_b^2 \end{pmatrix} \quad \text{e} \quad \Sigma_\epsilon = \sigma_\epsilon^2 \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

Da cui ne deriva che:

- $\mu + a_i$ è la media della riga $i$ condizionatamente agli effetti di riga;
- $\sigma_a^2$ descrive l'eterogeneità delle medie di riga;
- $\sigma_b^2$ descrive l'eterogeneità delle medie di colonna;
- $\sigma_{ab}$ descrive la correlazione tra medie di riga e di colonna;
- $\sigma_\epsilon^2$ descrive una variabilità addizionale, e la correlazione diadica è catturata da $\rho$ (oltre a quella descritta da $\sigma_{ab}$).

Tramite questo modello si può dunque tenere conto della dipendenza diadica, ovvero la dipendenza tra una coppia di nodi. Il *Social Relations Model* riesce dunque a superare i limiti del modello ANOVA, riuscendo a tenere in considerazione due differenti correlazioni, in primo luogo quella tra le osservazioni provenienti dalla

stessa coppia di nodi, e in secondo luogo quella degli effetti di riga e di colonna provenienti dallo stesso nodo.

### 2.4.3 Social Relations Regression Model (SRRM)

Il *Social Relations Regression Model* è utile nel caso in cui si voglia misurare la relazione tra una variabile risposta diadica e altre variabile diadiche o di nodo. La sua forma si ottiene combinando un modello di regressione lineare con un *Social Relations Model*, ottenendo la seguente formula:

$$y_{i,j} = \beta^T x_{i,j} + a_i + b_j + \epsilon_{i,j} \tag{2.16}$$

dove $x_{i,j}$ è un vettore di dimensione $p$ di regressori, mentre $B$ rappresenta il vettore di coefficienti di regressione. Il vettore $x_{i,j}$ può quindi contenere variabili di nodo o diadiche, per esempio $x_{i,j} = (x_{r,i}, x_{c,j}, x_{d,i,j})$, dove $x_{r,i}$ è il vettore delle caratteristiche del nodo $i$ come "mittente", $x_{c,j}$ è il vettore delle caratteristiche del nodo $j$ come "ricevente", e $x_{d,i,j}$ è il vettore della coppia ordinata di nodi $(i, j)$.

### 2.4.4 Additive and Multiplicative Effects model (AME)

Le reti possono spesso presentare delle strutture di dipendenza tra triplette di nodi, quantificabili tramite l'indice di transitività. Tale dipendenza triadica può essere sintetizzata tramite una misura globale di transitività: $\sum_{i,j,k} \hat{\epsilon}_{i,j}\hat{\epsilon}_{j,k}\hat{\epsilon}_{i,k}$, dove gli $\hat{\epsilon}$ sono i residui del modello stimato tramite i minimi quadrati. Il modello SRRM, come qualsiasi altro modello ad effetti casuali gaussiani, non è in grado di cogliere la dipendenza triadica in quanto tutti i momenti del terzo ordine di variabili casuali gaussiane con media zero, combinate in modo additivo, sono uguali a zero. Una soluzione è data dal modello AME (*Additive and Multiplicative Effects model*), ovvero quella di introdurre ulteriori effetti casuali non combinati in modo additivo:

$$\begin{aligned} y_{i,j} &= \beta^T x_{i,j} + a_i + b_j + u_i^T v_j + \epsilon_{i,j} \\ \{(u_1, v_1), ..., (u_n, v_n)\} &\sim N_{2r}(0, \Psi) \quad i.i.d. \\ \{(a_1, b_1), ..., (a_n, b_n)\} &\sim N_2(0, \Sigma) \quad i.i.d. \\ \{(\epsilon_{i,j}, \epsilon_{j,i}) : i < j\} &\sim N_2(0, \sigma^2 \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}) \quad i.i.d. \end{aligned} \tag{2.17}$$

dove $u_i$ e $v_j$ sono i vettori r-dimensionali di fattori latenti, non osservati, riferiti al comportamento di $i$ come "mittente" e rispettivamente di $j$ come "ricevente". L'effetto moltiplicativo $\gamma_{i,j} = u_i^T v_j$ è in grado di quantificare la dipendenza triadica,

e può può descrivere la possibilità di "variabili omesse".

$$E[\gamma_{i,j}\gamma_{j,k}\gamma_{k,i}] = E[u_i^T v_j u_j^T v_k u_k^T v_i] = E[u_i^T v_j]^3 = tr(\Psi_{uv})^3 \tag{2.18}$$

Il modello esplicitato nell'equazione 2.17 viene denominato AME Normale, in quanto condizionatamente a $\beta$ e agli effetti moltiplicativi, le osservazioni si distribuiscono come una normale. Risulta comunque possibile generalizzare il modello per un approccio tramite una variabile risposta ordinale o binaria, che verrà approfondito nella sezione successiva.
Il modello AME può anche essere interpretato come un modello a variabili latenti, in quanto $u_i$ e $v_i$ descrivono delle caratteristiche non osservate di $i$, in qualità di "mittente" nel primo caso e "ricevente" nel secondo. Ulteriori importanti modelli non-additivi per variabili latenti sono i seguenti:

- Modello a blocchi stocastici: è definito come modello a partizione casuale, e permette di rappresentare l'eterogeneità non osservata dei singoli nodi. Il concetto di base è l'appartenenza di ogni nodo ad una classe latente, chiamata blocco, e le relazione tra nodi vengono determinate stocasticamente dall'appartenenza alla loro classe. Il modello si basa sul concetto di equivalenza stocastica, secondo cui i nodi possono essere divisi in gruppi in modo tale che quelli appartenenti allo stesso gruppo abbiano relazioni simili.

- Modello a spazi latenti: si basa su uno spazio latente non osservato, nel quale ogni singolo nodo ha una precisa posizione basata sulla relazione con gli altri nodi, quindi se la forza della relazione tra una coppia di nodi aumenta, di conseguenza la loro distanza nello spazio latente diminuisce. Il modello fornisce dunque una rappresentazione dell'esistenza di gruppi di nodi legati da relazioni forti, nonché basati su proprietà come transitività e assortatività.

I modelli a variabili latenti con effetti moltiplicativi, come il modello AME, possono comunque fornire una generalizzazione del modello a blocchi stocastici e del modello a spazi latenti.

### 2.4.5 Generalizzazione del modello AME per dati binari

In molti frangenti alcune variabili diadiche non sono rappresentate in maniera ottimale da un modello con errori gaussiani. Ad esempio nel caso di presenza di variabili binari o ordinarie, non è possibile effettuare una trasformazione della variabile diadica in modo tale che il modello AME sia ragionevole. Tuttavia è possibile definire alcune estensioni del modello ad effetti additivi e moltiplicativi, in modo tale da

poterlo generalizzare per variabili risposta di tipo binario o ordinale. Nello specifico queste estensioni sono incentrate sulla rappresentazione a variabili latenti di modelli probit o probit per dati ordinali.
Sia $S$ la matrice di adiacenza osservata per una variabile diadica $s_{i,j}$. La rappresentazione più semplicistica di una variabile ordinale è data da una variabile binaria, la quale assume unicamente due valori distinti, che stanno ad indicare la presenza o meno di una relazione tra $i$ e $j$, quindi se $s_{i,j} = 0$ o $s_{i,j} = 1$ dipende unicamente dalla presenza di un arco che connette i due nodi. Il metodo proposto consiste nell'utilizzo di un modello di regressione probit, in questo modello la probabilità che esista una connessione tra $i$ e $j$ è data da $\Phi(\beta^T x_{i,j})$, dove $\Phi$ è la funzione di ripartizione di una normale standard. In questo contesto si utilizza il modello probit tramite una sua rappresentazione a variabile latente, nella quale la variabile diadica $s_{i,j}$ assume valore uno se la variabile latente $y_{i,j} \sim N(\beta^T x_{i,j}, 1)$ è maggiore di zero (Albert and Chib (1993)). Generalmente nella regressione probit, l'assunzione di indipendenza tra le osservazioni $y_{i,j}$ non risulta appropriata per i dati di rete, però, nel caso di un modello di regressione per dati binari, che tenta di catturare i vari tipi di dipendenza spiegati precedentemente, come quella di riga, di colonna, o diadica, si può utilizzare un modello AME per descrivere la variabile latente $y_{i,j}$:

$$\begin{aligned} y_{i,j} &= \beta^T x_{i,j} + u_i^T v_j + a_i + b_j + \epsilon_{i,j} \\ s_{i,j} &= g(y_{i,j}), \end{aligned} \tag{2.19}$$

in cui $a_i$, $b_i$ e $\epsilon_{i,j}$, seguono la struttura della covarianza del SRM, e $g(y)$ è l'indicatore binario per $y > 0$.

# Capitolo 3

# Analisi di una rete di prodotti acquistati su Amazon

## 3.1 I dati

I dati analizzati si riferiscono ad una rete di prodotti acquistati su Amazon, e sono contenuti in due differenti dataset; il primo al cui interno sono presenti i metadati, nei quali si trovano le caratteristiche descrittive di ciascun articolo, e il secondo al cui interno si osservano le recensioni degli utenti riferite ai corrispettivi prodotti. Il dataset è stato curato da Janmo Ni, *Ph.D. in Computer Science Department, University of California San Diego*, ed è disponibile gratuitamente al seguente link: http://deepyeti.ucsd.edu/jianmo/amazon/index.html.
I dati presentano un arco temporale abbastanza elevato, infatti si possono trovare recensioni a partire dal 1997 fino ad arrivare al 2018. Nella loro interezza oltre che molto estesi temporalmente, hanno anche dimensioni molto elevate, infatti presentano un peso totale di quasi 50 *gigabyte*; a questo proposito per renderne più semplice la consultazione, sono disponibili nel sito già suddivisi per categorie di prodotto, oltre che per metadati (15 milioni) e relative recensioni (233 milioni).
Il dataset completo risulta dunque troppo esteso, e si è quindi optato per l'utilizzo di in una determinata categoria di prodotti, ovvero *"Movies and TV"*. Questa categoria è stata scelta principalmente perché garantisce un buon compromesso tra dimensione dei dati e informazione contenuta, ed entrando nello specifico racchiude circa 10 milioni di recensioni con 12 variabili nel primo dataset, e 200 mila metadati con 19 variabili nel secondo. Prima di visualizzare i dati ed effettuare le prime operazioni al dataset, si è ritenuto necessario un cambio di formato, in quanto i due *file* di interesse risultavano scritti con estensione *.json* riga per riga, la quale

ne limitava fortemente la velocità di lettura tramite *software R*. Si è quindi optato per una trasformazione dei *file* in formato *.csv* attraverso *Python*, che ne ha reso più comoda e veloce l'apertura e la manipolazione, per poi continuare il resto delle analisi tramite *R*.
Inizialmente, prima di effettuare le varie analisi, si sono ritenute necessarie alcune ristrutturazioni nei dataset. In primo luogo si è deciso di osservare i valori mancanti, con la conseguente eliminazione delle variabili che ne presentavano molteplici, ovvero almeno l'80%, con lo scopo di snellire i due dataset per facilitarne la manipolazione. Rispettivamente sono state eliminate dalle recensioni *vote* e *image*, e dai metadati *fit*, *tech1*, *tech2*, *feature*, *similar_item*, *date*, *imageURL* e *imageURLHighRes*.
Successivamente a questa operazione, nel dataset riferito alle recensioni sono presenti le variabili :

- *Overall*: variabile quantitativa discreta che indica il voto complessivo della recensione, da 1 a 5;
- *Verified*: variabile dicotomica, che assume valore 1 se la recensione è verifica e 0 altrimenti (per verificato si intende che l'utente che ha scritto la recensione ha acquistato o utilizzato il prodotto su Amazon e non ha ricevuto il prodotto con uno sconto particolare);
- *ReviewTime*: data di pubblicazione della recensione;
- *ReviewerID*: codice univoco alfanumerico per l'identificazione di ogni utente;
- Asin: "Amazon standard identification number", codice univoco alfanumerico per l'identificazione di ogni prodotto su Amazon;
- *Style*: variabile qualitativa nominale, che specifica il formato in cui è stato venduto l'articolo;
- *ReviewerName*: nome dell'utente che ha scritto la recensione;
- *ReviewText*: testo della recensione;
- *Summary*: breve riassunto della recensione;
- *UnixReviewTime*: codice numerico per l'identificazione della sequenza temporale delle recensioni.

Nel dataset riferito ai metadati sono presenti le variabili:

- *Category*: categorie e sottocategorie Amazon in cui si può trovare il relativo prodotto;

- *Description*: breve descrizione dell'articolo;
- *Title*: nome dell'articolo;
- Asin: "Amazon standard identification number", codice univoco alfanumerico per l'identificazione di ogni prodotto su Amazon;
- *Also_buy*: Asin dei prodotti acquistati insieme al prodotto;
- *Brand*: marchio di un prodotto o di una linea di prodotti;
- *Rank*: posizione nella classifica dei prodotti più venduti per categoria principale;
- *Also_view*: Asin dei prodotti visti insieme al prodotto acquistato;
- *Main_cat*: categoria principale del prodotto;
- *Price*: prezzo del prodotto;
- *Details*: ulteriori dettagli riguardanti il prodotto.

### 3.1.1 Ristrutturazione del dataset

I due dataset così riportati comprendono alcune variabili poco utili e altre che necessitano un rimodellamento, di conseguenza si sono rese opportune alcune operazioni di ristrutturazione. Lo scopo principale è stato quello di pulire i dati in modo da renderli in primo luogo più informativi, e in secondo luogo più utili alle successive analisi. Si sono effettuate alcune operazioni come la rimozione dei duplicati, l'eliminazione di variabili ritenute poco interessanti, e la creazione e modellazione di altre per aggiungere informazione.

#### Dataset delle recensioni

Per quanto riguarda il dataset riferito alle recensioni, si è deciso inizialmente di eliminare tutte le osservazioni contenti Asin non presenti nel dataset dei metadati, in quanto per le successive analisi risultava necessario ricondursi per ogni recensione alle caratteristiche del prodotto. In riferimento alle variabili si è deciso di eliminare la variabile *reviewerName* perché apportava la stessa informazione contenuta in *reviewerID*, e le variabili *reviewText* e *summary* in quanto in parte già espresse dalla variabile *overall*, e così come riportate risultavano di poca utilità perché troppo dispersive e difficilmente riassumibili in categorie. Questa scelta è stata effettuata in maniera consapevole del fatto che attraverso varie operazioni di *text mining* si

sarebbero potute estrarre alcune informazioni. È stata aggiornata la variabile categoriale *style* creando due fattori principali che corrispondo ai formati più frequenti nel dataset (l'80%), ovvero Amazon Video e DVD, agglomerando in Amazon Video tutti quegli articoli con supporto *streaming*, come *Prime Video* e *Kindle*, e in DVD i restanti formati fisici come *VHS* e *Blu-ray*, in modo da creare due categorie ben distinte e basate sul tipo di prodotto venduto. Sono state poi create alcune variabili aggiuntive per ogni prodotto, ovvero *mean_overall*, variabile quantitativa continua che esprime la media delle recensioni per articolo, e *num_rev*, variabile quantitativa discreta che esprime il numero di recensioni presenti nel dataset per prodotto. Infine queste ultime tre variabili sono state collegate tramite un operazione di *merge* all'altro dataset, in modo da rendere più completi i metadati. Prima di questa operazione è stato però necessario cambiare le categorie della variabile *style*, in quanto uno stesso articolo con più recensioni poteva avere formati differenti. È stata quindi aggiunta un'ulteriore categoria "Amazon Video - DVD" che riuscisse ad esplicitare l'abbinamento tra quelle create in precedenza, giungendo dunque ad ottenere 3 diverse categorie.
Di seguito, nella Figura 3.1 , è riportato un esempio di prodotto presente nel *dataset*, con lo scopo di comprendere meglio la struttura dei dati.

```
asin          :  B0021AENJG
reviewerID    :  ALCNZ9FR0LZHH
overall       :  5
verified      :  False
style         :  Amazon Video
reviewTime    :  2013-12-30
unixReviewTime:  1388361600
```

Fig. 3.1: Struttura del dataset riferito alle recensioni.

### Dataset dei metadati

Per quanto riguarda il dataset riferito ai metadati, in primo luogo si è deciso di eliminare alcune variabili, tra cui la variabile *main_cat* perché assume lo stesso valore per tutte le osservazioni (ovvero *"Movies* & TV"), le variabili *description* e *details* in quanto come nel caso precedente servirebbero delle operazioni di *text mining* per provare ad estrarre informazioni utili alle analisi, e la variabile *brand* che assume troppe modalità e non risulta accorpabile.
Successivamente si è ritenuto necessario rielaborare la variabile *category*, che per ogni articolo si presentava con una struttura simile a quella di un *link*, ovvero con una successione di categorie, da quelle più agglomerative a quelle più divisive. Con

lo scopo di estrarre più informazione possibile sono state quindi create tre differenti variabili: la prima *num_cat* quantitativa discreta per indicare il numero di categorie in cui si può trovare il prodotto, e la seconda e la terza fattoriali, ovvero *max_category* e *min_category*, per indicare le categorie di appartenenza, rispettivamente quella più agglomerativa ovvero la categoria successiva a quella generale che comprende tutti i prodotti, e una più divisa in cui viene riportata l'ultima categoria in cui si può trovare il prodotto, che nella maggior parte dei casi coincide con il tipo o genere del film. Infine si sono modellate le variabili *price* e *rank*, in modo da renderle rispettivamente quantitativa continua e quantitativa discreta per consentirne l'utilizzo nelle analisi. Di seguito, nella Figura 3.2, è riportato un esempio di prodotto presente nel dataset, con lo scopo di comprendere al meglio la struttura dei dati.

```
asin:  B008KZY05S
title:  Lady Gaga: On the Edge
also_buy:  B01M5GB0U7 B0076YFJQS B0045JIM5W B005SV9X22 B0042AH2IW ...
rank:  130177
also_view:  B005SV9X22 B00P4IN10O B01M5GB0U7 B0045JIM5W B0076YFJQS ...
price:  12.5
num_cat:  3
max_category:  Independently Distributed
min_category:  Documentary
style:  DVD
mean_overall:  4
num_review:  12
```

Fig. 3.2: Struttura del dataset riferito ai metadati.

### 3.1.2 Statistiche descrittive

Una volta ristrutturati i dataset si è svolta una prima analisi descrittiva, in modo tale da evidenziare le prime caratteristiche dei dati, e capire meglio la distribuzione delle variabili. Innanzitutto il dataset riferito alle recensioni presenta 8.516.806 osservazioni e 7 variabili, mentre quello riferito ai metadati comprende 181.828 osservazioni e 12 variabili. Le recensioni risultano per il 77% verificate, e come detto precedentemente indica che chi ha scritto la recensione ha acquistato o utilizzato il prodotto su Amazon e non ha ricevuto il prodotto con uno sconto particolare. La distribuzione del voto complessivo è invece visualizzabile nel diagramma a barre in Figura 3.3, nel quale possiamo osservare come la grande maggioranza degli utenti che scrivono una recensione esprimano una votazione piuttosto elevata, con una grande maggioranza che vota 5 stelle, ovvero il 63%, e circa l'80% che vota almeno 4 stelle.

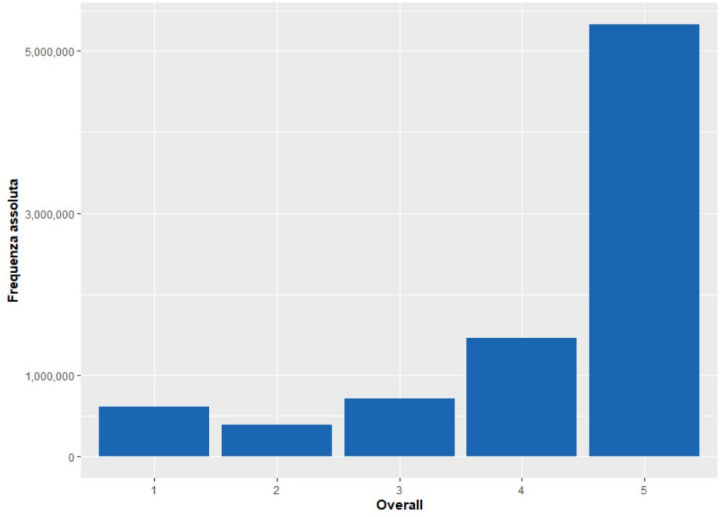


Fig. 3.3: Diagramma a barre sulla valutazione complessiva delle recensioni.

Questa forte asimmetria nella distribuzione del voto complessivo può portare a pensare che la grande maggioranza degli utenti risulti effettivamente molto soddisfatta degli articoli acquistati. Bisogna però tenere in considerazione alcuni aspetti molto importanti, come il fatto che gli utenti non ricevano alcun vantaggio nel lasciare una valutazione e non siano obbligati in alcun modo a farla, bensì sta a loro discrezione decidere se scrivere o meno una recensione, rendendo il processo di selezione non del tutto casuale. Ulteriori fattori distorsivi di natura psicologica si possono invece trovare nella *social desirability*, ovvero la propensione a dare risposte non veritiere e socialmente più accettabili.

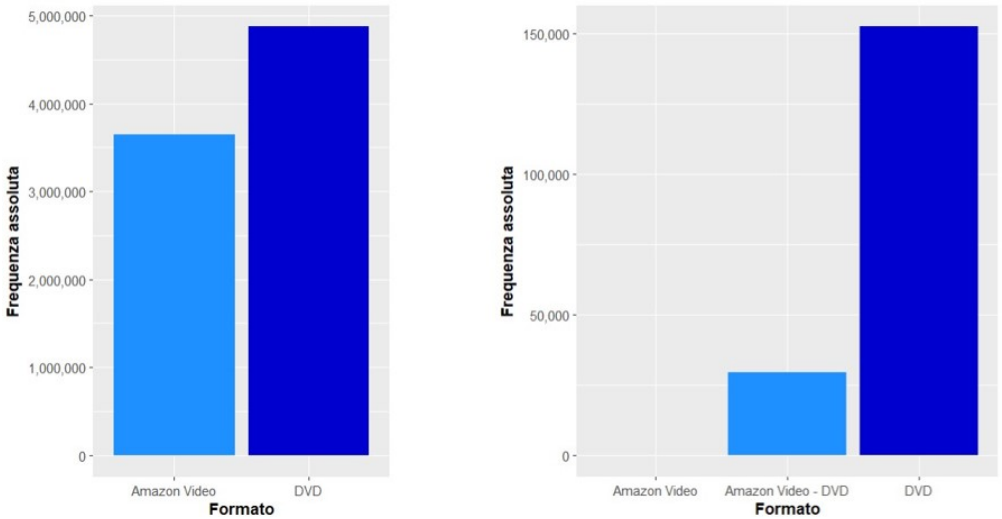


Fig. 3.4: Diagramma a barre per il tipo di formato (a sinistra per le recensioni e a destra per i metadati).

Per quanto riguarda la variabile *style*, che si ricorda indicare il tipo di formato in cui è stato venduto l'articolo, sono riportati nella Figura 3.4 due grafici a barre riferiti al dataset delle recensioni (a sinistra) e al dataset dei metadati (a destra), con la diversa categorizzazione data. Dai due grafici si può notare come le recensioni abbiano una ripartizione abbastanza equa per formati fisici e *streaming*, con una maggiore presenza dei primi che compongono il 57% delle osservazioni. Un altro dato che salta subito all'occhio è l'importante presenza nel dataset riferito ai metadati di prodotti unicamente in formato fisico, ovvero l'84%, e della quasi assenza di articoli completamente in formato *streaming*, 16 su 181.828. Dai grafici a barre si possono però osservare due particolarità molto interessanti, la prima si può riassumere nella presenza quasi totale di recensioni di prodotti in formato fisico per ogni recensione degli stessi prodotti in formato *streaming*, mentre la seconda sta nella maggiore presenza di recensioni per articoli in formato *streaming* rispetto a quelli in formato fisico, che può portare a pensare che gli utenti siano più favorevoli a recensire un prodotto acquistato nel primo formato indicato rispetto alla controparte. Facendo alcuni semplici calcoli possiamo infatti osservare come in media nel dataset siano presenti 27 recensioni per ogni articolo in formato fisico contro le 124 per gli articoli in formato *streaming*. A seguito di questa analisi si è deciso di modificare la variabile *style* presente nei metadati, accorpando la categoria "Amazon Video" alla categoria "Amazon Video - DVD", in quanto in primo luogo la numerosità delle osservazioni presenti nella categoria di riferimento risulta troppo esigua, e in secondo luogo per i 16 articoli osservati, si può pensare ad una mancanza di recensioni in formato fisico più che a una totale assenza dell'articolo nel suddetto formato. La variabile *style* nei metadati presenta ora solo due categorie, ovvero "Amazon Video - DVD" e "DVD", riconducibili a prodotti in formato "*streaming* e fisico" e "fisico".
Infine si è deciso di riportare in Figura 3.5 l'istogramma riferito al prezzo degli articoli, in cui si può notare come la distribuzione del prezzo sia centrata attorno ai 10 euro con una lunga coda a destra caratterizzata dai prezzi più elevati. In aggiunta si possono facilmente notale alcuni picchi ad intervalli costanti partendo dai $9,99$ euro, che si possono ricondurre ad una tecnica di marketing chiamata *left digit effect* o *charm price*, che consiste nell'imporre un prezzo di finta convenienza che termina con i 99 centesimi, con lo scopo di indurre il consumatore all'acquisto. Si è ritenuto necessario fare un breve appunto su questa variabile, in quanto sebbene risulti molto informativa presenta un elevata quantità di valori mancanti, ovvero il 47%, difficilmente trattabili con metodi di imputazione. Si è comunque deciso di non eliminarla e valutarne l'utilizzo in seguito.

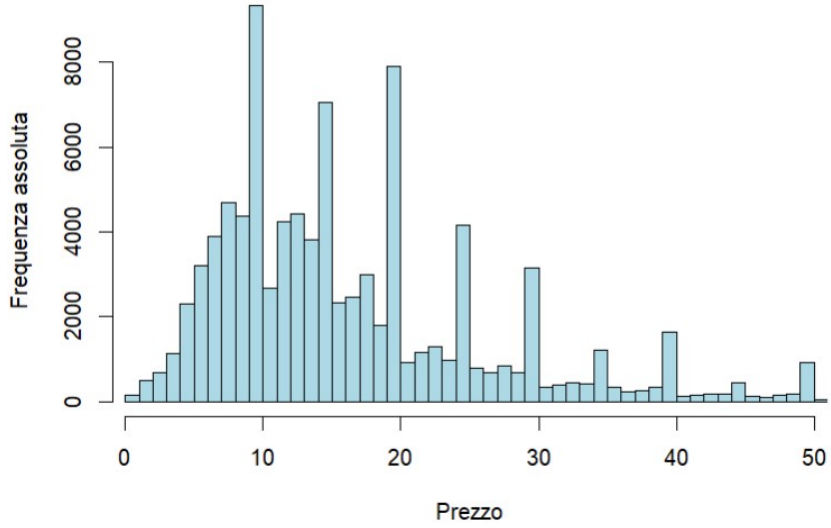


Fig. 3.5: Istogramma per la frequenza del prezzo degli articoli.

## 3.2 Applicazione delle regole associative

Il contesto degli acquisti *online* si presta nel migliore dei modi nell'estensioni del concetto di base delle regole associative, però nella maggior parte dei casi è utile rivalutare il significato di transazione commerciale. Infatti nell'applicazione classica della metodologia, ogni soggetto che compie una transazione, acquista di conseguenza un numero di *items* variabile, che in primo luogo dipende sicuramente dal tipo di esercizio commerciale, e in secondo luogo può essere influenzato dalla distanza fisica per il raggiungimento del negozio. Mentre nel caso degli acquisti *online* il fattore della distanza viene ovviamente a mancare, e il numero di articoli acquistati per transazione si riduce in maniera drastica; sebbene in alcuni siti di *e-commerce* sia presente una spesa minima per la spedizione gratuita, che di conseguenza può influenzare il numero di prodotti acquistati per singola transazione.
Nel caso di Amazon, e soprattutto nella categoria di prodotti che si sta considerando per l'analisi, questi fattori vengono sicuramente incentivati, con un numero di articoli per transazione quasi sempre pari a uno, che rende ovviamente inapplicabili le metodologie relative alle regole associative. Per ovviare al problema si è dunque deciso di riunire tutti gli items acquistati da un singolo utente sotto un'unica transazione, in modo tale che ad ogni soggetto venga riconosciuto un unico vettore di prodotti. Questa operazione è stata effettuata grazie alla presenza nelle transazioni in oggetto di un nome utente univoco riconducibile ad un *account* (necessario per svolgere acquisti sul sito). Una possibile alternativa a questa procedura è quella di dividere il vettore di prodotti in sotto vettori, in modo tale

da creare più transazioni per singolo utente, scegliendo come principio di divisione il tempo passato tra l'acquisto di due articoli sequenziali.
La finalità di questa analisi e dunque quella di individuare le associazioni più interessanti che collegano due o più articoli, e successivamente riconoscere le caratteristiche dei prodotti che ne favoriscono l'abbinamento, con lo scopo di identificare le regolarità più interessanti nel comportamento d'acquisto.
Innanzitutto per svolgere tali analisi si è resa necessaria una riorganizzazione del dataset riferito alle recensioni. Come prima cosa si è notata la presenza di recensioni duplicate per articoli identici ma con formato differente, infatti Amazon aggrega tutte le recensioni riferite al medesimo film seppure appartenenti a supporti differenti (Blu-ray, DVD, Amazon Video...), e allo stesso tempo genera un Asin distinto per ogni tipologia di formato. Questo aspetto fa dunque nascere un grosso problema nella creazione delle regole associative, in quanto porta alla generazione di regole insensate che collegano medesimi prodotti aventi Asin differenti con un livello di *confidence* pari a 1. Per risolvere tale problema si è recuperato il testo delle recensioni, che è stato poi collegato al dataset di interesse tramite un'operazione di *merge*, e infine si sono rese uniche tutte quelle righe aventi data della recensione, nome utente, e testo della recensione identici.
Successivamente, dato che il dataset delle recensioni risultava comunque troppo esteso per l'utilizzo dell'algoritmo Apriori (circa 180 mila Asin e quasi 4 milioni di utenti), si è resa necessaria una selezione mirata che garantisse lo svolgimento dell'analisi. Nello specifico si sono scelti due criteri di selezione, ovvero la data in cui è stata scritta la recensione, dando la precedenza a quelle più recenti, e il numero di recensioni effettuate per cliente, ritenendo gli utenti con un numero di acquisti maggiore più interessanti. Quindi a tal fine si sono prese in considerazione le recensioni dal 03-10-2012 al 03-10-2018, ovvero per gli ultimi 6 anni disponibili, e gli utenti con un minimo di 50 ordini totali effettuati, giungendo così ad ottenere 2.561 utenti differenti e 56.715 articoli specifici. Successivamente a questa operazione si è dunque costruita una matrice di transizione avente il numero di righe pari ai diversi valori di *reviewerID*, ossia 2.561, e il numero di colonne pari al numero totale di Asin presenti nel nuovo dataset, ovvero 56.715.
A questo punto, tramite la matrice di transizione appena creata, è stato possibile utilizzare l'algoritmo Apriori. Nel suo utilizzo si è deciso di porre dei valori di soglia per i termini di fiducia e supporto, rispettivamente *minfid*= 0.645 e *minsup*= 0.0142, in modo tale da identificare solamente le regole più interessanti.
Inizialmente si sono ottenute 198 regole, che sono poi state ridotte a 176 tramite l'eliminazione di quelle ridondanti. Il principale motivo della ridondanza è il fatto

che il numero delle regole associative cresce al crescere degli *itemset* frequenti, e di conseguenza l'algoritmo genera un numero di regole associative più elevato di quello effettivamente necessarie, dando ad alcune regole un significato che si può riscontrare in altre.
L'eliminazione di tali regole è stata operata tramite la logica per cui una specifica regola risulta ridondante se esistono regole più generali con una fiducia uguale o maggiore. Per il medesimo scopo si potevano utilizzare altre misure come ad esempio il *lift*.
In seguito si procede con l'analisi delle regole ottenute. Nel dettaglio si sono estratte 176 regole, le quali statistiche descrittive generali sono riassunte nella tabella 3.1.

| | **supporto** | **fiducia** | **lift** |
|---|---|---|---|
| Minimo | 0.0145 | 0.645 | 5.587 |
| Primo Quantile | 0.0148 | 0.667 | 6.530 |
| Mediana | 0.0156 | 0.696 | 7.361 |
| Media | 0.0163 | 0.707 | 7.575 |
| Terzo Quantile | 0.0171 | 0.731 | 8.474 |
| Massimo | 0.0246 | 0.878 | 10.636 |

Tab. 3.1: Statistiche descrittive relative al supporto, fiducia e *lift*.

Dalla tabella possiamo visualizzare la distribuzione delle 3 principali misure, il supporto la fiducia e il *lift*. Le prime due hanno un valore minimo molto prossimo a quello prestabilito a inizio analisi, ovvero pari a 0.0145 per la prima e 0.645 per la seconda, mentre il loro valore massimo è rispettivamente pari a 0.0246 e a 0.878, da cui ne consegue un intervallo di valori non eccessivamente esteso. Il lift invece presenta una variabilità più elevata delle precedenti misure, e va da un minimo di 5.587 ad un massimo di 10.636.
Per visualizzare al meglio la distribuzione delle regole associative in funzione delle tre misure di valutazione, si riporta nella figura 3.6 uno *scatter plot* rappresentativo di tutte le 176 regole, nel quale nei due assi principali viene riportato il supporto e la fiducia, mentre il *lift* viene identificato dalla differente gradazione di colore assunta dai punti.
Innanzitutto, dal grafico si può notare la distribuzione delle regole in base ai valori assunti dalle tre misure. Si può infatti osservare una più alta concentrazioni verso valori di fiducia e supporto relativamente più bassi, contrapposta da una quasi assenza di regole nello spazio con valori più elevati di entrambe le misure. Invece per quanto riguarda il *lift* le unità sembrano distribuirsi in maniera abbastanza omogenea, e non si riescono dunque ad identificare vari raggruppamenti o *pattern* rilevanti

nel grafico.

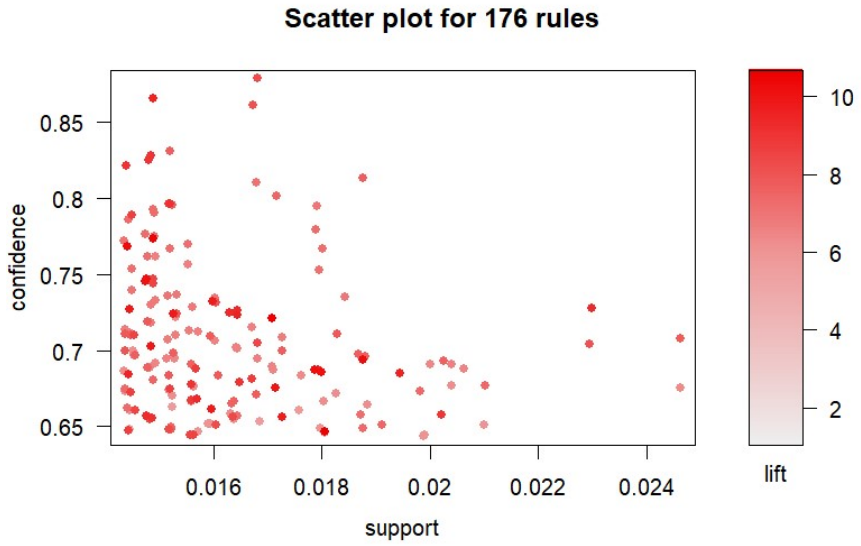


Fig. 3.6: *Scatter plot* rappresentativo di tutte le regole associative relativamente a supporto, fiducia e *lift*.

Un'ulteriore aspetto interessante è visualizzabile nelle figura 3.7, nella quale si può osservare la distribuzione delle regole in base alla loro fiducia, supporto e ordine. Si può innanzitutto notare come la totalità delle regole siano caratterizzate da 3 o 4 prodotti, e che la maggior parte delle associazioni ne comprendono 3, ovvero il 91%.

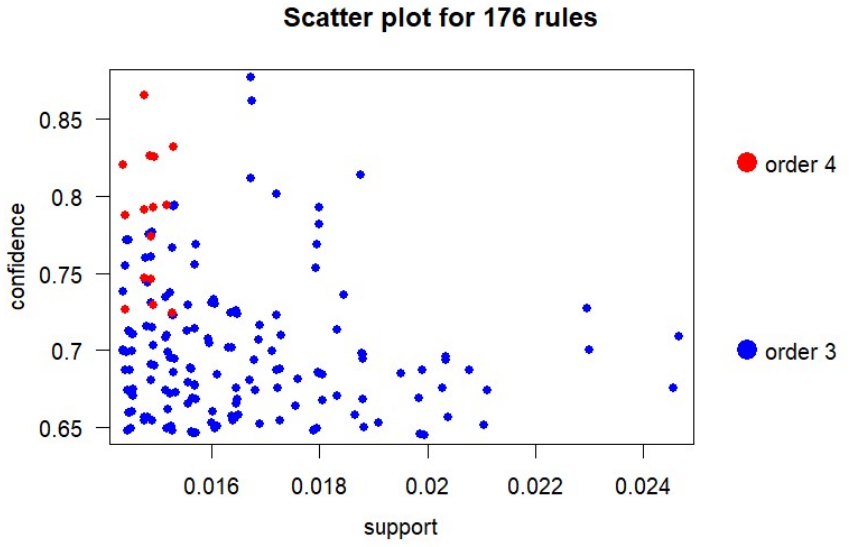


Fig. 3.7: *Scatter plot* rappresentativo di tutte le regole associative relativamente a supporto, fiducia e ordine.

Si può facilmente vedere come le regole con un'ordine maggiore, ossia quelle con 4 articoli, siano concentrate su valori del supporto più vicini all'estremo inferiore e valori

di fiducia più elevati. Questo comportamento risulta sensato per quanto riguarda il supporto, infatti in una transazione è più facile e più frequente ritrovare un *itemset* contenente pochi elementi rispetto ad un suo corrispettivo con più prodotti. Un ultimo aspetto interessante riguarda il *lift*, che per le regole con un numero di articoli maggiore risulta assumere livelli più elevati, infatti in media si registra un valore pari a 7.47 per le regole che comprendo tre articoli e un valore di 8.73 per quelle con un numero superiore.
Nel seguito si è deciso di analizzare più nel dettaglio le regole ottenute, partendo dall'analisi di quelle con un valore di *lift* più elevato e successivamente suddividendo il grafico in figura 3.6 in 3 differenti *clusters* di interesse. Si è quindi deciso di utilizzare il grafico interattivo per creare un *cluster* contenente le regole con maggior supporto, uno contenente quelle con fiducia più elevata e uno con le regole rimanenti, ovvero quelle di minor interesse. Nello specifico si sono analizzate le seguenti regole:

- Come primo gruppo di regole si sono scelte quelle con un valore di *lift* più elevato, ovvero con $lift > 10$. Nella tabella 3.2 si possono visualizzare le regole selezionate con le relative misure e informazioni.

| Regole | Supporto | Fiducia | Lift |
|---|---|---|---|
| {B0090SI3EI,B00NYC65M8} => {B00ZGDIGZ2} | 0.018 | 0.648 | 10.636 |
| {B00DY64A3U,B00G3D732Q} => {B00G7M190U} | 0.017 | 0.721 | 10.617 |
| {B0090SI3EI,B009934S5M,B00D91GRA4} => {B00BL1BJFW} | 0.015 | 0.776 | 10.398 |
| {B00DY64A3U,B00GMV8M2Y} => {B00G7M190U} | 0.019 | 0.696 | 10.239 |
| {B009934S5M,B00ZGDIGZ2} => {B0090SI3EI} | 0.014 | 0.771 | 10.176 |
| {B00BEIYSL4,B00DY64A3U} => {B00G7M190U} | 0.018 | 0.687 | 10.105 |
| {B005S9ELM6,B00DY64A3U} => {B00G7M190U} | 0.018 | 0.687 | 10.105 |

Tab. 3.2: Regole con *lift* maggiore di 10.

Nella tabella 3.3 sono riportati gli Asin con i corrispettivi titoli dei film, in modo tale da comprendere al meglio quali articoli sono effettivamente coinvolti nelle regole create.
Si può anzitutto notare che la maggior parte degli articoli presenti nelle associazioni corrispondono a film abbastanza famosi, e che tra i 22 Asin coinvolti ce ne sono solo 12 diversi. Questo è sicuramente riconducibile al fatto di aver scelto un supporto minimo per la creazione delle regole, infatti i film più conosciuti saranno di conseguenza quelli più acquistati, e si presteranno nel migliore dei modi nella creazione delle associazioni.
Si è poi deciso di analizzare nel dettaglio la prima regola, ovvero quella con *lift* maggiore. Questa regola sta ad indicare che gli Asin "B0090SI3EI" e "B00NYC65M8", che corrispondo al film di *"Captain America: The Winter*

*Soldier"* e a quello di *"Jurassic World"*, si trovano nelle transazioni insieme all'Asin "B00ZGDIGZ2", che corrisponde al film di *"Ant-Man"*, circa l'$1,8\%$ delle volte, e che la probabilità di acquistare il film *"Ant-Man"* dato l'acquisto di *"Captain America: The Winter Soldier"* e *"Jurassic World"* è circa pari al $65\%$.

| Asin | Film |
|---|---|
| {B0090SI3EI} | Captain America: The Winter Soldier |
| {B00NYC65M8} | Jurassic World |
| {B00ZGDIGZ2} | Ant-Man |
| {B00DY64A3U} | Gravity 2013 |
| {B00G3D732Q} | Lone Survivor |
| {B00G7M190U} | Captain Phillips Steelbook |
| {B009934S5M} | Star Trek Into Darkness |
| {B00D91GRA4} | Man of Steel |
| {B00BL1BJFW} | Iron Man 3 |
| {B00GMV8M2Y} | American Hustle |
| {B00BEIYSL4} | Saving Mr. Banks |
| {B005S9ELM6} | Argo |

Tab. 3.3: Articoli coinvolti nelle regole con *lift* maggiore di 10.

Dal valore del *lift*, che risulta essere largamente maggiore di 1, si capisce che la capacità predittiva è superiore alla semplice conoscenza del conseguente, e si può dunque affermare la presenza di una forte dipendenza tra i due *itemset*.

- Come secondo gruppo si è deciso di analizzare le regole con un grado di fiducia maggiore, ovvero quelle che nella figura 3.6 si presentano con un valore di tale misura superiore a 0.08. Nella tabella 3.4 sono riportate le regole di interesse con le relative misure.

| Regole | Supporto | Fiducia | Lift |
|---|---|---|---|
| {B00EHK2S04,B00G7M190U} => {B00DY64A3U} | 0.017 | 0.878 | 8.172 |
| {B009934S5M,B00BL1BJFW,B00DY64A3U} => {B00D91GRA4} | 0.015 | 0.864 | 9.412 |
| {B00A6UHC0U,B00G7M190U} => {B00DY64A3U} | 0.017 | 0.860 | 8.009 |
| {B00BL1BJFW,B00D91GRA4,B00GLPCKX8} => {B009934S5M} | 0.015 | 0.830 | 7.617 |
| {B0090SI3EI,B009934S5M,B00BL1BJFW} => {B00D91GRA4} | 0.015 | 0.826 | 9.003 |
| {B001GCUO16,B009934S5M,B00BL1BJFW} => {B00D91GRA4} | 0.015 | 0.826 | 9.003 |
| {B009934S5M,B00A6UHC0U,B00BL1BJFW} => {B00D91GRA4} | 0.014 | 0.822 | 8.960 |
| {B00G7M190U,B00GMV8M2Y} => {B00DY64A3U} | 0.019 | 0.814 | 7.576 |
| {B00XLX0Z62,B00XQ142MW} => {B00NYC65M8} | 0.017 | 0.811 | 6.996 |

Tab. 3.4: Regole con fiducia maggiore di 0.8.

Nella tabella 3.5 vengono riportati gli Asin con i rispettivi nomi degli articoli a cui si riferiscono.

| Asin | Film |
|---|---|
| {B00EHK2S04} | Elysium |
| {B00G7M190U} | Captain Phillips Steelbook |
| {B00DY64A3U} | Gravity 2013 |
| {B009934S5M} | Star Trek Into Darkness |
| {B00BL1BJFW} | Iron Man 3 |
| {B00D91GRA4} | Man of Steel |
| {B00A6UHC0U} | Pacific Rim |
| {B00GLPCKX8} | Thor: The Dark World |
| {B0090SI3EI} | Captain America: The Winter Soldier |
| {B001GCUO16} | X-Men Origins: Wolverine |
| {B00GMV8M2Y} | American Hustle |
| {B00XLX0Z62} | San Andreas |
| {B00XQ142MW} | Mad Max: Fury Road |
| {B00NYC65M8} | Jurassic World |

Tab. 3.5: Articoli coinvolti nelle regole con fiducia maggiore di 0,8.

Come nel caso precedente si nota la presenza di film abbastanza famosi, che in alcuni casi sono identici a quelli presenti nelle regole con *lift* maggiore. Nel complesso le regole con maggiore fiducia hanno un supporto che varia da 0.014 a 0.019, e dei valori di *lift* abbastanza elevati compresi tra 7 e 9.5.
La regola con maggior fiducia si presenta anche come la regola più interessante del dataset, ovvero quella con dei valori delle misure complessivamente migliori. Nello specifico questa regola sta ad indicare che gli Asin "B00EHK2S04" e "B00G7M190U", che corrispondo al film di *"Elysium"* e a quello di *"Captain Phillips Steelbook"*, si trovano nelle transazioni insieme all'Asin "B00DY64A3U", che corrisponde al film di *"Gravity 2013"*, circa l'1.7% delle volte, e che la probabilità di acquistare il film *"Gravity 2013"* dato l'acquisto di *"Elysium"* e *"Captain Phillips Steelbook"* è circa pari all'88%.
Dal valore del *lift*, che risulta essere circa uguale a 8.2, si capisce che la capacità predittiva è superiore alla semplice conoscenza del conseguente, e si può dunque affermare la presenza di una forte dipendenza tra i due *itemset*.

- Come terzo gruppo si è scelto di analizzare le regole con un livello di supporto più elevato, ovvero quelle che nella figura 3.6 si presentano con un valore di tale misura superiore a 0.022.

| Regole | Supporto | Fiducia | Lift |
|---|---|---|---|
| {B009934S5M,B00BL1BJFW} => {B00D91GRA4} | 0.025 | 0.708 | 7.714 |
| {B00BL1BJFW,B00D91GRA4} => {B009934S5M} | 0.025 | 0.677 | 6.218 |
| {B0059XTU1S,B0095HHLMO} => {B0059XTU3G} | 0.023 | 0.728 | 9.281 |
| {B0059XTU3G,B0095HHLMO} => {B0059XTU1S} | 0.023 | 0.702 | 7.959 |

Tab. 3.6: Regole con supporto maggiore di 0.22.

Nella tabella 3.6 sono riportate le regole di interesse con le relative misure. Nella tabella 3.7 vengono riportati gli Asin degli articoli presenti nelle regole con supporto minimo, con i rispettivi nomi dei film a cui si riferiscono.

| Asin | Film |
|---|---|
| {B009934S5M} | Star Trek Into Darkness |
| {B00BL1BJFW} | Iron Man 3 |
| {B00D91GRA4} | Man of Steel |
| {B0059XTU1S} | The Hobbit: An Unexpected Journey |
| {B0095HHLMO} | The Hobbit: The Battle of the Five Armies |
| {B0059XTU3G} | The Hobbit: The Desolation of Smaug |

Tab. 3.7: Articoli coinvolti nelle regole con supporto maggiore di 0.22.

Innanzitutto si può notare che le prime due regole e le seguenti due presentano al loro interno gli stessi articoli, con l'unica differenza relativa alla posizione di essi. Questo comporta che il loro supporto sia in entrambi i casi identico, ma d'altro canto fiducia e *lift* non sono vincolati, ed infatti variano in base alla posizione dei prodotti. Un ulteriore aspetto da tenere in considerazione è il fatto che la terza e di conseguenza anche la quarta regola, comprendono articoli molto affini, infatti presentano al loro interno i film della trilogia di *"The Hobbit"*, e dunque la regola che si viene a creare risulta in un qual senso essere abbastanza banale, e di conseguenza poco interessante.

- Infine come ultimo gruppo si prendono in considerazione tutte quelle regole con fiducia inferiore a 0.08, supporto inferiore a 0.022 e *lift* inferiore a 10, ovvero quelle non considerate nei tre *clusters* precedenti. In totale si hanno dunque 155 regole, e nella tabella 3.8 vengono riportate alcune statistiche descrittive per quanto riguarda supporto fiducia e *lift*.

| | **supporto** | **fiducia** | **lift** |
|---|---|---|---|
| Minimo | 0.0145 | 0.645 | 5.587 |
| Primo Quantile | 0.0148 | 0.667 | 6.503 |
| Mediana | 0.0156 | 0.691 | 7.261 |
| Media | 0.0161 | 0.698 | 7.404 |
| Terzo Quantile | 0.0168 | 0.723 | 8.260 |
| Massimo | 0.021 | 0.796 | 9.991 |

Tab. 3.8: Statistiche descrittive relative al supporto, fiducia e lift.

Come ci si poteva aspettare i valori che sono cambiati maggiormente sono quelli relativi alla coda destra della distribuzione. Sebbene queste regole risultino avere dei valori delle misure più bassi, non vuol dire che esse siano di poca

utilità. Infatti presentano comunque delle regole efficaci con valori di supporto, fiducia e *lift* più che accettabili, ma solamente risultano meno interessanti delle precedenti.

Infine si è deciso di proporre due rappresentazioni grafiche, per comprendere al meglio le regole più interessanti, e visualizzare i nodi maggiormente presenti nelle relazioni, nonché quelli che ricevono più connessioni. Si sono dunque applicati il *Parallel coordinates plot* e la *Grouped matrix-based visualization*. Per quanto riguarda il primo grafico, ovvero il *Parallel coordinates plot*, è utilizzato per visualizzare dati multidimensionali, in cui ogni dimensione viene rappresentata separatamente nell'asse delle $x$, mentre l'asse delle $y$ viene condiviso. Nel nostro caso nell'asse delle $y$ vengono riportati i nomi degli Asin, mentre in quello delle $x$ la posizione di ogni regola. Infine viene utilizzata una freccia la cui punta indica l'elemento conseguente. Perché il grafico risulti leggibile è necessario proporre un numero di regole non troppo elevato, quindi in Figura 3.8 si è deciso di riportare le 10 regole con maggior fiducia.

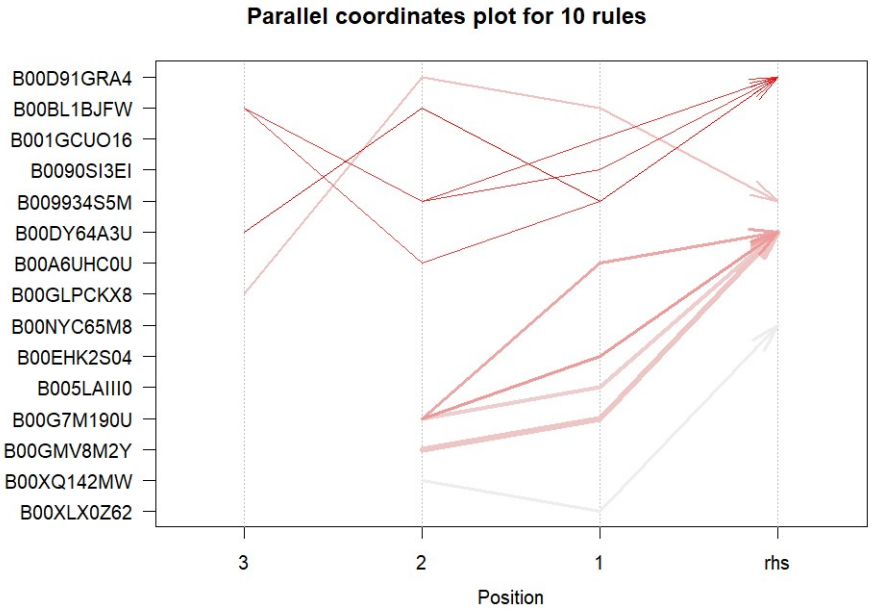


Fig. 3.8: *Parallel coordinates plot* per le 10 regole con maggior fiducia.

Nel grafico la larghezza delle frecce rappresenta il supporto, mentre l'intensità del colore il *lift*. Dal *Parallel coordinates plot* si può notare come vi siano 2 principali Asin in cui confluiscono la maggior parte delle regole, ovvero "B00D91GRA4" e "B00DY64A3U", che corrispondono rispettivamente al film *"Man of Steel"* e al film *"Gravity 2013"*. Dalla tipologie delle frecce si può invece notare come questi due articoli vincolino i valori delle misure delle regole associate, infatti avendo il primo

film come conseguente si riscontrano valori di *lift* molto elevati contrassegnati da una tonalità più scura, mentre avendo il secondo film come conseguente si ottengono delle regole con un supporto più alto, contraddistinte da un maggiore spessore delle frecce.

Per quanto riguarda la *Grouped matrix-based visualization*, è una tecnica ideale nel caso in cui si disponga di dataset molto densi, infatti è basata sul raggruppamento delle regole tramite *clustering*. L'idea di base di questo approccio è che gli antecedenti dipendenti dagli stessi conseguenti siano simili, e quindi possano essere raggruppati insieme. Per la creazione del grafico si sono utilizzate le 70 regole con maggior fiducia, in modo tale da apportare abbastanza informazioni per rendere il risultato interessante, e al contempo non creare troppi gruppi che renderebbero l'*output* dispersivo. Il grafico riporta nelle colonne i gruppi di antecedenti più importanti, mentre nelle righe gli Asin dei conseguenti. All'interno della griglia sono presenti dei punti che collegano i gruppi di antecedenti al corrispettivo conseguente, e questi punti, con una logica simile al grafico predente, assumano una graduazione differente in base al valore del *lift*, e una grandezza maggiore all'aumentare del supporto. In aggiunta le regole vengono riordinate in modo tale che che il *lift* diminuisca dall'alto verso il basso e da sinistra verso destra, il che porta ad ottenere la regola con maggior *lift* nell'angolo in alto a sinistra.

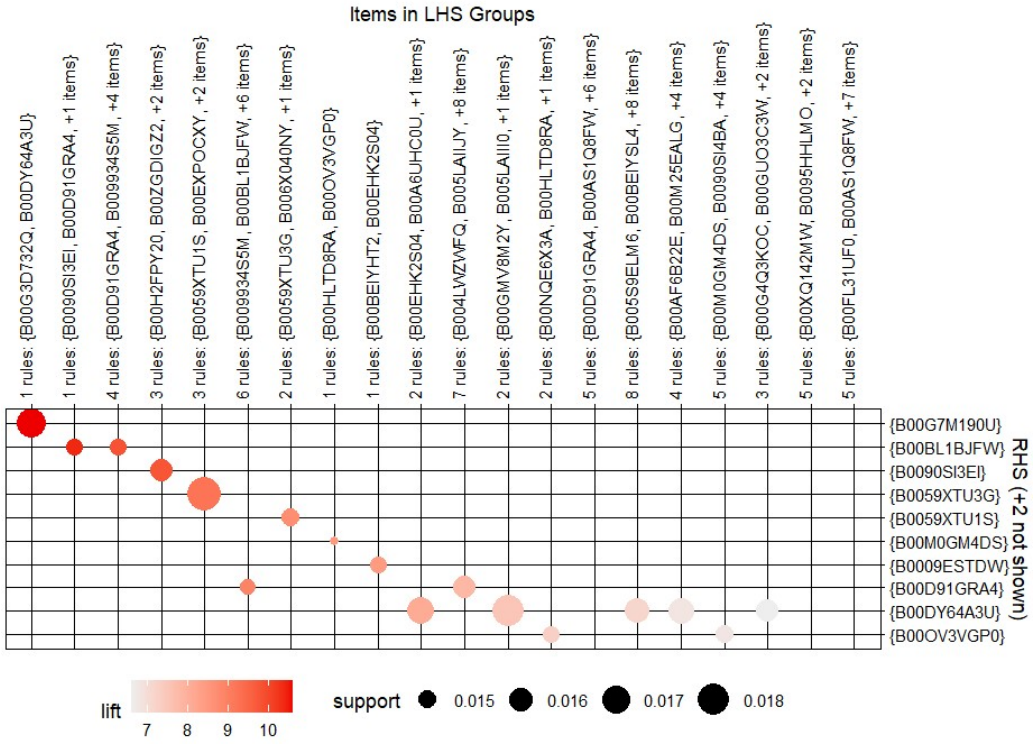


Fig. 3.9: *Grouped matrix* per le 70 regole con maggior fiducia raggruppate in 10 gruppi.

Dall'analisi del grafico si può vedere che il valore maggiore di regole con in parte gli stessi antecedenti è 8, infatti gli Asin "B005S9ELM6" e "B00BEIYSL4" sono presenti in ben 8 regole insieme ad un massimo di altri 8 *items*, e hanno in tutti i casi come conseguente l'Asin "B00DY64A3U", che corrisponde al film *"Gravity 2013"*. Sebbene questo appena esposto sia il caso in cui si ottiene il numero di regole maggiore con lo stesso antecedente, è importante confrontarlo anche per i valori di supporto e *lift*, infatti potrebbero risultare più interessanti raggruppamenti con un numero di regole minore ma con valori delle misure più elevati. Per esempio potrebbero risultare più interessanti le 3 regole con antecedenti gli Asin "B0059XTU1S" e "B00EXPOCXY" insieme ad un massimo di altri 2 *items*, e che hanno come conseguente l'Asin "B0059XTU3G", in quanto presentano valori si supporto e *lift* maggiori. Il medesimo ragionamento appena proposto può essere esteso per tutti gli altri insiemi di regole.

## 3.3 Costruzione delle reti

Utilizzando il dataset riferito alle caratteristiche dei prodotti, e più nello specifico tramite le variabili Asin e *Also_buy*, che contengono rispettivamente il codice identificativo del film e il vettore degli Asin collegato al singolo prodotto, è possibile costruire la rete dei film del catalogo di Amazon. Per fare ciò è necessario costruire una matrice di adiacenza diretta e binaria, che riesca dunque a contenere i collegamenti tra i vari prodotti, indicando con 1 la presenza di una connessione e con 0 l'assenza di essa.
Però, come si è osservato nella precedente sezione, il dataset è composto da circa 180.000 prodotti, che rendono molto difficile la creazione della rete nella sua interezza, infatti oltre ad una difficoltà grafica e rappresentativa, il problema principale risiede nel costruire la matrice di adiacenza e nello svolgimento delle successive analisi collegate ad essa, in quanto il costo computazionale risulterebbe eccessivamente elevato. Risulta dunque conveniente l'analisi di un sottogruppo di prodotti tramite una selezione di essi, in modo tale da condurre un'analisi su una parte della rete per cercare di capirne comunque il comportamento complessivo.
Un primo passo risiede nella selezione dei prodotti in base alla variabile *Also_buy*, che come si è precedentemente sottolineato contiene tutti prodotti acquistati successivamente a quello di interesse. Questa variabile può contenere da un minimo di 0 ad un massimo 100 codici identificativi, ed è importante ricordare che se un prodotto presenta un valore nullo nella variabile *Also_buy*, non significa che non sia collegato alla rete, in quanto potrebbe avere delle connessioni dirette provenienti da

altri nodi. Il criterio di selezione utilizzato è dunque stato la numerosità di codici identificativi presenti nella variabile *Also_buy*, tenendo in considerazione due principali fattori, ovvero che un numero troppo piccolo non permetteva di massimizzare la densità della rete, e un numero troppo elevato implicava la creazione di reti troppo estese con connessi problemi computazionali. Si è dunque deciso, dopo alcuni tentativi, di selezionare un *range* che va da un minimo di 5 ad un massimo di 20 codici identificativi per prodotto.
Una volta selezionati tutti gli articoli con tale caratteristica, è stato creato un dataset con una variabile contenente gli identificativi degli articoli osservati, e 20 variabili per tutti gli identificativi compresi in *Also_buy*. Successivamente per fare in modo di avere per ogni riga un solo collegamento tra un nodo emittente e un nodo ricevente, è stato modificato il formato da *wide* a *long*, così da ritornare ad avere 2 variabili, ovvero Asin e *Also_buy*.
Il secondo passo nella creazione delle reti è stato quello di selezionare uno tra i nodi più popolari, in modo da massimizzare la densità di rete, e per fare ciò è stato creato un vettore contenente tutti i nodi presenti in *Also_buy*, che è stato successivamente ordinato in base alla popolarità di essi, in modo tale da riuscire a visualizzare quelli maggiormente di interesse e che di conseguenza ricevono il maggior numero di collegamenti.
In seguito è stato ridimensionato il dataset tramite il vincolo di mantenere solamente gli Asin degli articoli che presentano in *Also_buy* l'articolo di riferimento fissato a priori. Infine per completezza della rete vengono aggiunti al dataset i possibili collegamenti tra gli articoli compresi in *Also_buy* verso quelli compresi in Asin o *Also_buy*, e viene dunque creata la matrice di adiacenza diretta e binaria riferita al dataset appena creato.
In aggiunta per ogni rete così creata vengono considerate le seguenti variabili di nodo:

- *max_categories*: variabile qualitativa nominale indicante la categoria maggiormente agglomerativa in cui trovare l'articolo, senza tenere in considerazione quella generale che comprende tutti gli articoli (*Movies* & TV);

- *min_categories*: variabile qualitativa nominale indicante la categoria maggiormente divisiva in cui trovare l'articolo, che nella maggior parte dei casi corrisponde al genere del film;

- *num_review*: variabile quantitativa discreta che indica il numero totale di recensioni disponibili;

- *num_categories*: variabile quantitativa discreta indicante il numero totale di categorie in cui è possibile trovare l'articolo;

- *avg_review*: variabile quantitativa discreta che può assumere valori da 1 a 5 ed indica la valutazione media del prodotto;

- *style*: variabile qualitativa nominale che indica il formato in cui è stato venduto l'articolo;

- *rank*: variabile quantitativa discreta indicante la posizione nella classifica degli articoli più venduti.

Successivamente all'identificazione degli articoli compresi nella rete e alla conseguente costruzione della matrice di adiacenza, è possibile creare la matrice delle variabili di nodo. Tale matrice è facilmente ottenibile tramite un'operazione di *merge*, e il procedimento è riassumibile in due semplici *step*. Il primo è quello della creazione di una matrice, comprensiva di tutti gli articoli e delle variabili di interesse con l'aggiunta dell'Asin, mentre il secondo step consiste appunto nell'effettuare un'operazione di *merge* nella quale saranno presenti due differenti dataset, ovvero quello appena creato con le variabili di nodo e uno comprensivo solamente degli Asin presenti nella rete.
Nella creazione delle reti è importante specificare che si è deciso di fissare a priori un numero di articoli limitato, questa scelta è stata effettuata principalmente per due ragioni, la prima è quella della densità di rete che tenderebbe a diminuire, rendendo le successive analisi più complicate, mentre il secondo motivo risiede nella difficoltà nel trovare articoli le cui reti possano essere interconnesse. Infatti nella maggior parte dei casi l'inclusione a priori di più articoli comporta la creazione di una rete globale suddivisa in reti più piccole non interconnesse. Questo problema è riconducibile al fatto di avere una rete di articoli di grandi dimensioni, e nell'impossibilità di selezionare gli articoli più popolari, in quanto comporterebbero la creazione di reti di dimensione troppo elevata con conseguenti costi computazionali. Si è dunque deciso di creare 3 differenti reti utilizzando come principali fattori distintivi il numero di articoli posti a priori e la categoria in cui è inserito l'articolo. A tale scopo si è dunque utilizzata la variabile *min_category*, al fine di selezionare film con generi differenti.
Innanzitutto per la prima rete si è optato per l'utilizzo di due articoli a priori della medesima categoria, ovvero *Kids & Family*, ed è stata così creata una rete contenente 70 nodi. Si è deciso di effettuare questa scelta in modo tale da facilitare i collegamenti tra i nodi.

Per quanto riguarda la seconda rete sono stati selezionati a priori due articoli di categorie differenti, ovvero *Documentary* e *Drama*, cercando dunque di unire due prodotti molto frequenti e di genere differente in un'unica rete. Da tale processo si è generata una rete contenente 100 nodi.
Infine per la terza rete si è deciso di selezionare a priori tre articoli di categorie differenti, ovvero *Western*, *Action & Adventure* e *Horror*, cercando quindi di estremizzare il concetto del punto precedente, e creando in tal modo una rete più grande delle precedenti, contenente 147 nodi.

## 3.4 Statistiche descrittive di rete

Nel seguito, per ogni rete costruita come enunciato nella sezione precedente, sono stati riportati alcuni grafici rappresentativi e alcune statistiche descrittive, con lo scopo di comprenderne al meglio le caratteristiche e gli aspetti più interessanti di ogni rete.

#### Rete con due nodi a priori della stessa categoria

Per la costruzione della prima rete si sono fissati a priori due articoli appartenenti alla medesima categoria *"Kids & Family"*, ovvero:

- *Super Mario World*: *The Complete Series*
- *Best of the Adventures of Sonic the Hedgehog*

La rete generata è visualizzabile nella Figura 3.10, ed è stata costruita tramite una matrice di adiacenza di dimensione $70 \times 70$, che di conseguenza implica l'impiego di 70 nodi.
Dal grafico riportato in figura si può facilmente notare la presenza di tre differenti *clusters*, due più piccoli ai lati e uno più grande al centro, nel quale risiedono i nodi scelti a priori. La partecipazione di questi due nodi riveste un ruolo sicuramente centrale nella rete, in quanto presentano un numero di connessioni dirette molto elevato, rispettivamente 46 per il primo articolo (in rosso) e 37 per il secondo (in blu).

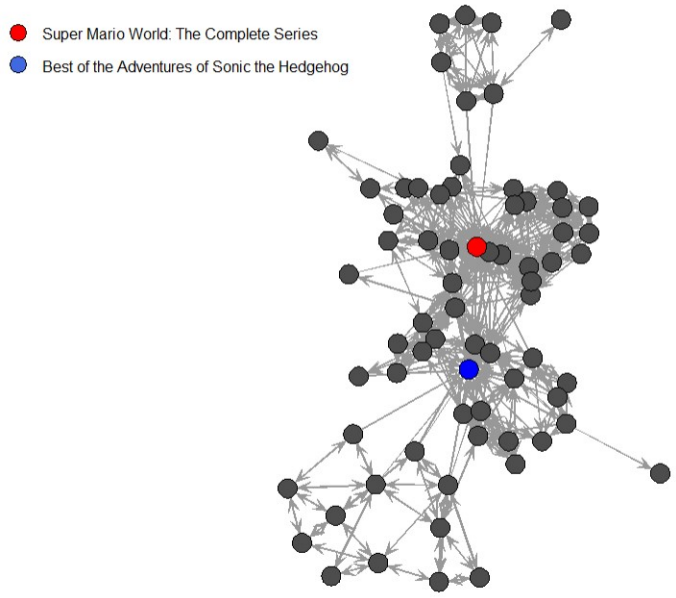


Fig. 3.10: Rete con nodi fissati a priori.

In una fase successiva, ipotizzando l'assenza dei due articoli scelti a priori, si è notato che la rete riesce comunque a mantenere intatta la sua struttura generale. Questo fatto è sicuramente riconducibile al numero di archi presenti, che risulta in ogni caso abbastanza elevato da permettere una buona connettività globale, infatti valutando la densità di rete si è notata una diminuzione di un punto percentuale, con un valore comunque elevato pari a 0.10.

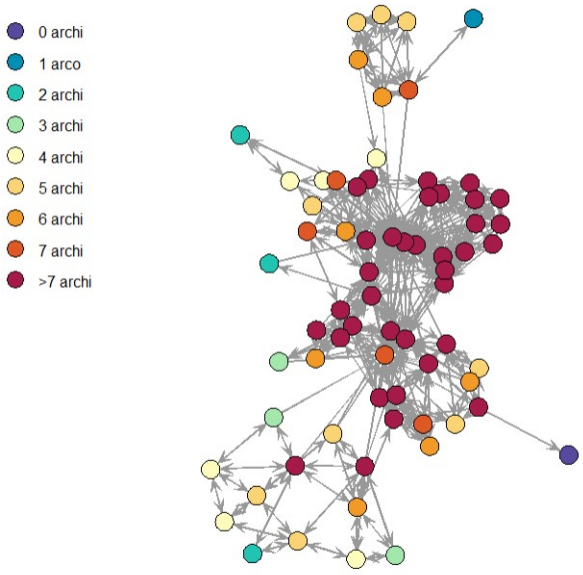


Fig. 3.11: Rete per numero di archi in uscita (*Kids & Family*).

Come si può vedere dalla Figura 3.11, che rappresentata la rete con il numero di archi in uscita presenti per ogni nodo, la maggior parte degli articoli possiede più di 7 connessioni, ovvero 35 nodi su 70, che, come si è notato precedentemente, oltre a garantire una buona connettività globale, garantiscono il buon funzionamento della rete anche in assenza degli articoli posti a priori.
Nel seguito si analizza la rete originale comprensiva di nodi a priori, fornendo nella Tabella 3.9 alcune statistiche descrittive, con lo scopo di capire al meglio le principali caratteristiche di rete.

| **Indice** | **Valore** |
|---|---|
| Densità | 0.1124 |
| Diametro | 7 |
| Transitività | 0.4834 |
| Reciprocità | 0.7624 |
| Assortatività | 0.6719 |

Tab. 3.9: Statistiche descrittive a livello di rete.

Dagli indici sopra riportati si può affermare che la rete risulta abbastanza connessa, in quanto si osservano l'$11,24\%$ di tutti i collegamenti possibili, d'altro canto non si può dire essere molto estesa infatti la distanza del più lungo *shortest path*, che si ricorda essere il cammino più breve tra due nodi, è pari a 7 archi. Dal valore assunto dall'indice di reciprocità si può desumere che il $76,24\%$ degli archi presenti nella rete sono reciproci, e dunque la maggior parte degli articoli presenta un collegamento bidirezionale, per cui se un articolo è presente nei prodotti co-acquistati di un altro, allora è probabile anche il contrario, il che porta ad un'assenza di un effetto causale diretto del tipo se acquisto "A" allora acquisterò anche "B", e quindi non è presente uno specifico ordine nell'acquisto. L'indice di transitività, che esprime la propensione della rete a formare triple transitive è circa pari a 0.48, e suggerisce che il 48% di triple all'interno della rete sono chiuse, mentre l'indice di assortatività, che può variare da $-1$ e 1, risulta essere abbastanza elevato, e denota la presenza di gruppi non banali, ovvero che i nodi presenti nella rete tendono a connettersi con nodi aveni caratteristiche simili.
Una di queste caratteristiche è visibile nella Figura 3.12, in cui viene riportata la rete suddivisa per il genere dei film, e quindi per la variabile *min_category*. Si può inizialmente notare come in tutto siano presenti solo 4 differenti categorie sulle 47 totali, il che può portare a pensare che le categorie presenti nella rete tendano a collegarsi principalmente con loro stesse, formando una specie di *cluster* a sé

stante. Considerando infatti i tre generi più frequenti in questa rete, ovvero *"Kids & Family"* che conta 36 nodi, *"Animation"* che conta 13 nodi, e *"Anime & Manga"* che ne conta 12, possiamo subito ricondurci ad una tipologia di categorie che nella maggior parte dei casi sono destinate ad un pubblico più giovane, o che comunque comprenda come utenti maggiormente interessati coloro sotto una specifica soglia di età. Questi generi di film infatti si prestano in maniere ottimale alla formazione di *clusters* isolati, in quanto oltre alla discriminazione della categoria, che può piacere o meno ad uno specifico utente, c'è in aggiunta una variabile molto importante che può influenzarne l'acquisto del prodotto, ovvero l'età dell'acquirente. Per quanto riguarda i film di genere differente non è infatti scontato trovare un *range* di età che possa risultare significativo per l'acquisto.

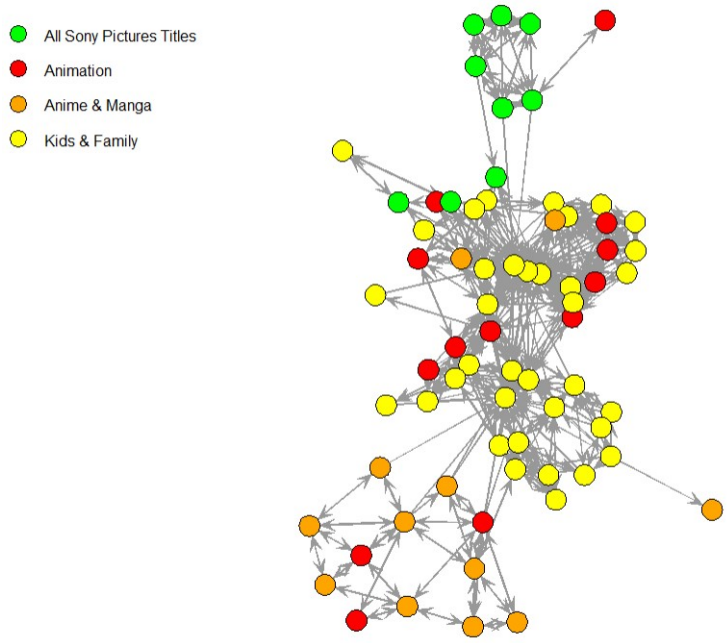


Fig. 3.12: Rete per categoria dei film.

Dalla distribuzione delle differenti categorie nella rete si possono confermare i tre *clusters* indicati precedentemente, ovvero quello centrale di dimensioni più elevate che comprende maggiormente articoli di genere *"Kids & Family"*, e gli altri due laterali che appartengono rispettivamente alle categorie *"Anime & Manga"* e *"All Sony Pictures Titles"*. Il genere *"Animation"*, al contrario dei precedenti, si distribuisce apparentemente in maniera abbastanza casuale nella rete, presentandosi come la categoria più vicina a tutte le altre.

Per quanto riguarda le statistiche a livello di nodo, nella Figura 3.13 sono riportate tre tipologie di reti, che si differenziano per la grandezza dei nodi, i quali sono proporzionali al grado del nodo ($a$), al livello di betweenness ($b$) e al livello di closeness ($c$).

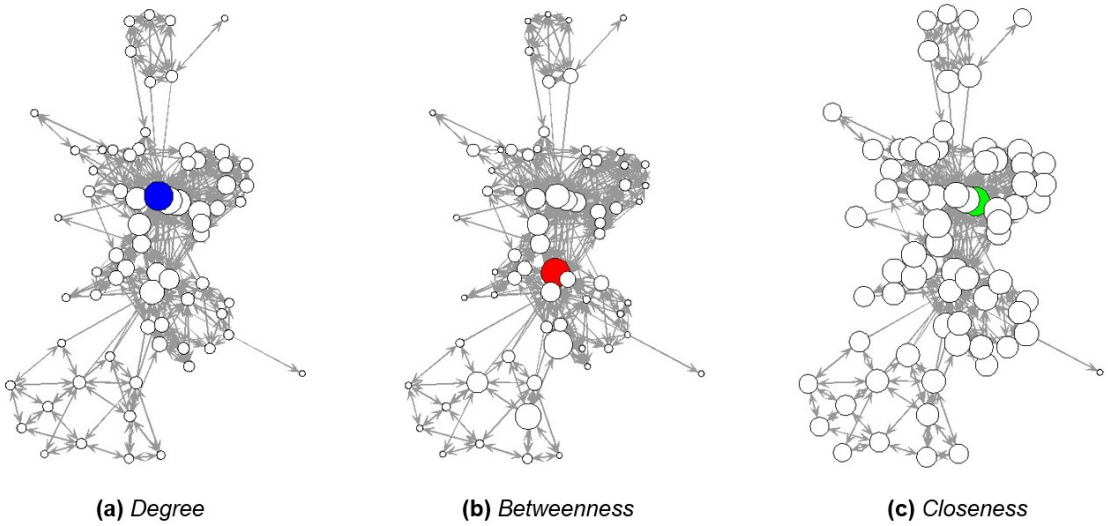


Fig. 3.13: Confronto tra indici di centralità.

Dal grafico possiamo visualizzare anche il nodo con indice di centralità maggiore, che è posto con un colore differente rispetto agli altri. Nella prima rete ($a$) il nodo con grado maggiore corrisponde a uno degli articoli scelti a priori, ovvero *"Super Mario World: The Complete Series"*, il quale è stato scelto anche per l'alta frequenza di connessioni nel dataset. Il livello di maggiore *betweenness* è invece pari a 954.6 ed è corrispondente all'articolo *"Adventures of Sonic the Hedgehog Vol 1"*, che risulta molto affine al secondo nodo selezionato a priori. Per quanto riguarda l'ultimo indice a livello di nodo ($c$), il valore più alto rilevato è 0.48 ed è riferito all'articolo *"Super Mario Bros Super Show Volume 1"*, che in questo caso appartiene alla stessa catena di prodotti dell'articolo con grado maggiore. Un aspetto interessante che si può visualizzare in questi 3 grafici, è la dimensione dei nodi appartenenti alla rete riferita alla *closeness*, che rispetto alle altri due reti che si analizzeranno in seguito, che presentano nodi più grandi principalmente nella parte centrale, essa presenta nodi per la maggior parte di grandi dimensioni. Questo aspetto è sicuramente collegato alla buona densità di rete, che permette alla maggior parte dei nodi di avere un valore di *closeness* piuttosto elevato, e senza particolari differenze rispetto agli altri nodi.

### Rete con due nodi a priori di categorie differenti

Per la costruzione della seconda rete si è deciso di riproporre due articoli a priori, ma questa volta scegliendoli di generi differenti, ovvero *"Drama"* e *"Documentary"*. Gli articoli selezionati sono enunciati di seguito:

- *Family of Strangers (Drama)*
- *Depression*: *Out of the Shadows (Documentary)*

La rete generata è visualizzabile nella Figura 3.14, e comprende una quantità di nodi maggiore rispetto alla rete precedente, infatti per la costruzione della matrice di adiacenza sono stati utilizzati 115 articoli.

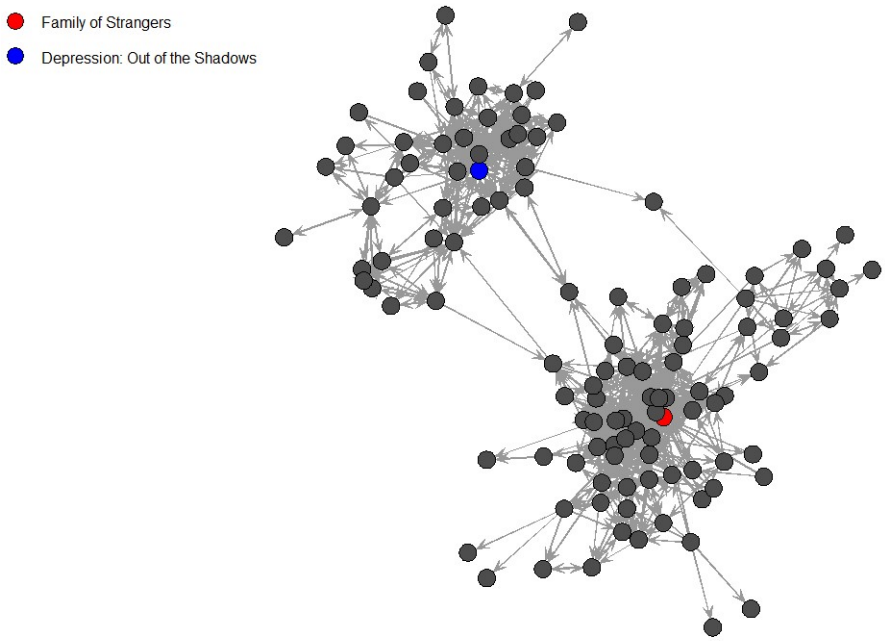


Fig. 3.14: Rete con nodi fissati a priori.

La prima cosa che si nota è la presenza di due *clusters* ben definiti, nei quali vi è la partecipazione di un nodo a priori ciascuno. Questi due articoli scelti in precedenza rivestono sicuramente un ruolo centrale nel gruppo di riferimento, infatti presentano un numero di connessioni molto elevato. La loro esclusione dalla rete non causerebbe però grosse differenze nei due *clusters*, in quanto entrambi i gruppi presentano al loro interno molte connessioni, che portano di conseguenza ad una buona densità di rete. La numerosità dei collegamenti all'interno della rete è rappresentata nella Figura 3.15, che riporta la rete divisa per numero di archi in uscita. Dal grafico si

può appunto notare una elevata presenza di nodi con molte connessioni (54 nodi su 115 hanno più di 7 archi), concentrati maggiormente nel cluster riferito all'articolo di genere *"Drama"*: *"Family of Strangers"*.

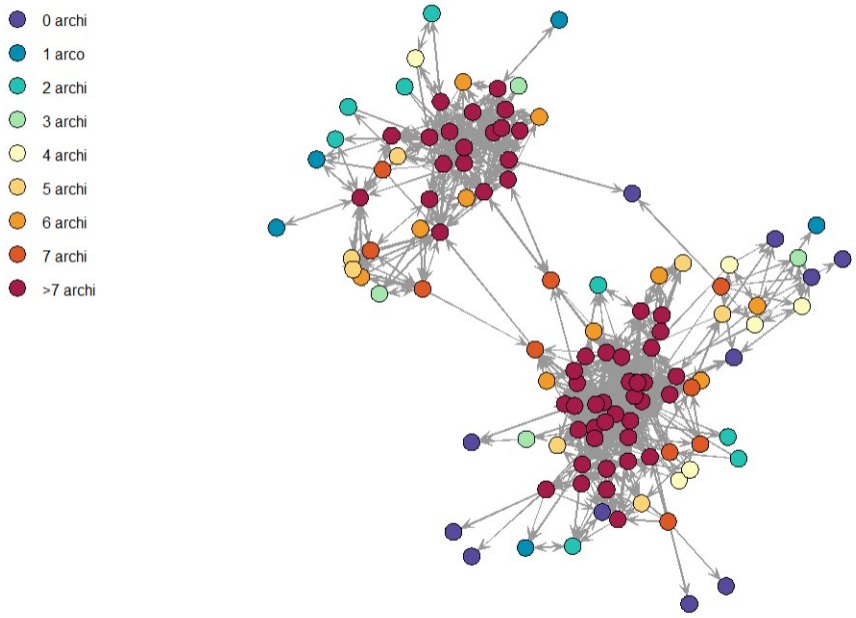


Fig. 3.15: Rete per numero di archi in uscita.

In seguito per capire al meglio le caratteristiche della rete, si sono riportate nella Tabella 3.10 le relative statistiche descrittive messe a confronto con i valori assunti dalla rete precedente, ovvero quella che presentava due articoli a priori della medesima categoria, in modo tale da cercare di comprendere al meglio le principali differenze.

| **Indice** | **Valore rete corrente** | **Valore rete precedente** |
|---|---|---|
| Densità | 0.0755 | 0.1124 |
| Diametro | 9 | 7 |
| Transitività | 0.5074 | 0.4834 |
| Reciprocità | 0.7333 | 0.7624 |
| Assortatività | 0.6329 | 0.6719 |

Tab. 3.10: Statistiche descrittive a livello di rete.

Paragonando il valore degli indici rispetto a quelli della rete precedente, si può affermare la presenza di una minore connettività globale, causata soprattutto dalla

separazione dei nodi in due grandi *clusters* ben distinti. Nel complesso si osserva comunque una densità non troppo bassa, in quanto sono presenti il 7.55% di tutti i collegamenti possibili. Il valore assunto dal diametro risulta essere più elevato delle rete precedente, infatti la distanza del più lungo cammino più breve tra due nodi è pari a 9, causata probabilmente dalla minora connettività e dall'aumento del numero di articoli nella rete. L'indice di reciprocità risulta leggermente più basso e ci indica che il 73.33% degli archi presenti nella rete sono di natura reciproca, e dunque la maggior parte degli articoli presenta un collegamento bidirezionale, per cui se un articolo è presente nei prodotti co-acquistati di un altro, allora è probabile anche il contrario. L'indice di transitività e quello di assortatività risultano molto simili in entrambe le reti; il primo sta ad indicare la propensione a formare triple transitive e risulta circa pari a 0.51, il che vuol dire che il 51% di triple all'interno della rete hanno natura chiusa, mentre il secondo, ovvero l'indice di assortatività, risulta circa pari a 0.63, e sta ad indicare la presenza di gruppi non banali, ossia che i nodi presenti nella rete tendono a connettersi con nodi aventi caratteristiche simili.

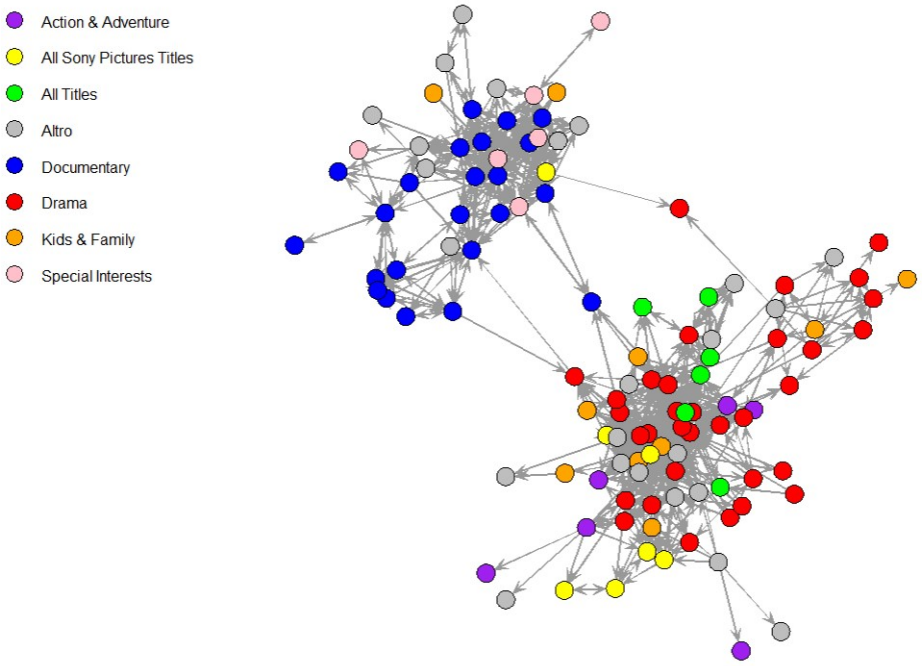


Fig. 3.16: Rete per categoria degli articoli.

Successivamente per cercare di comprendere al meglio le caratteristiche che possono

portare al collegamento tra due nodi, viene riproposta nella Figura 3.16 la rete suddivisa per categorie di appartenenza degli articoli, ovvero in base alla variabile *min_category*. Innanzitutto, per migliorare la visualizzazione del grafico si è eseguito un piccolo accorgimento, accorpando momentaneamente alcune categorie con bassa frequenza ($< 5$) in "Altro", passando quindi da 19 generi differenti a 8. Fatta questa premessa nella rete si sono ottenute 8 differenti categorie, tra le più frequenti *"Drama"* con 33 nodi e *"Documentary"* con 23, che corrispondono anche ai generi degli articoli scelti a priori. Dalla rete si può notare come gli articoli dei generi scelti a priori siano ben circoscritti nel gruppo di riferimento, infatti tutti i prodotti con genere *"Drama"* sono racchiusi nel *cluster* più grande che è stato creato dall'articolo *"Family of Strangers"*, e al contrario tutti i prodotti con genere *"Documentary"* sono racchiusi nel *cluster* creato dall'articolo *"Depression: Out of the Shadows"*. Un ulteriore aspetto interessante è dato dal posizionamento delle categorie con frequenza minore, infatti gli articoli di genere *"Action & Adventure"* sono tutti raggruppati nel *cluster* contenente i film di tipo *"Drama"*, e al contrario gli articoli di tipo *"Special Interests"* raffigurano tutti nel *cluster* contenente i film di tipo *"Documentary"*.

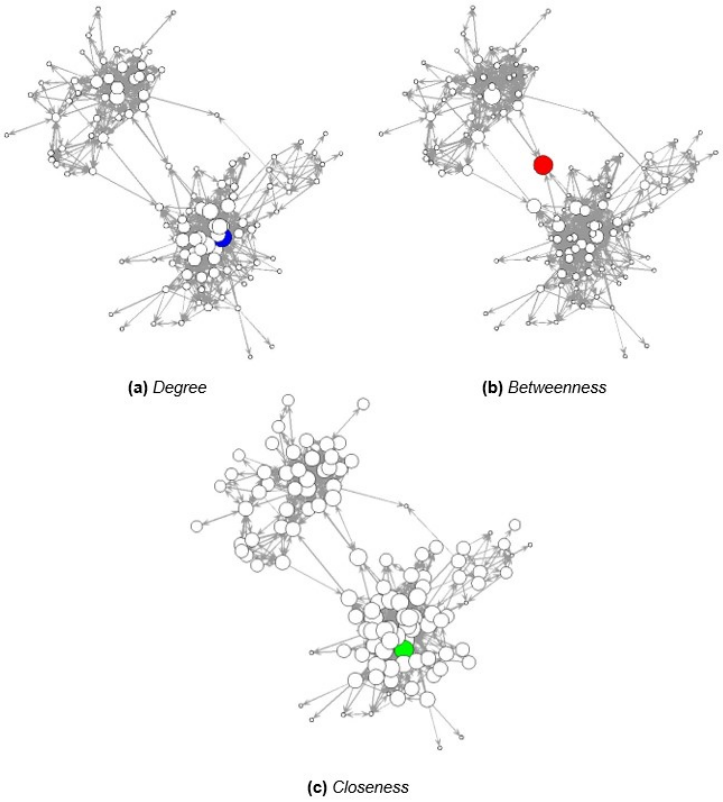


Fig. 3.17: Confronto tra indici di centralità.

Per quanto riguarda le statistiche a livello di nodo, sono riportate nella Figura 3.17 tre tipologie di reti che si differenziano per la grandezza dei loro nodi, che risultano proporzionali al grado del nodo ($a$), al livello di *betweenness* ($b$) e al livello di *closeness* ($c$).
Nella prima rete riferita al grado ($a$), il nodo con dimensione maggiore corrisponde al film *"Family of Strangers"* (in blu), ovvero come nell'altra rete ad uno degli articoli scelti a priori. Invece il nodo con maggior *betweenness*, visibile nel grafico ($b$) (in rosso), riporta un valore molto elevato e pari a $3246,458$, e corrisponde al prodotto *"Cry for He"*, ovvero un film appartenente alla categoria *"Documentary"*. Si ricorda che per *betweenness* si intende la media del numero di volte in cui un nodo si trova nel percorso più breve tra altri due, e dunque questo valore elevato è giustificato dal fatto che i due *clusters* raffigurati sono poco collegati tra loro, e l'articolo di riferimento si presenta come uno dei nodi più utili nel connettere assieme la rete. Per quanto riguarda il terzo e ultimo indice di centralità, vale a dire la *closeness*, il nodo che assume valore maggiore, ovvero pari a 0.45, è visualizzabile nel grafico ($c$) (in verde) e corrisponde all'articolo *"Seeds Of Deception"*, ossia un film appartenente alla categoria *"Drama"*, e quindi presente nel *cluster* di dimensione maggiore contenente tutti gli articoli della medesima categoria.

### Rete con tre nodi a priori di categorie differenti

Per la costruzione della terza rete si sono fissati a priori tre articoli appartenenti a tre categorie differenti, ovvero *"Action & Adventure"*, *"Western"* e *"Horror"*. Gli articoli scelti sono i seguenti:

- *12 Film Action Pack (Action & Adventure)*
- *Cast A Long Shadow (Western)*
- *Yeti: Maneater Series (Horror)*

In questo circostanza il primo articolo, ovvero quello appartenente alla categoria *"Action & Adventure"*, non è come negli altri casi un film a sé stante, bensì è un *box* contenente 12 film, che in ogni caso non crea alcun problema in quanto risultano tutti del mediamo genere. La rete è visualizzabile nella Figura 3.18 ed è stata costruita tramite una matrice di adiacenza di dimensioni più grandi delle precedenti, in quanto comprensiva di 147 nodi.
Da un primo sguardo alla rete si possono notare immediatamente 3 differenti *clusters*, per i quali vi è un nodo a priori ciascuno che funge da centro.

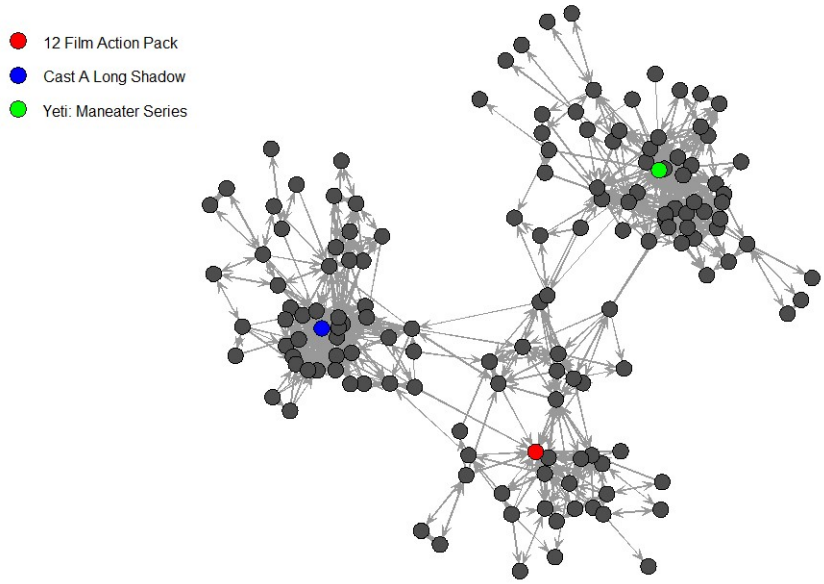


Fig. 3.18: Rete con nodi fissati a priori.

I tre gruppi sono collegati tramite un numero di connessioni abbastanza limitato, che come si è precedentemente affermato, è dovuto alla difficoltà nel trovare articoli le cui reti siano interconnesse. Un ulteriore aspetto non banale che si può notare è quello che interessa il *cluster* dell'articolo *"12 Film Action Pack"*, che in primo luogo sembra fungere da ponte tra gli altri due, e in secondo luogo appare come il *cluster* con meno connessioni interne.

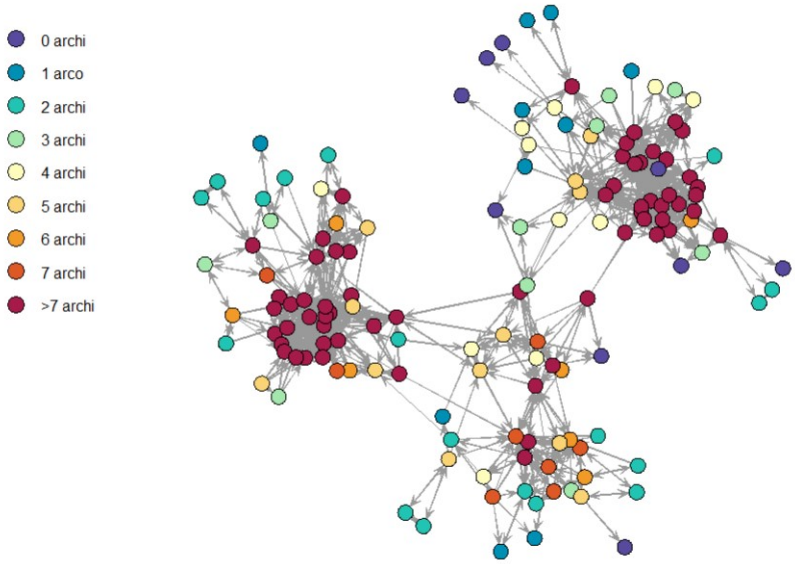


Fig. 3.19: Rete per numero di archi in uscita.

Questo ultimo aspetto si può visualizzare nella Figura 3.19, che riporta la rete per numero di archi in uscita presenti per ogni nodo, infatti nel *cluster* riferito all'articolo di interesse possiamo notare la presenza di soli 6 nodi con più di 7 archi, rispetto ai 62 totali presenti nella rete.
Si è di conseguenza deciso di riportare nella Figura 3.20, la rete senza i nodi a priori precedentemente selezionati, in modo tale da valutarne i cambiamenti più significativi, con una particolare attenzione al *cluster* contenente meno connessioni. Dal grafico si può notare che la rete rimane quasi interamente connessa, a discapito di un singolo nodo che non presenta alcun arco in entrata o in uscita. Per quanto riguarda i due gruppi con un numero di connessioni più elevato, come ci si poteva aspettare, non sono state state riportate grosse modifiche, per il terzo *cluster* invece si può osservare un cambiamento che lo porta alla separazione in due differenti gruppi, tra cui uno che funge da principale connettore tra tutti gli altri.

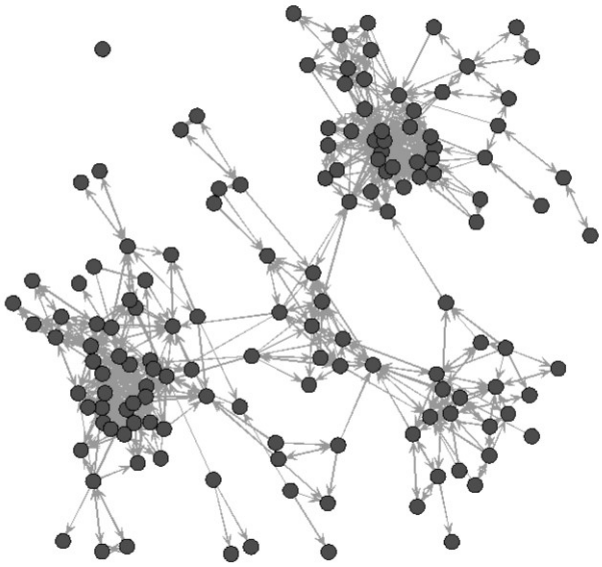

Fig. 3.20: Rete senza nodi a priori.

Per comprendere al meglio le caratteristiche della rete si è deciso di riportare le principali statistiche descrittive nella Tabella 3.11, mettendo a confronto la rete originaria comprensiva di nodi a priori, con quella ristretta, ovvero quella in cui sono stati rimossi.
Dal valore degli indici riportati si può notare che la rete risulta meno connessa delle precedenti, con una densità pari a 0.046 per la rete con nodi a priori, e 0.042 per la rete che non li comprende.

| **Indice** | **Valore rete originale** | **Valore rete ristretta** |
|---|---|---|
| Densità | 0.0458 | 0.0417 |
| Diametro | 12 | 14 |
| Transitività | 0.5146 | 0.5209 |
| Reciprocità | 0.6029 | 0.6100 |
| Assortatività | 0.8596 | 0.9759 |

Tab. 3.11: Statistiche descrittive a livello di rete

Un altro aspetto che conferma la poca connettività della rete è il valore riportato dal diametro, infatti la distanza del più lungo *shortest path* risulta pari a 12 nella rete originale e 14 in quella ristretta, ovvero quasi il doppio rispetto alla prima rete analizzata. La presenza di una connettività limitata si poteva intuire ben prima dell'osservazione degli indici descrittivi, in quanto in primo luogo è stata creata una rete di dimensioni maggiori rispetto alle precedenti, e in secondo luogo sono stati presi in considerazioni più nodi a priori che hanno generato tre *clusters* poco connessi tra loro. Per quanto riguarda la riduzione della connettività tra la rete originale e quella ristretta, è sicuramente dovuta al fatto che sono stati rimossi tre articoli che risultavano molto popolari, in quanto è stato fin dal principio un criterio di scelta per essi.
Dal valore assunto dall'indice di reciprocità si può desumere che il $60,29\%$ degli archi presenti nella rete hanno natura reciproca, e dunque la maggior parte degli articoli presenta un collegamento bidirezionale, per cui se un articolo è presente nei prodotti co-acquistati di un altro, allora è probabile anche il contrario. Un risultato interessante è dato dal valore assunto dal medesimo indice nella rete ristretta, che risulta maggiore rispetto a quello della rete originale. Questo risultato è stato causato probabilmente dalla discrepanza tra popolarità e socialità dei nodi posti a priori, in quanto risulta maggiore la quantità di archi diretti in entrata rispetto a quelli in uscita. L'indice di transitività, che indica la propensione della rete a formare triple transitive, è circa pari a 0.51, e sta ad indicare che il 51% di triple all'interno della rete risultano chiuse, mentre l'indice di assortatività risulta essere abbastanza elevato, e sta ad indicare la presenza di gruppi non banali, ovvero che i nodi presenti nella rete tendono a connettersi con nodi aveni caratteristiche simili. Anche per questi due indici i valori assunti dalla rete ristretta risultano più elevati, soprattutto per quanto riguarda l'assortatività.
Per cercare di comprendere al meglio le caratteristiche che possono portare al collegamento tra due nodi, viene riproposta nella Figura 3.21 la rete comprensiva di nodi a priori e suddivisa per categoria di articolo, ovvero in base alla variabile

*min_category*, e nella Figura 3.22 la rete originale suddivisa per formato dei prodotti, ovvero fisico o fisico e *streaming*.

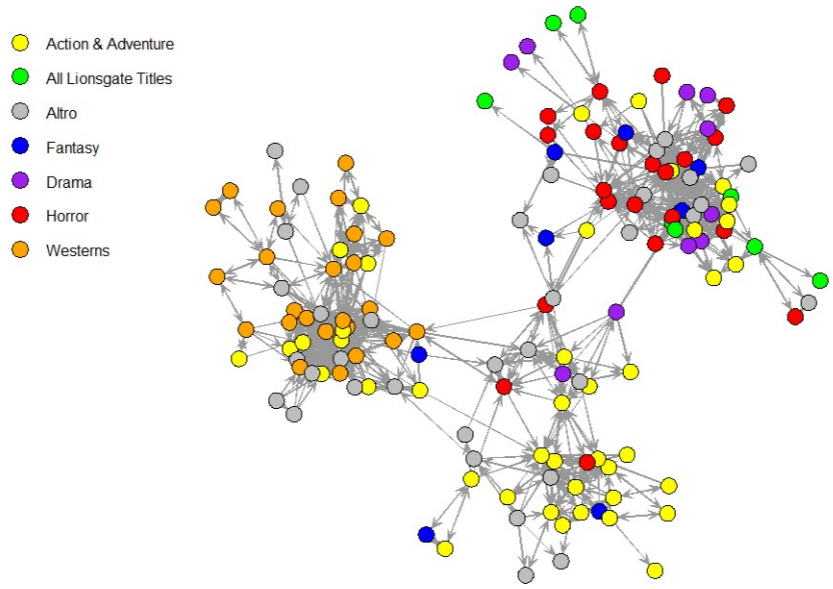


Fig. 3.21: Rete per categoria degli articoli.

Per quanto riguarda il primo grafico è necessario specificare che nella rete sono presenti 20 categorie differenti, ma per facilitarne la visualizzazione si è deciso di creare una nuova variabile uguale alla precedente, spostando in "Altro" tutte quelle categorie con una frequenza minore di 5. Fatta questa premessa nella rete si sono ottenute 7 differenti categorie, tra le più frequenti *"Action & Adventure"* con 42 nodi, *"Western"* con 24 nodi, e *"Horror"* con 21 nodi, che corrispondono anche ai generi degli articoli scelti a priori. Si può osservare che i film di tipo *"Western"* sono completamente racchiusi in un unico *cluster*, ovvero quello dell'articolo dello stesso genere scelto a priori. Per quanto riguarda il *cluster* del film horror *"Yeti: Maneater Series"*, si può notare come raggruppi quasi tutti i prodotti dello stesso genere, ad eccezioni di tre nodi, e come contenga tutti gli articoli di tipo *"All Lionsgate Titles"*, che corrispondono ai titoli di una compagnia di intrattenimento statunitense. Il terzo *cluster* ha un comportamento leggermente diverso dai precedenti, perché sebbene contenga come gli altri gruppi una componente elevata di articoli della stessa categoria di quello posto a priori, possiamo notare che effettivamente funge da collegamento tra gli altri due. Infatti gli articoli di genere *"Action & Adventure"* non tendono a rimanere in un unico gruppo ma bensì entrano con una quota rilevante anche nei due *cluster* non di riferimento, e questo può condurre a pensare che

gli utenti che prediligono generi *"Western"* e *"Horror"* siano abbastanza propensi anche al genere *"Action & Adventure"*, che quindi funge da collegamento tra i due. Mentre per quanto riguarda il secondo grafico in Figura 3.22, si può immediatamente notare come gli articoli con formato "Amazon Video - DVD", ovvero con formato misto, siano quasi unicamente concentrati nel *cluster* del genere *"Horror"*, sebbene l'articolo posto a priori sia unicamente di tipo fisico. Questa concentrazione di articoli con doppio formato potrebbe essere causata da un fattore temporale, e dunque dalla presenza di prodotti più recenti rispetto agli altri due *clusters*. Sarebbe a tal punto interessante, in una futura analisi, introdurre nel dataset una variabile indicante l'anno di uscita dell'articolo, anche se potrebbe essere complicata da interpretare per i prodotti contenti più film.

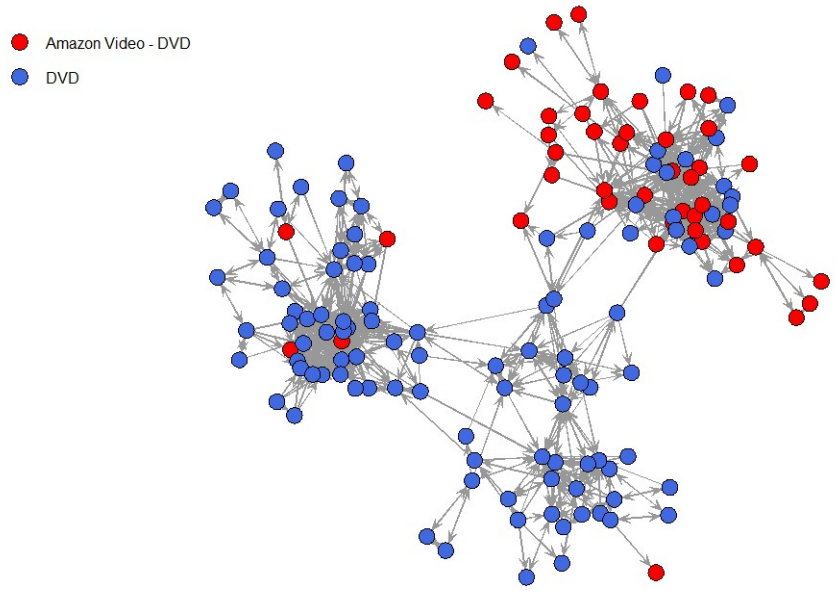


Fig. 3.22: Rete per formato degli articoli.

Per quanto riguarda le statistiche a livello di nodo, sono riportati nella Figura 3.23 tre tipologie di reti che si differenziano per la grandezza dei nodi, i quali sono proporzionali al grado del nodo ($a$), al livello di *betweenness* ($b$) e al livello di *closeness* ($c$). Dai tre grafici, oltre alla grandezza dei nodi, si può visualizzare l'articolo con indice di centralità maggiore, essendo posto con una colorazione differente.

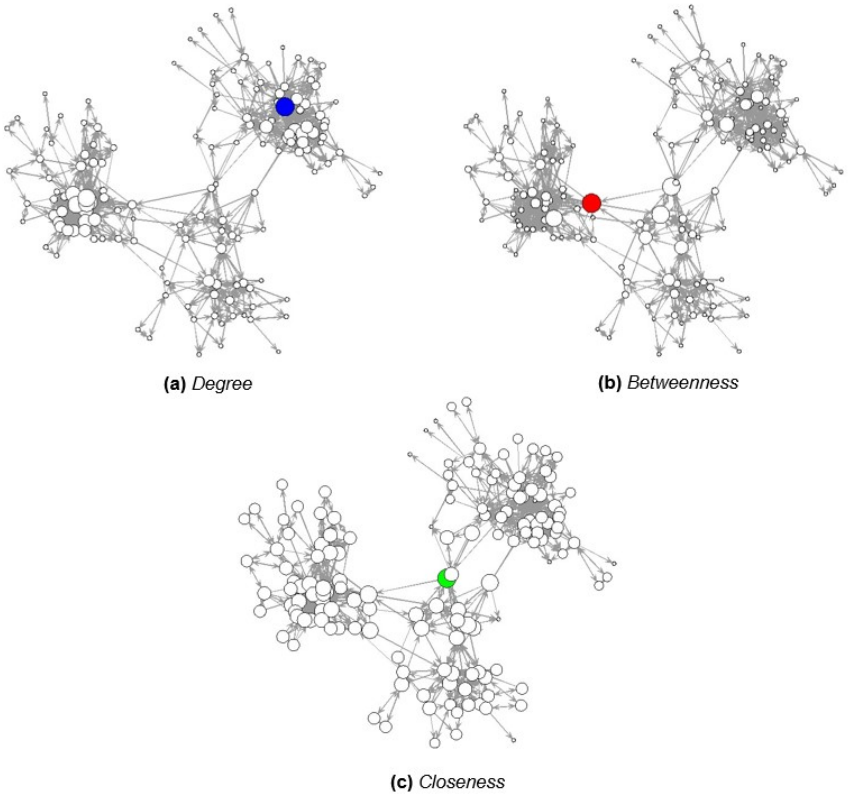


Fig. 3.23: Confronto tra indici di centralità.

Nella prima rete riferita al grado ($a$), il nodo con dimensione maggiore corrisponde al film *"Yeti: Maneater Series"*, ovvero come nelle altre reti ad uno degli articoli scelti a priori. Invece il nodo con maggior *betweenness*, visibile nel grafico ($b$), riporta un valore molto elevato e pari a $4566,693$, e corrisponde al prodotto *"Western Outlaws - 50 Movie MegaPack Digital"*. Si ricorda che per *betweenness* si intende la media del numero di volte in cui un nodo si trova nel percorso più breve tra altri due, e dunque questo valore elevato è giustificato dal fatto che i tre clusters raffigurati sono poco collegati tra loro, e l'articolo di riferimento si presenta come uno dei nodi più utili nel connettere assieme la rete. Per quanto riguarda il terzo e ultimo indice di centralità, ovvero la *closeness* ($c$), il nodo che assume valore maggiore, ovvero pari a 0.3, corrisponde all'articolo *"Best of the Worst - 12 Horror Movie Collection"*. Anche in questo caso ricordando che la *closeness* fornisce la velocità di propagazione dell'informazione da un singolo nodo, è ragionevole pensare che l'articolo che possiede un valore maggiore di questo indice, sia proprio uno di quelli che lega assieme i tre differenti *clusters*.

Nel complesso si può affermare che le tre reti analizzate dipendono molto dai nodi scelti a priori e dalle loro caratteristiche, infatti come nel primo caso,

scegliendo due articoli dello stesso genere, la rete riesce ad essere più unita e la densità tende ad aumentare. D'altro canto, optando per nodi con generi differenti, la rete risultante ha la tendenza nel formare *clusters* ben definiti, nei quali i nodi hanno caratteristiche simili ha quelli scelti a priori.
Infine, per quanto riguarda le statistiche descrittive, si può sicuramente osservare che la densità risulta essere inversamente proporzionale al numero di nodi presenti nella rete, e in aggiunta tende ad aumentare se gli articoli presentano caratteristiche simili; discorso inverso vale per il diametro, che essendo dato dal più lungo percorso più breve, tende ad aumentare al diminuire della densità e all'aumento dei nodi. Per quanto riguarda gli indici di reciprocità e transitività, risultano avere valori simili ed abbastanza elevati in tutte e tre le reti, e stanno ad indicare rispettivamente che la maggior parte degli archi sono di natura reciproca, e la tendenza dei nodi a formare triple chiuse; l'indice si assortatività invece assume valori leggermente più elevati nell'ultima rete, ovvero quella con tre nodi a priori, ma nel complesso sta ad indicare la presenza di gruppi non banali in tutte e tre le reti.

## 3.5 Applicazione dei modelli per dati di rete

Prima di stimare i modelli illustrati nella Sezione 2.4 (ANOVA, SRM, SRRM e AME), si sono create due matrici contenenti reciprocamente tutte le variabili nodali definite precedentemente, $X_n$, e tutte le variabili diadiche d'interesse, $X_{dyad}$. Queste ultime si riferiscono al tipo di formato (variabile *style*) e alle categoria di appartenenza dell'articolo (variabile *min_category* e *max_category*), e sono definite come segue:

$$x_{i,j}^{dyad} = \begin{cases} 1 & \text{se } \{i,j\} \text{ appartengono alla stessa categoria} \\ 0 & \text{altrimenti} \end{cases}$$

Si è dunque deciso, per maggiore completezza, di applicare i modelli alla rete più grande, ovvero quella definita con tre nodi a priori, partendo dal modello più semplice ed aumentando gradualmente la complessità .
Il primo modello proposto è un modello ANOVA, nel quale la variabile risposta, ovvero il livello di associazione tra due articoli, viene spiegato dalla media generale della matrice di adiacenza, e da due effetti di riga e di colonna, che si sintetizzano in due fattori che riguardano l'articolo $i$ come mittente e l'articolo $j$ come ricevente. Nel modello ogni Asin viene riportato due volte, in quanto nel primo caso sta a descrivere l'effetto di riga e nel secondo quello di colonna; I risultati ottenuti dal modello ANOVA sono i seguenti:

| | **Df** | **Sum Sq** | **Mean Sq** | **F value** | **Pr(>F)** |
|---|---|---|---|---|---|
| Row_asin | 146 | 21.77 | 0.15 | 3.69 | 0.000 |
| Col_asin | 146 | 54.98 | 0.38 | 9.33 | 0.000 |
| Residuals | 21316 | 860.63 | 0.04 | | |

Tab. 3.12: Risultati del modello ANOVA.

Dai risultati esposti nella Tabella 3.12, si può affermare la presenza di una forte eterogeneità degli articoli sia come "mittenti" che come "riceventi", in quanto i coefficienti riferiti agli effetti di riga e di colonna risultano in entrambi i casi significativi. Per ottenere le stime degli effetti di riga e di colonna del modello ANOVA bisogna seguire alcuni semplici passi, innanzitutto si deve calcolare la media generale $\hat{\mu}$, e le medie di riga e di colonna della matrice di adiacenza; successivamente per ottenere i vettori $\hat{a}$ e $\hat{b}$, ovvero gli effetti di riga e di colonna, è necessario sottrarre rispettivamente alle medie di riga e colonna quella generale. In questo modo ordinando in modo decrescente i vettori $\hat{a}$ e $\hat{b}$, riferiti agli effetti, si può capire quali siano i nodi con valori di socialità o popolarità più elevati.
L'ANOVA resta comunque un modello molto limitato per l'analisi dei dati di rete, in quanto in primo luogo presuppone che i vettori $\hat{a}$ e $\hat{b}$ siano indipendenti, e quindi non considera che ogni nodo è contemporaneamente mittente e ricevente, e in secondo luogo ignora il fatto che ogni coppia di nodi condivide due osservazioni.

Per proseguire l'analisi si sceglie di considerare il modello SRM (*Social Relations Model*), che supera i limiti precedentemente esposti. Rispetto all'ANOVA, questo modello fornisce diverse stime aggiuntive, infatti nello specifico inserisce 3 elementi della matrice di varianze e covarianze tra gli effetti di riga e colonna, la varianza dell'errore e $\rho$. Come si può vedere nella Tabella 3.13, che riporta i parametri stimati del SRM, l'unico parametro di regressione presente nel modello è la media $\mu$, che assume un valore pari a $-2.009$, e sta ad indicare la media a posteriori tra tutte le simulazioni del parametro; le altre colonne indicano in ordine di lettura la deviazione standard a posteriori (*psd*), il rapporto tra la media a posteriori e la deviazione standard (*z-stat*), e infine il livello di significatività osservato (*p-val*). Successivamente nella tabella vengono riportate anche le stime dei parametri relativi alla matrice di varianze e covarianze, ovvero nell'ordine $\sigma_a^2$, $\sigma_{ab}$, $\sigma_b^2$, $\rho$ e $\sigma_e$.

| | **pmean** | **psd** | **z-stat** | **p-val** |
|---|---|---|---|---|
| intercept | -2.009 | 0.067 | -29.821 | 0.000 |

| | **pmean** | **psd** |
|---|---|---|
| $\sigma_a^2$ | 0.13 | 0.02 |
| $\sigma_{ab}$ | 0.02 | |
| $\sigma_b^2$ | 0.26 | 0.04 |
| $\rho$ | 0.99 | 0.01 |
| $\sigma_\epsilon$ | 1.00 | 0.00 |

Tab. 3.13: Stima dei parametri del modello SRM.

Gli istogrammi riportati nella Figura 3.24 consentono di fare una valutazione sulla bontà di adattamento del modello, in quanto ripropongono una stima della distribuzione a posteriori di alcune quantità. Nello specifico i grafici in alto a sinistra e a destra rappresentano rispettivamente l'andamento iterato della stima dei parametri di covarianza e di regressione (in questo caso solo l'intercetta), mentre i 5 istogrammi che vengono proposti in seguito si riferiscono in ordine alla deviazione standard delle medie di riga $\hat{\sigma_a}$ (*sd.rowmeans*), alla deviazione standard delle medie di colonna $\hat{\sigma_b}$ (*sd.colmeans*), alla dipendenza diadica $\hat{\rho}$ (*dyad.dep*), e alla dipendenza triadica (*cycle.dep* e *trans.dep*).

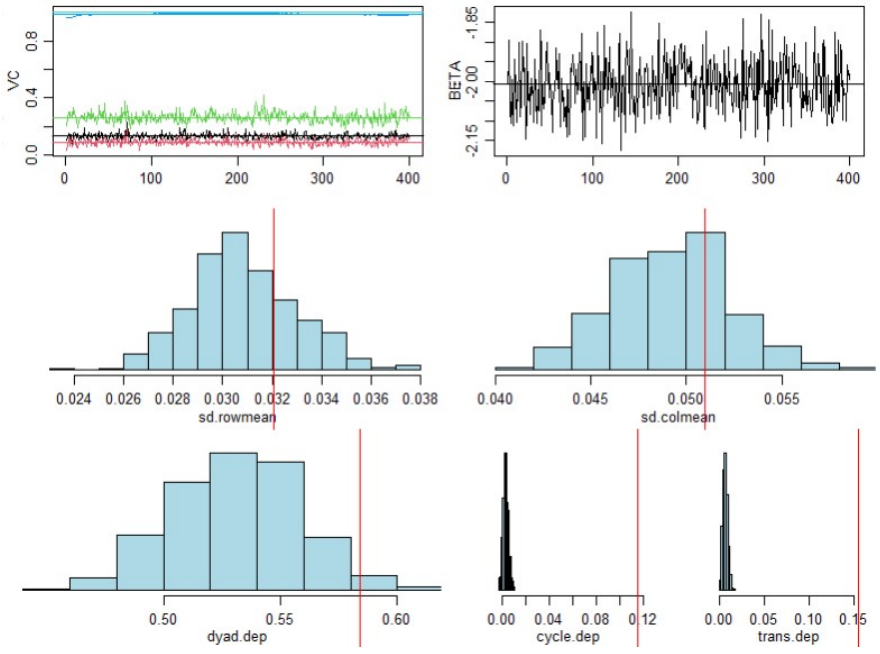


Fig. 3.24: Bontà di adattamento del modello SRM.

La linea rossa presente in tutti gli istogrammi rappresenta il valore osservato della statistica di riferimento, e dunque, valori lontani dalla distribuzione a posteriori stanno ad indicare uno scarso adattamento del modello ai dati. Nel grafico riproposto possiamo quindi osservare un buon adattamento in termini di deviazione standard delle medie di riga e di colonna, in quanto la linea rossa che sta ad

indicare la statistica osservata è centrale alla distribuzione a posteriori. In aggiunta il modello riesce a cogliere la dipendenza diadica, anche se in maniera non ottimale, mentre è presente una forte discrepanza per quanto riguarda la dipendenza triadica, ma questo fatto era prevedibile in quanto il modello SRM è costruito in modo tale da non poterla cogliere.

Il successivo modello che si va a considerare è il *Social Relations Regression Model* (SRRM), che in aggiunta all'SRM permette di tenere in considerazione le variabili presenti nel dataset, con lo scopo di aggiungere informazione e riuscire a spiegare meglio la variabilità dei dati. Per selezionare le variabili diadiche o di nodo da inserire nel modello sono state effettuate alcune prove, partendo da un modello con una sola variabile e aggiungendone via via di ulteriori. Per quanto riguarda le variabili di nodo, è giusto sottolineare che si è tentato di inserirle congiuntamente come effetti di riga e di colonna, in quanto non si posseggono informazioni che portano a valutarne singolarmente gli effetti. Nella Tabella 3.14 vengono riportate le stime del miglior modello ottenuto, ovvero quello contenente 3 variabili diadiche: *max.category, min.category* e *style*. Nel modello proposto tutte le variabili risultano significative, e i valori dei parametri assunti dalle variabili diadiche riferite alla categoria, ovvero *max.category.dyad* e *min.category.dyad* sono rispettivamente pari a 0.148 e 0.394, e stanno ad indicare che i film appartenenti alla stessa categoria hanno probabilità maggiore di essere acquistati insieme. La stima del parametro *style.dyad*, che si riferisce al formato dell'articolo, è pari a 0.502, e con lo stesso ragionamento precedentemente riportato, si può affermare che i film aventi lo stesso supporto hanno probabilità maggiore di essere acquistati congiuntamente.

| | **pmean** | **psd** | **z-stat** | **p-val** |
|---|---|---|---|---|
| intercept | -2.420 | 0.083 | -29.287 | 0.000 |
| max.category.dyad | 0.148 | 0.052 | 2.854 | 0.004 |
| min.category.dyad | 0.393 | 0.049 | 8.061 | 0.000 |
| style.dyad | 0.502 | 0.056 | 8.887 | 0.000 |

| | **pmean** | **psd** |
|---|---|---|
| $\sigma_a^2$ | 0.14 | 0.02 |
| $\sigma_{ab}$ | 0.10 | 0.02 |
| $\sigma_b^2$ | 0.28 | 0.04 |
| $\rho$ | 0.99 | 0.01 |
| $\sigma_\epsilon^2$ | 1.00 | 0.00 |

Tab. 3.14: Stima dei parametri del modello SRRM.

Gli istogrammi riportati nella Figura 3.25 permettono di fare una valutazione sulla bontà di adattamento del modello. Nello specifico i grafici risultano molto simili all'adattamento ottenuto col SRM, con un lieve miglioramento nel catturare la dipendenza diadica. Per quanto riguarda la dipendenza triadica, anche in questo caso non è possibile coglierla per i limiti imposti dal modello.

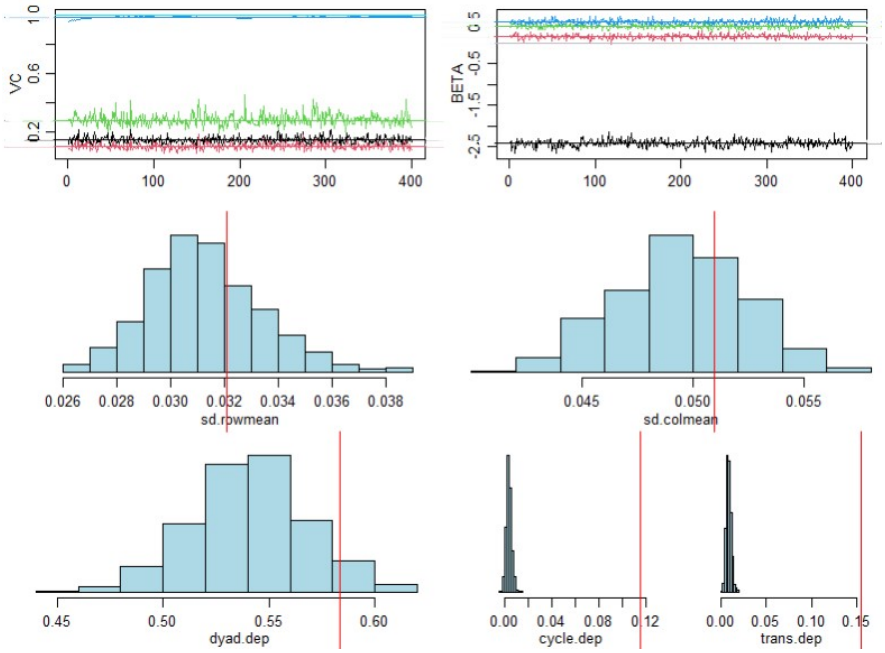


Fig. 3.25: Bontà di adattamento del modello SRRM.

Con l'intento di migliorare il modello, e di cogliere la dipendenza triadica, è stato implementato un modello AME (*Additive and Moltiplicative Effects model*), che contempla l'inserimento di fattori latenti tramite un'interazione moltiplicativa di caratteristiche di nodo non osservate. Rispetto al precedente modello, oltre a dover selezionare le variabili da inserire, è necessario determinare la dimensione dei fattori latenti, $u_i$ e $v_i$, ovvero il rango $r$ della matrice $UV^T$. Come operato precedentemente si sono dunque stimati vari modelli, e si è scelto quello che garantiva un'adattabilità migliore. Nella Tabella 3.15 sono riportate le stime del miglior modello ottenuto, ovvero un AME con 3 variabili diadiche e un fattore latente. Innanzitutto dalla tabella si può osservare che tutte le variabili risultano significative, e spostando l'attenzione sulle stime dei parametri delle variabili, si osserva che la variabile diadica *max.category* presenta un valore pari a 0.154, mentre la variabile diadica *min.category* pari a 0.377, a conferma del fatto che c'è una tendenza nel

co-acquistare articoli appartenenti alla stessa categoria; discorso analogo vale per la variabile diadica *style*, che presenta un valore pari a 0.342, che anche in questo caso sta ad indicare la propensione nell'acquistare congiuntamente articoli con il medesimo formato.

| | **pmean** | **psd** | **z-stat** | **p-val** |
|---|---|---|---|---|
| intercept | -3.131 | 0.129 | -24.328 | 0.000 |
| max.category.dyad | 0.154 | 0.070 | 2.209 | 0.027 |
| min.category.dyad | 0.377 | 0.070 | 5.359 | 0.000 |
| style.dyad | 0.342 | 0.067 | 5.074 | 0.000 |

| | **pmean** | **psd** |
|---|---|---|
| $\sigma_a^2$ | 0.28 | 0.06 |
| $\sigma_{ab}$ | 0.15 | 0.05 |
| $\sigma_b^2$ | 0.24 | 0.05 |
| $\rho$ | 0.98 | 0.02 |
| $\sigma_\epsilon^2$ | 1.00 | 0.00 |

Tab. 3.15: Stima dei parametri del modello AME.

I grafici riportati nella Figura 3.26 consentono, come nei precedenti casi, di fare una valutazione sulla bontà di adattamento del modello, in quanto ripropongono una stima della distribuzione a posteriori di alcune quantità. Rispetto al modello precedentemente esposto, ovvero l'SRRM, si possono notare vari cambiamenti nelle distribuzioni a posteriori, fatta eccezione per la dipendenza diadica che presenta un grafico molto simile al precedete, e che dunque sta ad indicare che il modello riesce a coglierla anche se con una distribuzione non troppo centrata. L'adattamento del modello è invece leggermente peggiorato per quanto riguarda la deviazione standard delle medie di colonna, nella quale la distribuzione a posteriori risulta meno centrata rispetto alla statistica osservata, mentre per quanto riguarda la deviazione standard delle medie di riga si può notare un netto peggioramento, con la statistica osservata situata ai limiti della coda sinistra della distribuzione. Infine per quanto riguarda l'istogramma relativo alla dipendenza triadica, si può osservare che essa viene colta solo in parte, infatti la statistica osservata non è centrata nella distribuzione a posteriori.

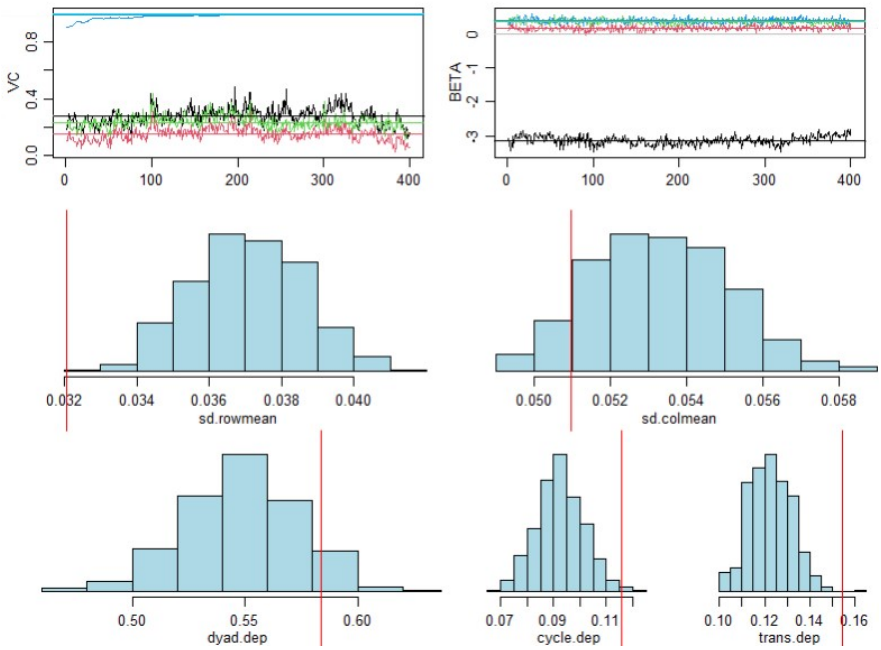


Fig. 3.26: Bontà di adattamento del modello AME.

Nel complesso i modelli che si adattano meglio ai dati sono l'SRRM e l'AME, il primo infatti riesce a cogliere la variabilità delle medie di riga, la variabilità delle medie di colonna, e la dipendenza diadica, mentre il secondo riesce in parte a cogliere la dipendenza triadica a discapito di un peggioramento nella distribuzione a posteriori delle statistiche riferite alla devianza degli effetti di riga e di colonna. Le variabili diadiche risultano in entrambi i casi significative, e sia nel primo che nel secondo modello stanno ad indicare la propensione degli utenti nell'acquistare prodotti della medesima categoria, sia essa più ampia o specifica, e nel co-acquistare articoli con lo stesso formato (fisico o *streaming*).

# Conclusioni

L'obiettivo di questa tesi è stato in primo luogo quello di comprendere il funzionamento delle raccomandazioni personalizzate di Amazon, cercando di capire quali variabili potessero influenzare maggiormente l'acquisto di due o più articoli, e in secondo luogo quello di individuare le relazioni più interessanti in termini di utilità commerciale e capacità previsiva. Inizialmente nei primi due capitoli sono stati introdotti i concetti fondamentali per lo svolgimento dell'analisi, ovvero quelli relativi ai modelli per dati di rete e quelli attinente alle regole associative, mentre nel terzo capitolo, si sono applicate ai dati Amazon le metodologie precedentemente esposte. Nei due dataset su cui è stata svolta l'analisi si sono ritenute necessarie alcune operazioni di ristrutturazione, nelle quali sono state eliminate le variabili che non risultavano di interesse, e ne sono state modificate e create delle altre. In seguito, dopo la pulizia dei dati, sono state effettuate alcune analisi descrittive per comprendere al meglio il dataset, e si sono notati principalmente due aspetti interessanti, il primo che riguarda la valutazione delle recensioni e il secondo che riguarda il formato degli articoli. Nello specifico la distribuzione dei voti lasciati dagli utenti risulta fortemente asimmetrica, con l'80% di essi che vota almeno 4 stelle, mentre per quanto riguarda il tipo di formato degli articoli, si è notata la presenza quasi totale di recensioni di prodotti in formato fisico per ogni recensione degli stessi in formato *streaming*, e in aggiunta si è riscontrata una componente di recensioni in formato *streaming* molto più elevata della corrispettiva, che porta a pensare che gli utenti siano più favorevoli nel recensire articoli nel primo formato. Inoltre è stata effettuata una piccola analisi sulla variabile "*price*", riscontrando la presenza di diversi picchi nella sua distribuzione di frequenza, che si possono ricondurre alla tecnica di *marketing* chiamata *left digit effect* o *charm price*, che consiste nell'imporre un prezzo di finta convenienza. Questa ultima variabile, sebbene molto interessante e informativa, non è stata utilizzata nelle analisi svolte, in quanto comprendeva un'elevata quantità di valori mancanti.

Successivamente si è deciso di applicare le regole associative, con la finalità di scovare le associazioni più interessanti che collegano due o più articoli. Per rendere

possibile l'utilizzo di questa metodologia è stato necessario eseguire una selezione dei dati, utilizzando come criteri di scelta la data in cui è stata scritta la recensione, e il numero di recensioni effettuate per cliente, ritenendo gli utenti con un numero di acquisti maggiore più interessanti. Nello specifico sono state selezionate le recensioni dal 03-10-2012 al 03-10-2018, ovvero quelle relative agli ultimi 6 anni disponibili, e gli utenti che in questo periodo di tempo hanno effettuato almeno 50 ordini. Utilizzando l'algoritmo a priori sono state estratte 176 regole, che sono state poi raffigurate e raggruppate per i valori più interessanti. Nei grafici proposti si può visualizzare la distribuzione delle regole in base a supporto, fiducia e *lift*, e si può notare che la maggior parte di esse è costituita da 3 tre articoli. Successivamente sono state analizzate le regole più interessanti suddividendole in vari gruppi, utilizzando i valori più elevati di ciascuna misura per definirli. La regola giudicata più interessante, per i valori di fiducia supporto e *lift*, comprende come articoli antecedenti i film *"Elysium"* e *"Captain Phillips Steelbook"* e come conseguente il film *"Gravity 2013"*. Questi articoli si trovano nella stessa transazione circa l'1.7% delle volte, e la probabilità di acquistare il conseguente dato l'antecedente è circa pari all'88%. Nell'analisi dei gruppi si è notata la partecipazione di alcuni articoli in più di una regola, e si è quindi deciso di proporre alcuni grafici che riuscissero ad evidenziarlo. Per quanto riguarda la seconda parte dell'analisi è stato necessario un ridimensionamento del dataset, in quanto nella sua interezza comprende circa 180.000 articoli differenti, che ne rendono impossibile la creazione di una rete generale, sia per difficoltà nelle rappresentazioni grafiche ma soprattutto per l'inevitabile costo computazionale. A questo scopo si sono selezionati gli articoli con un numero di prodotti nella variabile *Also_buy* non troppo elevato ma nemmeno troppo esiguo, per garantire un buon numero di associazioni che consentissero lo svolgimento dell'analisi, nello specifico si è deciso di impostare un valore minimo di 5 e un valore massimo di 20. In un secondo momento sono stati scelti gli articoli a priori, e sono stati considerati esclusivamente i prodotti a cui ne sono raccomandati almeno uno di essi. In particolare sono state create 3 differenti reti, utilizzando come principali fattori distintivi il numero di articoli posti a priori e la categoria in cui è inserito l'articolo. In seguito sono state fornite alcune rappresentazioni grafiche delle reti, e sono state svolte le prime analisi descrittive a livello di nodo e a livello di rete, in modo tale da comprenderne le principali caratteristiche e gli aspetti più interessanti. Per tutte le reti si è riportata la rappresentazione con i nodi posti priori e quella per il numero di archi diretti in uscita, in modo tale da comprendere la struttura della rete e capire quali fossero i nodi con più connessioni. Successivamente sono state svolte le analisi descrittive a livello di rete e a livello di nodo, le quali sono state in seguito messe

a confronto per le tipologie di reti create, in modo tale da evidenziare le principali differenze o identificare i fattori comuni. Le reti nel complesso presentano una densità abbastanza buona, con un valore minimo riscontrato nella terza rete di circa 0.05. Come ci si poteva aspettare questo valore risulta inversamente proporzionale al numero di nodi presenti, che sono direttamente collegati al valore del diametro. Dunque più nodi sono presenti, più la densità tende a diminuire, e più il valore del diametro aumenta. Gli indici di transitività e di reciprocità, ovvero la propensione a formare triplette di nodi e la proporzione di nodi reciproci, risultano abbastanza elevati in tutte e tre le reti, con un valore di reciprocità leggermente più elevato di quello di transitività. L'indice di assortatività, ovvero la tendenza dei nodi a connettersi con altri nodi che presentano caratteristiche simili, risulta avere un valore maggiore nella terza rete, ovvero quella più estesa, soprattutto nel caso in cui vengano tolti i 3 nodi posti a priori. Nel complesso sta comunque ad indicare la presenza di gruppi non banali in tutte e 3 le reti. In seguito, si è proposta un'ulteriore rappresentazione grafica della rete, suddividendo i nodi per categoria minima di appartenenza, e si è potuto notare in tutte e tre le reti come la maggior parte di essi tendano a collegarsi con nodi simili, ovvero con nodi appartenenti alla medesima categoria. Per quanto riguarda la terza rete, si è riportata una rappresentazione grafica aggiuntiva per il formato degli articoli, e si è potuta notare la presenza di un *cluster* contenente quasi unicamente articoli con lo stesso formato. Nel complesso si può affermare che le reti analizzate dipendano in maniera rilevante dai nodi posti a priori e dalle loro caratteristiche, infatti scegliendo due articoli dello stesso genere la rete riesce ad essere più unita e la densità tende ad aumentare. D'altro canto, optando per nodi con generi differenti, la rete presenta la tendenza nel formare *clusters* ben definiti, nei quali i nodi hanno nella maggior parte dei casi caratteristiche simili a quelli scelti a priori.

Nella sezione successiva si sono applicati alla terza rete i modelli descritti nel Capitolo 2, partendo dal modello più semplice ed aumentando gradualmente la complessità. Si è deciso di scegliere la terza rete in parte perché dall'analisi descrittiva risultava essere la più interessante, e in parte perché si mostrava come quella con il maggior numero di nodi. Nel complesso i modelli che si adattano meglio ai dati sono i due più complessi, ovvero l'SRRM e l'AME, il primo infatti riesce a cogliere la variabilità delle medie di riga, la variabilità delle medie di colonna e la dipendenza diadica, mentre il secondo riesce in parte a cogliere la dipendenza triadica a discapito di un peggioramento nella distribuzione a posteriori delle statistiche riferite alla devianza degli effetti di riga e colonna. Coerentemente con quanto visto precedentemente la variabile indicante la categoria minima risulta essere significativa, e

suggerisce che gli utenti tendono ad acquistare congiuntamente articoli appartenenti al medesimo genere. Lo stesso ragionamento può essere riproposto per le variabili indicanti la categoria massima di appartenenza e il formato dei prodotti, infatti entrambe risultano essere significative e stanno ad indicare una propensione nel stabilire connessioni tra articoli con la medesima caratteristica. Per analisi successive sarebbe interessante introdurre nel dataset altre variabili, come per esempio l'anno di pubblicazione del film o il paese di produzione, in modo tale da aggiungere informazione e comprendere ulteriori strutture di dipendenza.

L'utilizzo delle regole associative e dei modelli per dati di rete ha evidenziato alcuni lati in comune nelle due metodologie, ma allo stesso tempo anche qualche differenza. In primo luogo in entrambi gli approcci si può notare come alcuni articoli rivestano un ruolo più importante e centrale di altri, infatti nelle regole associative, dopo aver ridimensionato i dati, viene considerato solamente un numero esiguo di articoli, a causa anche dell'impostazione di un valore minimo di fiducia e supporto, e tra questi articoli se ne possono notare alcuni presenti in molteplici regole. Allo stesso tempo nelle reti si possono individuare, tramite gli indici a livello di nodo, alcuni prodotti con un numero di connessioni in entrata e in uscita più elevate di altri, i quali rivestono di conseguenza un ruolo più centrale. Avendo però utilizzato diversi criteri nel ridimensionamento del dataset per le rispettive analisi, risulta impossibile il confronto degli articoli. Questi due metodi, sebbene possano portare ad alcune conclusioni simili, presentano anche alcune importanti differenze. L'analisi per dati di rete riesce infatti ad indicare alcuni risultati molto interessanti, come la propensione nel stabilire connessioni tra articoli con caratteristiche simili, ovvero per medesima categoria o formato, mentre d'altra parte le regole associative non riescono ad includere le informazioni presenti nei nodi, e di conseguenza non possono cogliere queste strutture di dipendenza.

Da una parte l'approccio più tradizionale delle regole associative si basa sulla probabilità di un *itemset* conseguente dato un corrispettivo antecedente, e dunque considera solo le transazioni passate con lo scopo individuare le regole più interessanti. L'approccio dell'analisi per dati di rete può dare invece una visione più ampia del contesto, non limitandosi semplicemente ai nodi e ai loro collegamenti. Tramite questa metodologia è infatti possibile aggiungere informazione al problema studiando nel dettaglio le caratteristiche di ogni nodo, e al contempo applicare dei modelli che riescano a quantificare il livello di influenza delle variabili d'interesse sulla base delle relazioni presenti nella rete.

L'analisi per dati di rete può quindi generare un numero rilevante di informazioni, e allo stesso tempo ha il vantaggio di poter essere applicata in svariati ambiti.

# Bibliografia


[1] James H. Albert and Siddhartha Chib. Bayesian analysis of binary and polychotomous response data. *Journal of the American Statistical Association*, 88(422):669–679, 1993.

[2] Adelchi Azzalini and Bruno Scarpa. *Analisi dei dati e data mining*. 2009.

[3] Albert-László Barabási, Réka Albert, and Hawoong Jeong. Mean-field theory for scale-free random networks. *Physica A*, 272(1):173–187, 1999.

[4] Roberto J. Bayardo Jr, Rakesh Agrawal, and Dimitrios Gunopulos. Constraint-based rule mining in large, dense databases. *Data mining and knowledge discovery*, 4(2-3):217–240, 2000.

[5] P. J. Bickel and Y. Ritov. *Local Asymptotic Normality of Ranks and Covariates in Transformation Models*, pages 43–54. Springer New York, New York, NY, 1997.

[6] Ronald S. Burt. *Toward a Structural Theory of Action*. 1982.

[7] Patrick Doreian. *Advances in network clustering and blockmodeling*. Wiley Series in Computational and Quantitative Social Science. Wiley, Hoboken, NJ, 1st edition edition, 2020.

[8] Sergei N Dorogovtsev. *Evolution of networks : from biological nets to the Internet and WWW / S. N. Dorogovtsev and J. F. F. Mendes*. Oxford University press, Oxford, 2003.

[9] L.C. Freeman, D.R. White, and A.K. Romney. *Research Methods in Social Network Analysis*. 1989.

[10] Michael Hahsler and Sudheer Chelluboina. Visualizing association rules : Introduction to the r-extension package arulesviz. 2011.

[11] Peter Hoff. Additive and Multiplicative Effects Network Models. *Statistical Science*, 36(1):34 – 50, 2021.

[12] Peter D Hoff. Dyadic data analysis with amen. *arXiv.org*, 2015.

[13] Peter D Hoff, Adrian E Raftery, and Mark S Handcock. Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97(460):1090–1098, 2002.

[14] Stanley Milgram. The Small-World Problem. pages 61–67, 1967.

[15] Shahryar Minhas, Peter D Hoff, and Michael D Ward. Inferential approaches for network analyses: Amen for latent factor models. *arXiv.org*, 2018.

[16] Mark E. J. Newman. *Networks : an introduction / M. E. J. Newman.* Oxford University Press, Oxford New York, 2010.

[17] Mark E. J. Newman. Networks / mark newman, 2018.

[18] Krzysztof Nowicki and Tom A. B Snijders. Estimation and prediction for stochastic blockstructures. 96(455):1077–1087, 2001.

[19] Gaurav Pandey, Sanjay Chawla, Simon Poon, Bavani Arunasalam, and Joseph G. Davis. Association rules network: Definition and applications: Association rules network. *Statistical analysis and data mining*, 1(4):260–279, 2009.

[20] Troy Raeder and Nitesh V. Chawla. Market basket analysis with networks. *Social network analysis and mining*, 1(2):97–113, 2011.

[21] John Scott. *Social Network Analysis.* 2nd edition, 2000.

[22] Kazuhiro Takemoto and Chikoo Oosawa. *Introduction to Complex Networks: Measures, Statistical Properties, and Models*, chapter 2, pages 45–75. John Wiley  Sons, Ltd, 2012.

[23] Rebecca Warner, David Kenny, and Michael Stoto. A new round robin analysis of variance for social interaction data. *Journal of Personality and Social Psychology*, 37:1742–1757, 1979.

[24] Stanley Wasserman and Katherine Faust. *Social Network Analysis: Methods and Applications.* Structural Analysis in the Social Sciences. Cambridge University Press, 1994.

[25] Duncan J. Watts. *Small worlds : the dynamics of networks between order and randomness / Duncan J. Watts*. Princeton studies in complexity. Princeton university press, Princeton, 1999.

[26] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. 393(6684):440–442, 1998.

# Capitolo A

# Codice R

## Caricamento librerie

```
library(tidyr)
library(dplyr)
library(plyr)
library(rjson)
library(jsonlite)
library(stringi)
library(arules)
library(lubridate)
library(lattice)
library(igraph)
library(amen)
library(stringr)
library(scales)
library(ggplot2)
```

## A.1 Ristrutturazione metadati

```
Meta_Movies=read.csv("Meta_Movies.csv", sep=",")

#Si elimina la variabile di conteggio
Meta_Movies = Meta_Movies %>% dplyr::select(-c(X))

#Si eliminano i duplicati
```

```
Meta_Movies=distinct(Meta_Movies)

#Si verifica la presenza di valori nulli
colSums(is.na(Meta_Movies))

NA_fun = function(Meta_Movies){
  n_NA = sapply(Meta_Movies, function(col) sum(is.na(col) | col=="" |
      col=="[]"))
  n_NA = sort(n_NA[n_NA > 0])
  n_NA = data.frame(
    variabile = names(n_NA),
    freq_assoluta = as.numeric(n_NA),
    freq_relativa = round(as.numeric(n_NA)/nrow(Meta_Movies), 4)
  )
  n_NA
}
#Tabella con frequenze assolute e relative di dati mancanti
n_NA = NA_fun(Meta_Movies)
n_NA

#Si eliminano variabili con troppi valori mancanti
Meta_Movies = Meta_Movies %>% dplyr::select(-c(fit, tech2, tech1,
    similar_item, date, feature, imageURL,imageURLHighRes))

#Si eliminano variabili poco utili allo scopo dell'analisi
Meta_Movies = Meta_Movies %>% dplyr::select(-c(description, main_cat,
    details, brand))

#Creazione variabile "num_cat"
Meta_Movies$num_cat=str_count(Meta_Movies$category, ',')
Meta_Movies$num_cat[Meta_Movies$num_cat=="0"]=""
Meta_Movies$num_cat=as.numeric(Meta_Movies$num_cat)
Meta_Movies$num_cat=Meta_Movies$num_cat+1

#Riorganizzazione variabile "rank" (si eliminano rank di altre categorie)
Meta_Movies=arrange(Meta_Movies,rank)
Meta_Movies$rank[1:79]="[]"
Meta_Movies$rank[Meta_Movies$rank=="[]"]="0"
Meta_Movies$rank=gsub('\\D+','', Meta_Movies$rank)
Meta_Movies$rank[Meta_Movies$rank=="0"]<-""
```

```
Meta_Movies$rank=as.numeric(Meta_Movies$rank)

#Creazione variabile "min_category"
Meta_Movies$min_category=Meta_Movies$category
Meta_Movies$min_category=gsub(".*,", "", Meta_Movies$min_category)
Meta_Movies$min_category=gsub(".*,", "", Meta_Movies$min_category)
Meta_Movies$min_category=gsub(".*,", "", Meta_Movies$min_category)
Meta_Movies$min_category=gsub(".*,", "", Meta_Movies$min_category)
Meta_Movies$min_category=gsub(".*,", "", Meta_Movies$min_category)
Meta_Movies$min_category = substring(Meta_Movies$min_category,1,
    nchar(Meta_Movies$min_category)-2)
Meta_Movies$min_category=gsub(".*'", "", Meta_Movies$min_category)

#Si imposta la soglia a 500
Meta_Movies$min_category=as.factor(Meta_Movies$min_category)
var_da_accorpare = Meta_Movies$min_category
sum(table(var_da_accorpare)>500)
var_da_accorpare = factor(var_da_accorpare, levels =
    c(levels(var_da_accorpare), "altro"))
#aggiungo la modalita' "altro" alla variabile
var_da_accorpare[var_da_accorpare %in%
    attr(table(var_da_accorpare)[table(var_da_accorpare)<=500],"dimnames")[[1]]]
    = "altro"
var_da_accorpare = factor(var_da_accorpare)
table(var_da_accorpare) # tabella di frequenza
Meta_Movies$min_category = var_da_accorpare

#Creazione variabile "max_category"
Meta_Movies$max_category=Meta_Movies$category
Meta_Movies$max_category= substring(Meta_Movies$max_category, 18)
Meta_Movies$max_category= substring(Meta_Movies$max_category,1,
    nchar(Meta_Movies$max_category)-1)
Meta_Movies$max_category=gsub("(.*),.*", "\\1", Meta_Movies$max_category)
Meta_Movies$max_category=gsub("(.*),.*", "\\1", Meta_Movies$max_category)
Meta_Movies$max_category=gsub("(.*),.*", "\\1", Meta_Movies$max_category)
Meta_Movies$max_category=gsub("(.*),.*", "\\1", Meta_Movies$max_category)
Meta_Movies$max_category=gsub("(.*),.*", "\\1", Meta_Movies$max_category)
Meta_Movies$max_category = substring(Meta_Movies$max_category,1,
    nchar(Meta_Movies$max_category)-1)
```

```
#Si imosta la soglia a 5.000
Meta_Movies$max_category=as.factor(Meta_Movies$max_category)
var_da_accorpare = Meta_Movies$max_category
sum(table(var_da_accorpare)>5000)
var_da_accorpare = factor(var_da_accorpare, levels =
    c(levels(var_da_accorpare), "altro"))
#aggiungo la modalita' "altro" alla variabile
var_da_accorpare[var_da_accorpare %in%
    attr(table(var_da_accorpare)[table(var_da_accorpare)<=5000],"dimnames")[[1]]]
    = "altro"
var_da_accorpare = factor(var_da_accorpare)
table(var_da_accorpare)
Meta_Movies$max_category = var_da_accorpare

Meta_Movies = Meta_Movies %>% dplyr::select(-c(category))

#Riorganizzazione variabile "price"
for (j in 1:length(Meta_Movies$price)){
  if (nchar(Meta_Movies$price[j])>10)
    Meta_Movies$price[j]=""
}
rm(j)
Meta_Movies$price = substring(Meta_Movies$price, 2)
Meta_Movies$price=as.numeric(Meta_Movies$price)

#Riorganizzazione variabile "also_buy"
Meta_Movies$also_buy=gsub("c","",as.character(Meta_Movies$also_buy))
Meta_Movies$also_buy=gsub("[[:punct:]]",
    "",as.character(Meta_Movies$also_buy))

#Riorganizzazione variabile "also_view"
Meta_Movies$also_view=gsub("c","",as.character(Meta_Movies$also_view))
Meta_Movies$also_view=gsub("[[:punct:]]",
    "",as.character(Meta_Movies$also_view))
```

## A.2 Ristrutturazione recensioni

```
Review_Movies_and_TV = read.csv("Movies.csv", sep=",")
```

```
#Si elimina la variabile di conteggio
Review_Movies_and_TV = Review_Movies_and_TV %>% dplyr::select(-c(X))

#Si eliminano i duplicati
Review_Movies_and_TV=distinct(Review_Movies_and_TV)

#Si verifica la presenza di valori nulli
colSums(is.na(Review_Movies_and_TV))

NA_fun = function(Review_Movies_and_TV){
  n_NA = sapply(Review_Movies_and_TV, function(col) sum(is.na(col) |
      col=="" | col=="[]"))
  n_NA = sort(n_NA[n_NA > 0])
  n_NA = data.frame(
    variabile = names(n_NA),
    freq_assoluta = as.numeric(n_NA),
    freq_relativa = round(as.numeric(n_NA)/nrow(Review_Movies_and_TV), 4)
  )
  n_NA
}

#Tabella con frequenze assolute e relative di dati mancanti
n_NA = NA_fun(Review_Movies_and_TV)
n_NA

#Si eliminano variabili poco utili o con troppi valori mancanti
Meta_Movies = Meta_Movies %>% dplyr::select(-c(vote, image, reviewText,
    summary, reviewerName))

#Si cambia il formato di "reviewTime"
Review_Movies_and_TV$reviewTime=mdy(Review_Movies_and_TV$reviewTime)

#Si eliminano prodotti senza metadati
Review_Movies_and_TV=subset(Review_Movies_and_TV,
    Review_Movies_and_TV$asin %in% Meta_Movies$asin)

#Riorganizzazione variabile "style"
Review_Movies_and_TV$style=as.character(Review_Movies_and_TV$style)
table(Review_Movies_and_TV$style)
```

```
Review_Movies_and_TV$style[Review_Movies_and_TV$style=="{'Format:': '
    Amazon Video'}"]="Amazon Video"
Review_Movies_and_TV$style[Review_Movies_and_TV$style=="{'Format:': '
    Prime Video'}"]="Amazon Video"
Review_Movies_and_TV$style[Review_Movies_and_TV$style=="{'Format:': '
    Amazon Instant Video'}"]="Amazon Video"
Review_Movies_and_TV$style[Review_Movies_and_TV$style=="{'Format:': '
    Kindle Edition'}"]="Amazon Video"
Review_Movies_and_TV$style[Review_Movies_and_TV$style=="{'Format:': '
    DVD'}"]="DVD"

#Si imposta la soglia a 700.000
Review_Movies_and_TV$style=as.factor(Review_Movies_and_TV$style)
var_da_accorpare = Review_Movies_and_TV$style
sum(table(var_da_accorpare)>700000)
var_da_accorpare = factor(var_da_accorpare, levels =
    c(levels(var_da_accorpare), "altro"))
var_da_accorpare[var_da_accorpare %in%
    attr(table(var_da_accorpare)[table(var_da_accorpare)<=700000],"dimnames")[[1]]]
    = "altro"
var_da_accorpare = factor(var_da_accorpare)
table(var_da_accorpare)
Review_Movies_and_TV$style= var_da_accorpare
Review_Movies_and_TV$style=as.character(Review_Movies_and_TV$style)
Review_Movies_and_TV$style[Review_Movies_and_TV$style=="altro"]="DVD"
Review_Movies_and_TV$style=as.factor(Review_Movies_and_TV$style)
table(Review_Movies_and_TV$style)

#Si collega la variabile "style" ai metadata
A=Review_Movies_and_TV[,5:6]
a=unique(A)
a$time=1
a=a %>% arrange(asin, style)
for(j in 1:(length(a$asin)-1)){
  if(a$asin[j+1]==a$asin[j])
    a$time[j+1]=a$time[j]+1
}

b=reshape(a, idvar = "asin", timevar = "time", direction = "wide")
```

```
for(j in 1:(length(b$asin))){
  if(is.na(b$style.2[j]))
    b$style[j]=paste(b$style.1[j])
  else{
     b$style[j]=paste(b$style.1[j],b$style.2[j], sep=" - ")
  }
}


b=b[,c(1,4)]
b$style=as.factor(b$style)

#Si collega la nuova variabile al dataset dei metadati
Meta_Movies <- merge(Meta_Movies,b, by.x = "asin",all=F)
rm(a,A,b,j)

#Si collega "mean_overall" ai metadati
A=Review_Movies_and_TV %>% group_by(asin) %>% summarise_at(vars(overall),
    list(MeanOverall = mean))
Meta_Movies <- merge(Meta_Movies,A, by.x = "asin",all=F)
rm(A)
Meta_Movies$MeanOverall=as.numeric(Meta_Movies$MeanOverall)

#Tabella di frequenza per il voto
table(Review_Movies_and_TV$overall)
round(prop.table(table(Review_Movies_and_TV$overall))*100,digits=2)

#Creazione variabile "num_review"
A=table(Review_Movies_and_TV$asin)
A=as.data.frame(A)
A$asin=A$Var1
A$num_review=A$Freq
A = A %>% dplyr::select(-c(Var1,Freq))
Meta_Movies <- merge(Meta_Movies,A, by.x = "asin",all=F)
```

# Statistiche descrittive

```
#Figura 3.3: Diagramma a barre sulla valutazione complessiva delle
    recensioni
a=data.frame(table(Review_Movies_and_TV$overall))
ggplot(a, aes(x=Var1, y=Freq)) +
  geom_bar(fill=rgb(0.1,0.4,0.7,1) ,stat = "identity")+
  scale_y_continuous(breaks=c(0,1000000,3000000,5000000),labels =
      label_comma())+
  ylab(expression(bold('Frequenza assoluta'))) +
  xlab(expression(bold('Overall')))

#Figura 3.4: Diagramma a barre per il tipo di formato
a=data.frame(table(Review_Movies_and_TV$style))
ggplot(a, aes(x=Var1, y=Freq, fill=Var1)) +
  geom_bar(stat="identity")+
  scale_fill_manual(values=c("#1E90FF",
                             "#0000CD"))+
  scale_y_continuous(breaks=c(0,1000000,2000000,3000000,4000000,5000000),labels
      = label_comma())+
  ylab(expression(bold('Frequenza assoluta'))) +
  xlab(expression(bold('Formato')))

a=table(Meta_Movies$style)
prop.table(a)
a=data.frame(a)
ggplot(a, aes(x=Var1, y=Freq, fill=Var1)) +
  geom_bar(stat = "identity")+
  scale_fill_manual(values=c("#F0F8FF",
                             "#1E90FF",
                             "#0000CD"))+
  scale_y_continuous(breaks=c(0,50000,100000,150000),labels =
      label_comma())+
  ylab(expression(bold('Frequenza assoluta'))) +
  xlab(expression(bold('Formato')))

#Ricodifica variabile "style"
Meta_Movies$style=as.character(Meta_Movies$style)
table(Meta_Movies$style)
Meta_Movies$style[Meta_Movies$style=="Amazon Video"]="Amazon Video - DVD"
Meta_Movies$style=as.factor(Meta_Movies$style)
table(Meta_Movies$style)
```

```
#Figura 3.5: Istogramma per la frequenza del prezzo degli articoli.
hist(Meta_Movies$price, xlim = c(1,49), xlab = "Prezzo", ylim =
    c(1,9000), ylab = "Frequenza assoluta", breaks = 1000, col =
    "lightblue")
```

## A.3 Implementazione delle regole associative

```
#Caricamento file con testo delle recensioni
TextReviews=read.csv("Text_Reviwes.csv", sep=",")

#Si elimina la variabile di conteggio
TextReviews = TextReviews %>% dplyr::select(-c(X))

#Si eliminano i duplicati
TextReviews=distinct(TextReviews) #dataset senza duplicati

#Sieliminano i prodotti che non hanno metadata
TextReviews=subset(TextReviews, TextReviews$asin %in% Meta_Movies$asin)

#Si collega il testo delle recensioni al dataset
Review_MT <- merge(Review_MT,TextReviews, by=
    c("asin","unixReviewTime","reviewerID"),all=F)

#Si collega il titolo dei film al dataset
MMs = Meta_Movies %>% dplyr::select(c(asin,title))
Review_MT <- merge(Review_MT,MMs, by.x = "asin",all=F)

#Si eliminano le recensioni duplicate per articoli con formato diverso
Review_MT=distinct(Review_MT, overall, reviewTime, reviewerID,
    reviewText, .keep_all = T)

#Si considera un range temporale di 6 anni (l'ultima recensione
    disponibile risale al 2018-10-03)
Review_MT=subset(Review_Movies_and_TV, reviewTime>"2012-10-03")
#Si eliminano acquirenti con ordini complessivi inferiori a 50
Review_MT=Review_MT[Review_MT$reviewerID %in%
    names(which(table(Review_MT$reviewerID) > 50)), ]
```

```
#Trasformazione in fattore delle variabili "reviewerID" e "asin"
Review_MT$reviewerID=factor(Review_MT$reviewerID)
Review_MT$asin=factor(Review_MT$asin)

nlevels(Review_MT$reviewerID) #2561 utenti
nlevels(Review_MT$asin) #61866 prodotti

#Si crea una lista
acquisti = split(x=Review_MT[,"asin"],f=Review_MT$reviewerID)

#rimozione dei duplicati
acquisti = lapply(acquisti,unique)

#Trasformazione della lista in oggetto di classe 'transactions'
acquisti = as(acquisti,"transactions")

#Costruzione delle regole con algoritmo a priori
moviesrules <-
    apriori(acquisti,parameter=list(support=.0142,confidence=.645,minlen=1))
inspect(moviesrules)

moviesrules=sort(moviesrules, by="confidence")
moviesrules.df<-as(moviesrules,"data.frame")

#Separazione delle regole ridondanti
is.redundant(moviesrules)
ruled.pruned=inspect(moviesrules[is.redundant(moviesrules)])
rule=inspect(moviesrules[!is.redundant(moviesrules)])
rule_df=as(rule,"data.frame")
rule_df=rule_df[order(rule_df$rhs,rule_df$confidence),]

#Si prendono regole non ridondanti in base alla fiducia
rule_nr<-moviesrules[!is.redundant(moviesrules,measure="confidence")]

#Dataset regole finali
rule_nrdf <- as(rule_nr, "data.frame")
rule_nrdf<-rule_nrdf[order(rule_nrdf$lift,decreasing = TRUE),]
#save(rule_nrdf,file="regole.RData")
load("regole.Rdata")
```

```
library(arulesViz)

#Figura 3.6: Scatter plot rappresentativo di tutte le regole associative
    relativamente a supporto, fiducia e lift
plot(rule_nr, jitter=0.5, measure=c("support", "confidence"), shading =
    "lift", interactive = T)

#Figura 3.7: Scatter plot rappresentativo di tutte le regole associative
    relativamente a supporto, fiducia e ordine
plot(rule_nr, jitter=0.5, measure=c("support", "confidence"), shading =
    "lift", interactive = T, method = "two-key plot")

#Figura 3.8: Parallel coordinates plot per le 10 regole con maggior
    fiducia
subrules2 <- head(rule_nr, n = 10, by = "confidence")
plot(subrules2, method = "paracoord", control = list(reorder = TRUE))

#Figura 3.9: Grouped matrix per le 70 regole con maggior fiducia
    raggruppate in 10 gruppi
subrules2 <- head(rule_nr, n = 70, by = "confidence")
plot(subrules2, method = "grouped")
```

## A.4 Implementazione dei modelli per dati di rete

```
#Costruzione della matrice di adiacenza per "asin" e "also_buy"
MM=Meta_Movies %>% select(c(asin,also_buy))

#Si eliminano le righe con "also_buy" <5 o >20
MM=subset(MM, stri_length(MM$also_buy)>50)
MM=subset(MM, stri_length(MM$also_buy)<220)

#Divisione della variabile "also_buy" in piu' variabili
MM=arrange(MM,asin)
a=max(nchar(MM$also_buy))
b=(a+1)/11
MM2=MM %>% separate(also_buy, paste("Also_Buy",1:b,sep="_"), " ", extra =
    "merge")
```

```
#Creazione di un nuovo dataset con 2 variabili
MM3=MM2 %>% select(-asin)
pr=as.matrix(MM3)
ve=as.vector(pr)
ve2=rep(MM2$asin,b)
MM4=cbind(ve2,ve)

colnames(MM4)[1] <- "asin"
colnames(MM4)[2] <- "also_buy"
MM4=as.data.frame(MM4)

#Si eliminano i valori mancanti generati
MM4= MM4 %>% drop_na("also_buy")

#Si eliminano le righe con "aslo_buy" non presenti in "asin"
MM4=subset(MM4, MM4$also_buy %in% Meta_Movies$asin)

#Si selezionano alcuni degli asin piu' popolari per la creazione della
    matrice di adiacenza
MM$b=0
for(j in 1:length(MM$also_buy)){
  if(sum(str_detect(MM$also_buy[j], "B00WR531Q6 | B009INAHZS |
      B001H5X6S0")) > 0)
    MM$b[j]=1
}
MM=subset(MM, MM$b==1)

MM2=MM %>% separate(also_buy, paste("Also_Buy",1:b,sep="_"), " ", extra =
    "merge")

MM3=MM2 %>% select(-c(asin,b))
pr=as.matrix(MM3)
ve=as.vector(pr)
ve2=rep(MM2$asin,b)
MM4=cbind(ve2,ve)

colnames(MM4)[1] <- "asin"
colnames(MM4)[2] <- "also_buy"
MM4=as.data.frame(MM4)
```

```
#Si eliminano i valori mancanti generati
MM4= MM4 %>% drop_na("also_buy")

#Si eliminano le righe con "aslo_buy" non presenti in "asin"
MM4=subset(MM4, MM4$also_buy %in% Meta_Movies$asin)
MM4=arrange(MM4,asin)


#Si trovano ulteriori collegamenti da "also_buy" ad "asin"
DD=Meta_Movies %>% select(c(asin,also_buy))
DD$also_buy=gsub("c","",as.character(DD$also_buy))
DD$also_buy=gsub("[[:punct:]]", "",as.character(DD$also_buy))
DD=subset(DD, DD$asin %in% MM4$also_buy)
a=max(nchar(DD$also_buy))
b=(a+1)/11
DD2=DD %>% separate(also_buy, paste("Also_Buy",1:b,sep="_"), " ", extra =
    "merge")

DD3=DD2 %>% select(-c(asin))
pr=as.matrix(DD3)
ve=as.vector(pr)
head(ve)
ve2=rep(DD$asin,b)
DD4=cbind(ve2,ve)

colnames(DD4)[1] <- "asin"
colnames(DD4)[2] <- "also_buy"
DD4=as.data.frame(DD4)

#Si eliminano i valori mancanti generati
DD4= DD4 %>% drop_na("also_buy")

#Si eliminano le righe con "aslo_buy" non presenti in "asin"
DD4=subset(DD4, DD4$also_buy %in% Meta_Movies$asin)

#Generazione connessioni da "also_buy" a "asin" o "also_buy"
DD5=subset(DD4, DD4$also_buy %in% MM4$asin)
DD6=subset(DD4, DD4$also_buy %in% MM4$also_buy)
DD4=rbind(DD5,DD6)
```

```
CC=rbind(MM4,DD4)
MM4=distinct(CC)

#Funzione per la creazione della matrice di adiacenza
matrice_di_adiacenza <- function(dat,
                                          symmetric = TRUE,
                                          node_columns = c(1, 2)
) {
  #Vincolo di 2 colonne
  if (length(node_columns) != 2) {
    stop("Le colonne devono essere 2")
  }

  col1 <- node_columns[1]
  col2 <- node_columns[2]
  nodes1 <- as.character(dat[[col1]])
  nodes2 <- as.character(dat[[col2]])
  dat <- dat[nodes1 != nodes2, ]

  #Creazione di una matrice vuota
  asin <- unique(c(nodes1, nodes2))
  n_asin <- length(asin)
  adjacency <- matrix(0, nrow = n_asin, ncol = n_asin,
                      dimnames = list(asin, asin))

  #Identificazione delle interazioni a coppie
  adjacency[as.matrix(dat[, node_columns])] <- 1
  #Aggiunta di interazioni simmetriche
  if (symmetric) {
    adjacency[as.matrix(dat[, rev(node_columns)])] <- 1
  }
  return(adjacency)
}

ADJ=matrice_di_adiacenza(MM4,symmetric = F,
                                   node_columns = c(1, 2))

net = graph_from_adjacency_matrix(ADJ, mode = 'directed', weighted =
    NULL, diag=FALSE)
isSymmetric(ADJ)
```

```
edge_density(net)
degree(net)
betweenness(net)
diameter(net)

transitivity(net)
reciprocity(net)
gr=membership(cluster_fast_greedy(as.undirected(net)))
assortativity(net,gr)
```

### Creazione delle reti

```
set.seed(111)

product=rbind(MM4$asin,MM4$also_buy)
product=as.data.frame(unique(as.vector(product)))
product=as.data.frame(product[match(colnames(ADJ), product$product),])
product$asin=product$product
product=product%>% dplyr::select(-c(product))

#Costruzione variabili diadiche
#Max_category
P=Meta_Movies[,c(1,9)]
cat <- merge(product,P, by.x = "asin",all=F)
cat=cat[match(colnames(ADJ), cat$asin),]

categorie=matrix(NA,length(cat$asin),length(cat$asin))
rownames(categorie)=colnames(categorie)=colnames(ADJ)

for(i in 1:length(cat$asin)){
  for(j in 1:length(cat$asin)){
    if(cat$max_category[i]== cat$max_category[j]){
      categorie[i,j]=1}
    else {categorie[i,j]=0}
  }
}

for(i in 1:length(cat$asin)){
```

```
  categorie[i,i]=0
}

#Min_category
P=Meta_Movies[,c(1,8)]
catmin <- merge(product,P, by.x = "asin",all=F)
catmin=catmin[match(colnames(ADJ), catmin$asin),]

categorie_min=matrix(NA,length(catmin$asin),length(catmin$asin))
rownames(categorie_min)=colnames(categorie_min)=colnames(ADJ)

for(i in 1:length(catmin$asin)){
  for(j in 1:length(catmin$asin)){
    if(catmin$min_category[i]== catmin$min_category[j]){
      categorie_min[i,j]=1}
    else {categorie_min[i,j]=0}
  }
}

for(i in 1:length(catmin$asin)){
  categorie_min[i,i]=0
}

#Style
P=Meta_Movies[,c(1,10)]
style <- merge(product,P, by.x = "asin",all=F)
style=style[match(colnames(ADJ), style$asin),]

sty=matrix(NA,length(style$asin),length(style$asin))
rownames(sty)=colnames(sty)=colnames(ADJ)

for(i in 1:length(style$asin)){
  for(j in 1:length(style$asin)){
    if(style$sty[i]== style$sty[j]){
      sty[i,j]=1}
    else {sty[i,j]=0}
  }
}

for(i in 1:length(style$asin)){
```

```
  sty[i,i]=0
}

#Variabili di nodo
P=Meta_Movies[,c(1,4,6,7,8,9,10,11,12)]
node <- merge(P,product, by.x = "asin",all=F)
node=node[match(colnames(ADJ), node$asin),]

dati=list(dyadvars=array(dim=c(length(cat$asin),length(cat$asin),4)),
    nodevars=array(dim=c(length(cat$asin),8)))
dimnames(dati$dyadvars)[[1]]=dimnames(dati$dyadvars)[[2]]=rownames(ADJ)
dimnames(dati$dyadvars)[[3]]=c("ADJ","max_category","sty","min_category")
dati$dyadvars[,,1]=ADJ
dati$dyadvars[,,2]=categorie
dati$dyadvars[,,3]=sty
dati$dyadvars[,,4]=categorie_min
dimnames(dati$nodevars)[[1]]=rownames(ADJ)
dimnames(dati$nodevars)[[2]]=colnames(node)[2:9]
dati$nodevars[,1]=node$max_category
dati$nodevars[,2]=node$rank
dati$nodevars[,3]=node$price
dati$nodevars[,4]=node$num_cat
dati$nodevars[,5]=node$min_category
dati$nodevars[,6]=node$style
dati$nodevars[,7]=node$MeanOverall
dati$nodevars[,8]=node$num_review

X=dati$nodevars
Xc=dati$dyadvars[,,c(2,3,4)]

V(net)$category=node$max_category
V(net)$rank=node$rank
V(net)$price=node$price
V(net)$num_cat=node$num_cat
V(net)$min_category=node$min_category
V(net)$sty=node$style

V(net)$count=rowSums(ADJ) + 1
colrs2=rev(hcl.colors(9, "spectral"))
V(net)$color2=colrs2[V(net)$count]
```

```
colori2=V(net)$color2

for(i in 1:length(catmin$asin)){
 if(is.na(colori2)[i]){
   colori2[i]=colrs2[9]}}

#Grafici per la prima rete
library(statnet)

V(net)$colori=rep("grey30",70)
colori=as.vector(V(net)$colori)
colori[68]="red"
colori[64]="blue"

#Figura 3.10: Rete con nodi fissati a priori
set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex = 1.3,
      vertex.col = colori, vertex.border = "black",
         mode="fruchtermanreingold")
legend(-45,25, c("Super Mario World: The Complete Series","Best of the
    Adventures of Sonic the Hedgehog"),
       pch=21, pt.bg = colrs, pt.cex = 2, cex=0.8, bty="n",ncol = 1)

#Figura 3.11: Rete per numero di archi in uscita (Kids & Family)
set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex = 1.3,
      vertex.col = colori2, vertex.border = "black",
         mode="fruchtermanreingold")
legend(-40,25, c("0 archi","1 arco","2 archi","3 archi","4 archi",
                 "5 archi","6 archi","7 archi", ">7 archi"),
       pch=21, pt.bg = colrs2, pt.cex = 2, cex=0.8, bty="n",ncol = 1)

node$min_category=as.character(node$min_category)
genere=node$min_category
genere[genere=="Anime"]="Anime & Manga"
genere=as.factor(genere)
V(net)$genere=genere

colrs=c("green","red","orange","yellow")
V(net)$color=colrs[V(net)$genere]
```

```
colori=as.vector(V(net)$color)

#Figura 3.12: Rete per categoria dei film
set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex = 1.3,
      vertex.col = colori, vertex.border = "black",
          mode="fruchtermanreingold")
legend(-45,25, c("All Sony Pictures Titles","Animation","Anime &
    Manga","Kids & Family"),
       pch=21, pt.bg = colrs, pt.cex = 2, cex=0.8, bty="n",ncol = 1)

#Statistiche a livello di nodo
deg=degree(ADJ, gmode="digraph")
bet=betweenness(ADJ, gmode="digraph")
cls=closeness(ADJ, gmode="digraph", cmode = "suminvdir")

rescale= function(nchar, low, high){
  min_d=min(nchar)
  max_d=max(nchar)
  rslc=((high-low)*(nchar-min_d))/(max_d-min_d)+low
  rslc}

col=rep("gray100", 70)
col[order(deg, decreasing = TRUE)[1]]="blue"

#Figura 3.13: Confronto tra indici di centralita'
set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex =
    rescale(deg,0.5,2.5),
      vertex.col = col, vertex.border = "black",
          mode="fruchtermanreingold")

col=rep("gray100", 70)
col[order(bet, decreasing = TRUE)[1]]="red"

set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex =
    rescale(sqrt(bet+1),0.5,2.5),
      vertex.col = col, vertex.border = "black",
          mode="fruchtermanreingold")
```

```
col=rep("gray100", 70)
col[order(cls, decreasing = TRUE)[1]]="green"

set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex =
    rescale(cls+1,0.5,2.5),
      vertex.col = col, vertex.border = "black",
          mode="fruchtermanreingold")

detach(package:statnet)

#Grafici per la seconda rete
library(statnet)

#rete con nodi a priori
V(net)$colori=rep("grey30",115)
colori=as.vector(V(net)$colori)
colori[78]="red"
colori[36]="blue"
colrs=c("red","blue")

#Figura 3.14: Rete con nodi fissati a priori
set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex = 1.3,
      vertex.col = colori, vertex.border = "black",
          mode="fruchtermanreingold")
legend(-60,45, c("Family of Strangers","Depression: Out of the Shadows"),
       pch=21, pt.bg = colrs, pt.cex = 2, cex=0.8, bty="n",ncol = 1)

#Figura 3.15: Rete per numero di archi in uscita
set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex = 1.3,
      vertex.col = colori2, vertex.border = "black",
          mode="fruchtermanreingold")
legend(-60,45, c("0 archi","1 arco","2 archi","3 archi","4 archi",
                 "5 archi","6 archi","7 archi", ">7 archi"),
       pch=21, pt.bg = colrs2, pt.cex = 2, cex=0.8, bty="n",ncol = 1)

node$min_category=as.character(node$min_category)
```

```
genere=node$min_category
genere[genere=="All A&E Titles"]="altro"
genere[genere=="All HBO Titles"]="altro"
genere[genere=="All Lionsgate Titles"]="altro"
genere[genere=="All MGM Titles"]="altro"
genere[genere=="Christmas"]="altro"
genere[genere=="Exercise & Fitness"]="altro"
genere[genere=="MOD CreateSpace Video"]="altro"
genere[genere=="General"]="altro"
genere[genere=="Movies"]="altro"
genere[genere=="Romance"]="altro"
genere[genere=="Horror"]="altro"

genere=as.factor(genere)
table(genere)
V(net)$genere=genere

colrs=c("purple","yellow","green","gray","blue","red","orange","pink")
V(net)$color=colrs[V(net)$genere]
colori=as.vector(V(net)$color)

#Figura 3.16: Rete per categoria degli articoli
set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex = 1.3,
      vertex.col = colori, vertex.border = "black",
         mode="fruchtermanreingold")
legend(-60,45, c("Action & Adventure","All Sony Pictures Titles","All
    Titles","Altro",
                "Documentary","Drama","Kids & Family","Special
                    Interests"),
       pch=21, pt.bg = colrs, pt.cex = 2, cex=0.8, bty="n",ncol = 1)

#Statistiche a livello di nodo
deg=degree(ADJ, gmode="digraph")
bet=betweenness(ADJ, gmode="digraph")
cls=closeness(ADJ, gmode="digraph", cmode = "suminvdir")

rescale= function(nchar, low, high){
  min_d=min(nchar)
  max_d=max(nchar)
```

```
  rslc=((high-low)*(nchar-min_d))/(max_d-min_d)+low
  rslc}

col=rep("gray100", 115)
col[order(deg, decreasing = TRUE)[1]]="blue"

#Fig. 3.17: Confronto tra indici di centralita'
set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex =
    rescale(deg,0.5,2.5),
      vertex.col = col, vertex.border = "black",
          mode="fruchtermanreingold")

col=rep("gray100", 115)
col[order(bet, decreasing = TRUE)[1]]="red"

set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex =
    rescale(sqrt(bet+1),0.5,2.5),
      vertex.col = col, vertex.border = "black",
          mode="fruchtermanreingold")

col=rep("gray100", 115)
col[order(cls, decreasing = TRUE)[1]]="green"

set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex =
    rescale(cls+1,0.5,2.5),
      vertex.col = col, vertex.border = "black",
          mode="fruchtermanreingold")

detach(package:statnet)

#Grafici per la terza rete
library(statnet)

#rete con nodi a priori
V(net)$colori=rep("grey30",147)
colori=as.vector(V(net)$colori)
colori[130]="red"
```

```
colori[111]="blue"
colori[91]="green"
colrs=c("red","blue","green")

#Figura 3.18: Rete con nodi fissati a priori
set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex = 1.3,
      vertex.col = colori, vertex.border = "black",
         mode="fruchtermanreingold")
legend(-70,38, c("12 Film Action Pack","Cast A Long Shadow","Yeti:
   Maneater Series"),
       pch=21, pt.bg = colrs, pt.cex = 2, cex=0.8, bty="n",ncol = 1)

#Figura 3.19: Rete per numuero di archi
set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex = 1.3,
      vertex.col = colori2, vertex.border = "black",
         mode="fruchtermanreingold")
legend(-70,38, c("0 archi","1 arco","2 archi","3 archi","4 archi",
                 "5 archi","6 archi","7 archi", ">7 archi"),
       pch=21, pt.bg = colrs2, pt.cex = 2, cex=0.8, bty="n",ncol = 1)

#Rete senza nodi a priori
ADJ2=ADJ[-c(130,111,91),-c(130,111,91)]
net2 = graph_from_adjacency_matrix(ADJ2, mode = 'directed', weighted =
   NULL, diag=FALSE)

#Fig. 3.20: Rete senza nodi a priori
set.seed(123)
gplot(ADJ2, gmode = "digraph", edge.col = "gray60", vertex.cex = 1.3,
      vertex.col = "gray30", vertex.border = "black",
         mode="fruchtermanreingold")

#rete min_category
node$min_category=as.character(node$min_category)
genere=node$min_category
genere[genere=="All MGM Titles"]="altro"
genere[genere=="Animation"]="altro"
genere[genere=="Classics"]="altro"
genere[genere=="Comedy"]="altro"
```

```
genere[genere=="Documentary"]="altro"
genere[genere=="Kids & Family"]="altro"
genere[genere=="MOD CreateSpace Video"]="altro"
genere[genere=="Movies"]="altro"
genere[genere=="Mystery & Thrillers"]="altro"
genere[genere=="Romance"]="altro"
genere[genere=="Science Fiction"]="altro"
genere[genere=="All Sony Pictures Titles"]="altro"
genere[genere=="All Titles"]="altro"
genere=as.factor(genere)
V(net)$genere=genere

colrs=c("yellow","green","gray","blue","purple","red","orange")
V(net)$color=colrs[V(net)$genere]
colori=as.vector(V(net)$color)

#Figura 3.21: Rete per categoria degli articoli
set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex = 1.3,
      vertex.col = colori, vertex.border = "black",
         mode="fruchtermanreingold")
legend(-70,38, c("Action & Adventure","All Lionsgate
    Titles","Altro","Fantasy","Drama","Horror","Westerns"),
       pch=21, pt.bg = colrs, pt.cex = 2, cex=0.8, bty="n",ncol = 1)

#Figura 3.22: Rete per formato degli articoli
colrs=c("red","royalblue")
V(net)$color=colrs[V(net)$sty]
colori=as.vector(V(net)$color)

set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex = 1.3,
      vertex.col = colori, vertex.border = "black",
         mode="fruchtermanreingold")
legend(-70,38, c("Amazon Video - DVD","DVD"),
       pch=21, pt.bg = colrs, pt.cex = 2, cex=0.8, bty="n",ncol = 1)

#Statistiche a livello di nodo
deg=degree(ADJ, gmode="digraph")
bet=betweenness(ADJ, gmode="digraph")
```

```
cls=closeness(ADJ, gmode="digraph", cmode = "suminvdir")

rescale= function(nchar, low, high){
  min_d=min(nchar)
  max_d=max(nchar)
  rslc=((high-low)*(nchar-min_d))/(max_d-min_d)+low
  rslc}

col=rep("gray100", 147)
col[order(deg, decreasing = TRUE)[1]]="blue"

#Figura 3.23: Confronto tra indici di centralita'
set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex =
    rescale(deg,0.5,2.5),
      vertex.col = col, vertex.border = "black",
          mode="fruchtermanreingold")

col=rep("gray100", 147)
col[order(bet, decreasing = TRUE)[1]]="red"

set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex =
    rescale(sqrt(bet+1),0.5,2.5),
      vertex.col = col, vertex.border = "black",
          mode="fruchtermanreingold")

col=rep("gray100", 147)
col[order(cls, decreasing = TRUE)[1]]="green"

set.seed(123)
gplot(ADJ, gmode = "digraph", edge.col = "gray60", vertex.cex =
    rescale(cls+1,0.5,2.5),
      vertex.col = col, vertex.border = "black",
          mode="fruchtermanreingold")

detach(package:statnet)
```

### Modelli per dati di rete

```
#Modello ANOVA

RowAsin<-matrix(rownames(ADJ),nrow(ADJ),ncol(ADJ))
ColAsin<-t(RowAsin)

mod = lm(c(ADJ) ~ c(RowAsin) + c(ColAsin), na.action = na.exclude)
anova(mod)
summary(mod)

rmean<-rowMeans(ADJ,na.rm=TRUE)
cmean<-colMeans(ADJ,na.rm=TRUE)
muhat<-mean(ADJ,na.rm=TRUE)
ahat<-rmean-muhat
bhat<-cmean-muhat

# effetti di riga
ahat[1:5]
head(sort(ahat,decreasing=TRUE) )
# effetti di colonna
bhat[1:5]
head(sort(bhat,decreasing=TRUE) )

# Social Relation Model
fit_SRM<-ame(ADJ, family = "bin")
summary(fit_SRM)

#Media generale
mean(fit_SRM$BETA)

#Stima degli effetti di riga
fit_SRM$APM
#Stima degli effetti di colonna
fit_SRM$BPM

# statistiche osservate
gofstats(ADJ)

#SRRM
```

```
fit_srrm<-ame(ADJ, family = "bin", Xdyad = Xc)
summary(fit_srrm)

#AME
fit_ame<-ame(ADJ, family = "bin", Xdyad = Xc, R=1)
summary(fit_ame)
```