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

Scienze Statistiche

Relazione Finale

CONFRONTO DI METODI STATISTICI PER LA RILEVAZIONE DELLA DIFFERENZIALE ESPRESSIONE IN STUDI DI RNA-SEQ

Relatore Prof.ssa Chiara Romualdi
Dipartimento di Scienze Statistiche

Laureando Manuela Ravagnan
Matricola N. 1111351

Anno Accademico 2016/2017

Sommario

È noto che differenti condizioni biologiche e di salute sono ampiamente caratterizzate da diversi livelli di espressione genica. Conoscere le specifiche caratteristiche genetiche dei singoli individui è fondamentale nel passaggio da un approccio classico della medicina, che propone la stessa cura a tutti i pazienti che hanno la stessa diagnosi, ad un approccio di medicina più personalizzata. Per questo trovare i geni che sono differenzialmente espressi tra due condizioni è una parte importante nel processo di comprensione delle basi molecolari delle variazioni fenotipiche. Negli anni passati, la tecnologia *microarray* è stata la più usata per la fase di quantificazione dell'espressione genica, ma recentemente la tecnologia RNA-Seq è diventata un'alternativa molto competitiva. Paragonata ai microarrays, la tecnologia RNA-Seq offre diversi vantaggi che includono un range di livelli di espressione più ampio, un elevato throughput, un miglior coverage del genoma, rumore di fondo inferiore e la possibilità di esplorare nuovi trascritti di cui non sono note le sequenze. Per queste ragioni, l'RNA-Seq è pronto a rimpiazzare la tecnologia microarray e a diventare il principale strumento per la quantificazione dell'espressione genica, favorito inoltre dalla diminuzione dei tempi e dei costi di sequenziamento derivanti dallo sviluppo di sequenziatori di nuova generazione (Next Generation Sequencing). Per sfruttare le opportunità e affrontare le sfide poste da questo tipo di dato relativamente nuovo, sono stati sviluppati numerosi pacchetti software creati appositamente per l'analisi della differenziale espressione di dati RNA-Seq. Questi metodi differiscono in termini di modelli per le conte, applicazione dello shrinkage, flessibilità del disegno e tipo di inferenza, ma attualmente non c'è un chiaro consenso su quale pratica sia la migliore. Per esplorare tale problema, in questo elaborato si è operato un confronto sistematico di otto metodi per l'analisi della differenziale espressione di dati di RNA-Seq (due dei quali declinati in due modi diversi). Tutti i metodi sono disponibili gratuitamente all'interno dell'ambiente R e sono stati valutati sia su dati simulati che su dati di RNA-Seq reali in modo da essere messi alla prova su una vasta gamma di situazioni che variano per numerosità campionaria, tecnica si simulazione utilizzata e eterogeneità del data set. L'obbiettivo di questo elaborato è capire se esistono dei metodi che funzionino uniformemente bene in tutte le condizioni sperimentali o che abbiano prestazioni affidabili anche quando la disponibilità delle informazione è scarsa (poca differenziale espressione o scarsa numerosità campionaria), in modo da definire delle linee guida da seguire per scegliere il metodo migliore

in base alle caratteristiche del data set oggetto di studio.
La tesi è suddivisa in sei capitoli: nel primo vengono introdotti i concetti biologici utili per la comprensione del fenomeno che si va ad analizzare, nel secondo vengono esposti i passi standard utilizzati nelle analisi, nel terzo si spiegano nel dettaglio le caratteristiche dei metodi che si andranno a comparare, nel quarto si riportano i disegni di simulazione, nel quinto si presentano i dati trattati e la patologia del caso di studio reale e nel sesto si espongono le analisi fatte e i risultati ottenuti.

Indice

Capitolo 1

Introduzione

1.1 L'espressione genica

L'espressione genica è il processo mediante il quale le informazioni contenute in un gene sono convertite in un prodotto genico funzionale, come una proteina o una molecola di RNA. L' espressione genica fa parte di tutte le creature viventi (eucarioti e procarioti), permette di generare le componenti macromolecolari fondamentali per la vita ed è quindi un procedimento molto complesso, composto da più fasi e finemente regolato.
Per capire meglio l'utilità dello studio dell'espressione genica è necessario introdurre alcune nozioni di base di biologia molecolare.

Il DNA è un polimero costituito da un insieme di nucleotidi disposti a formare una struttura a doppia elica, simile ad una scala a pioli disposta a spirale (Figura 1.1). Ciascun nucleotide è composto da uno scheletro di zucchero e da un gruppo fosfato, mentre i pioli sono costituiti da una delle quattro basi azotate. Ogni base presente su un filamento si lega in modo univoco a una base del filamento opposto mediante legame idrogeno: così l'Adenina (A) è appaiata alla Timina (T) e la Guanina (G) alla Citosina (C). La disposizione in sequenza di queste quattro basi costituisce l'informazione genetica, leggibile attraverso il codice genetico, che ne permette la traduzione in amminoacidi.
La sintesi della proteina avviene principalmente in due fasi:

1. fase di trascrizione: nella quale un gene, ovvero una porzione di DNA che contiene le informazione per creare una proteina, viene copiato su un filamento di RNA messaggero (mRNA).

2. fase di traduzione: nella quale il filamento di mRNA esce dal nucleo cellulare per unirsi ai ribosomi e creare la catena di amminoacidi che andranno a formare la proteina.

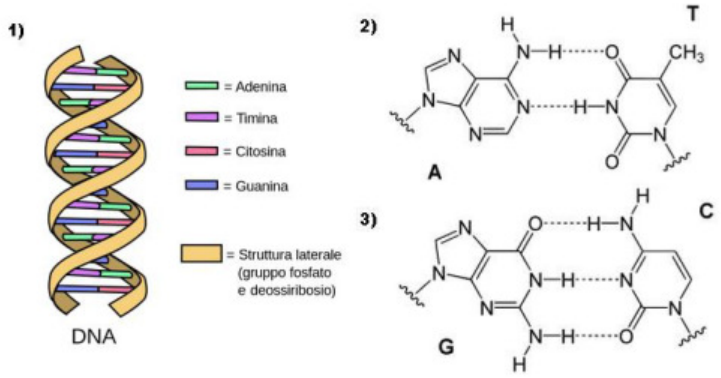


Figura 1.1: **1)** Rappresentazione semplificata della struttura a doppia elica del DNA. **2)** Legame idrogeno tra Adenina e Timina. **3)** Legame idrogeno tra Guanina e Citosina.

Ogni nucleo cellulare contiene il genoma completo. In termini molecolari il DNA di tutte le cellule di un individuo è identico, ma solo una piccola frazione di esso viene trascritta. I geni non espressi nelle cellule mantengono il loro potenziale di espressione, ma non vengono trascritti. Nei mammiferi alcuni geni detti *housekeeping* sono attivi in tutte le cellule perché codificano prodotti necessari al funzionamento generale della cellula (per esempio, i geni implicati nella sintesi proteica, nel metabolismo energetico cellulare, ecc.), mentre l' espressione genica di altri geni può essere fortemente ristretta a un tipo cellulare o di tessuto, grazie all'utilizzo di promotori tessuto-specifici (per esempio, i geni che partecipano alla funzione visiva sono attivi solamente nelle cellule della retina).

Lo studio dell'espressione genica in una determinata condizione sperimentale consiste nella misurazione della quantità di mRNA trascritto per ciascun gene presente nel DNA, al fine di trovare gruppi di geni più o meno espressi nell'organismo. A seconda della quantità di copie di mRNA trascritte, i caratteri o le funzioni biologiche regolate dal segmento di codice genetico in questione saranno più o meno espressi. L'utilità di ciò sta nella possibilità di effettuare un confronto tra il livello di espressione genica in tipi cellulari diversi o in condizioni patologiche diverse, per determinare il ruolo che i geni hanno in queste. Con geni *differenzialmente espressi* (DE) si intendono quei geni che hanno livelli di espressione significativamente diversi in condizioni biologiche diverse. Fissata una condizione di riferimento, i geni differenzialmente espressi possono essere classificati in *sovraespressi* o *sottoe-*

spressi. Naturalmente la significatività deve essere verificata con l'impiego di opportune tecniche statistiche e ciò apre la strada ad una moltitudine di problemi che sorgono dall'utilizzo di dati di questo tipo. Gli studi di espressione genica sono largamente usati e affrontano problematiche diverse. Sono utilizzati, per esempio, per comprendere meglio alcuni meccanismi biologici latenti, per individuare sottogruppi di malattie, per esaminare la risposta ai farmaci, per classificare i pazienti in gruppi diagnostici e prognostici.

1.2 Il sequenziamento

Il trascrittoma è l'insieme delle molecole di mRNA (o trascritti) presenti in una cellula. Un'applicazione della trascrittomica quantitativa è l'analisi differenziale dell'espressione genica, ottenuta confrontando i profili trascrizionali di due o più individui, tessuti o tipi cellulari. Ad esempio, informazioni sui trascritti in soggetti sani e soggetti malati possono permettere di rilevare quali geni sono espressi in maniera significativamente diversa fra i due gruppi e quindi di evidenziare differenze che la condizione patologica comporta.
L'analisi del trascrittoma può avvenire secondo due tecnologie: quelle basate su *ibridazione* e quelle, più recenti, basate su *sequenziamento*.

L'ibridazione si basa sulla proprietà dei nucleotidi di appaiarsi con i loro complementari fissati su un supporto. Di questa categoria fanno parte i *microarrays*, da anni largamente utilizzati per ottenere informazioni sull'espressione genica. Sono costituiti da un supporto solido a cui sono ancorate delle sonde di DNA, dette *probe*, in numero molto elevato per ogni gene e disposte in posizioni note. L'RNA estratto dalla cellula viene retrotrascritto, marcato con una particella fluorescente e ibridato con il microarray. L'intensità della fluorescenza è una misura di quante molecole hanno ibridato il probe, ovvero di quanto il gene associato al probe è espresso nella cellula. Questa tecnica presenta numerosi limiti: la necessità di conoscere a priori le sequenze geniche per la progettazione dei probe e il limitato range dinamico (cioè il rapporto fra i livelli di massima e minima espressione genica misurabili) dovuto al rumore di fondo e al fenomeno di saturazione del segnale.

Per sequenziamento si intende, invece, l'identificazione della sequenza di DNA fornita in input alla strumentazione. Una tecnica recente per la misura del trascrittoma basata sul sequenziamento è l'*RNA Sequencing* (RNA-Seq)[1], che si fonda sulle tecnologie di sequenziamento NGS (Next Generation Sequencing).
Il protocollo di un esperimento di RNA-Seq varia in base alla tecnologia utilizzata, ma è comunque possibile descriverne in linea generale i passaggi

principali (si veda Figura 1.2). I campioni di mRNA sono estratti dalle cellule

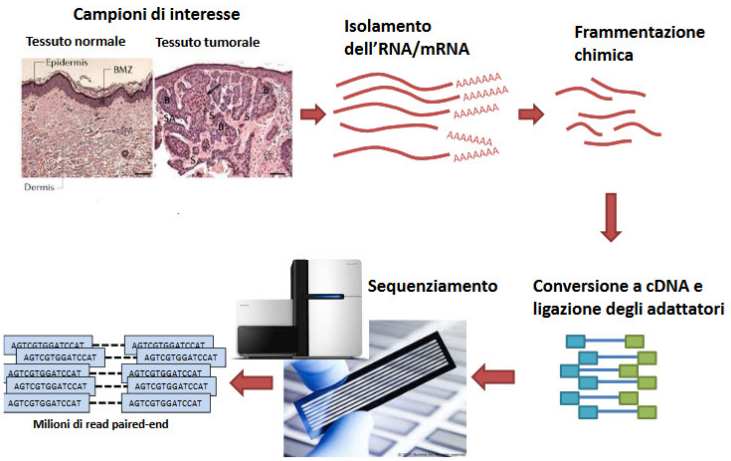


Figura 1.2: Preparazione della libreria e sequenziamento.

in analisi e preparati al sequenziamento separatamente: ciascuno campione viene frammentato casualmente in sequenze di dimensione inferiore, compresa tra i 200 e i 500 bp, per ottenere frammenti di dimensione compatibile con i sequenziatori in uso. Il processo di frammentazione è realizzato tramite idrolisi o nebulizzazione.
Ciascun frammento viene poi retrotrascritto in DNA (cDNA). La retrotrascrizione (o trascrizione inversa) è il processo di sintesi di un filamento di DNA complementare a partire da un filamento di RNA. Questo procedimento è eseguito al fine di aumentare la stabilità della molecola.
Successivamente ad ogni frammento di cDNA vengono legate delle sequenze specifiche, chiamate *adattatori*, che dipendono dal tipo di sequenziatore e che servono in varie fasi della preparazione del campione.
Le molecole di cDNA vengono amplificate mediante *Polymerase Chain Reaction* (PCR), procedimento che moltiplica il numero di copie di ciascun frammento per aumentarne la massa critica, e infine vengono fornite in input ai sequenziatori.

Il DNA ottenuto viene sequenziato ad alto throughput per ottenere frammenti a sequenza nota, detti *reads*, di lunghezza variabile in base alla tecnologia del sequenziatore utilizzato. Le reads sono quindi le sequenze, ottenute dal sequenziatore, che identificano l'ordine in cui si susseguono le basi nei frammenti di DNA. Esistono diversi tipi di sequenziatori che differiscono per le tecniche di sequenziamento, il tempo impiegato per le analisi, la lunghezza

delle reads prodotte, la quantità di reads prodotte e la percentuale di errore per ogni run.

1.3 La tecnologia RNA-Seq

In un esperimento RNA-Seq, le reads rappresentano i dati grezzi dai quali ricavare l'informazione sul livello di espressione dei geni nel campione. Più numerose sono le copie di un trascritto in un campione, più probabilità avrà quel trascritto di essere sequenziato e di generare reads. Per quantificare il numero di reads riferite a ciascun gene, le reads di ogni campione sono mappate su un genoma o un trascrittoma di riferimento.
La scelta del riferimento a cui allineare le reads (genoma o trascrittoma), cambia leggermente l'impostazione degli algoritmi di allineamento (si veda Figura 1.3). Per spiegare in cosa consiste questa differenza è necessario fare alcune precisazioni.

Il genoma è la totalità del DNA contenuto in una cellula di un organismo vivente ed è composto per la maggior parte da DNA intra-genico (67,5% [2]), ovvero sequenze spesso ripetute di cui non si è ancora compreso a fondo lo scopo, e da DNA genico (37,5%) che comprende quelle sequenze che costituiscono un gene o che partecipano alla sua regolazione. Nei geni, la sequenza nucleotidica del DNA che codifica un polipeptide eucariotico non è di norma continua, ma è al contrario divisa in segmenti. I segmenti non codificanti di DNA (ovvero rimossi prima della traduzione) che si trovano fra le regioni codificanti (ovvero le sequenze che vengono tradotte) sono noti come *introni*. Gli altri segmenti sono detti *esoni*, poiché vengono generalmente espressi attraverso la traduzione in sequenze di aminoacidi. Successivamente alla trascrizione, l'mRNA subisce un processo di maturazione in cui le sequenze introniche vengono eliminate, mentre gli esoni vengono legati insieme formando un'unica molecola di mRNA con una sequenza codificante continua. Questo processo prende il nome di *splicing dell'RNA*, e l'insieme degli mRNA maturi (trascritti) costituisce il trascrittoma.

Le reads che coprono due differenti esoni sono allineate senza problemi al trascrittoma poiché i trascritti maturi non hanno introni al loro interno e di conseguenza gli algoritmi di allineamento non permettono lunghi buchi tra una base e l'altra di una read rispetto al riferimento (Unspliced Aligners). Viceversa quando il riferimento è il genoma bisogna tenere in considerazione il fatto che una read che mappa a cavallo tra due esoni può essere divisa da potenzialmente migliaia di basi di sequenze introniche rispetto al riferimento, quindi gli algoritmi di allineamento, per allocare correttamente la read,

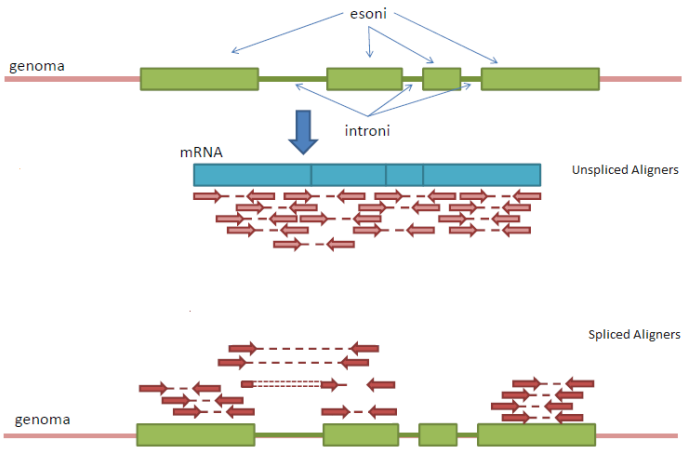


Figura 1.3: Esempio di allineamento al trascrtittoma e al genoma. Ciascuna coppia di frecce rosse indica una read.

devono permettere lunghi buchi tra una base e l'altra di una read (Spliced Aligners).

La fase di allineamento delle reads sul riferimento presenta un ulteriore aspetto critico: idealmente si vorrebbe trovare l'univoca posizione del genoma in cui il riferimento sia identico alla read. In realtà il riferimento non sarà mai una rappresentazione perfettamente identica del campione biologico, a causa di errori di sequenziamento e/o di imperfetta similarità tra le sequenze del campione d'interessa e quelle del riferimento. Lo scopo dell'allineamento diventa quindi l'identificazione della posizione del genoma o del trascrittoma in cui ogni read ottiene il miglior match con il riferimento.

Una volta determinate le posizioni delle reads sul riferimento, è possibile contare il numero di reads allineate su un gene o trascritto. Di nuovo si presentano due situazioni leggermente diverse: ad un gene possono corrispondere più trascritti, ovvero un gene può codificare più polipeptidi diversi, a seconda di quali tratti del gene siano considerati esoni nel corso della maturazione dell'RNA. Questo processo è chiamato *splincing alternativo* dell'RNA e i diversi trascritti che derivano dal medesimo gene sono chiamate *isoforme*. Le isoforme si caratterizzano per avere alcuni esoni in comune e questo può generare ambiguità nella fase di conta. Se l'obbiettivo è quello di contare tutte le reads che mappano su un gene allora il problema non si pone, ma se si vogliono contare le reads che mappano su un certo trascritto che ha delle isoforme, allora bisogna cercare di inferire a quale trascritto si riferiscono le reads che mappano sugli esoni comuni alle due isoforme e questo può portare

a stime distorte e variabili.

Il totale delle reads allineate su una regione d'interesse del genoma (un gene o un trascritto) è detto *count* e può essere inteso come una misura del livello di espressione di quella regione. I counts sono i dati finali di un esperimento di RNA-Seq per la quantificazione del trascrittoma. In esperimenti comparativi, le misurazioni coinvolte in un esperimento di RNA-Seq sono fatte su più campioni in modo da avere più variabilità biologica. Le espressioni di ciascun campione sono riassunte in un vettore di conte, mentre le espressioni di tutti i campioni sono messe insieme a formare una matrice. Si supponga di avere dati provenienti da m esperimenti di RNA-Seq, e ciascuno di essi produca conte per G regioni d'interesse (ad esempio per G trascritti). Statisticamente si tratta ciascun esperimento come un campione, e ciascuna regione d'interesse come una variabile. I dati a disposizione sono una matrice K di dimensione $G \times m$, il cui elemento K_{ij} è il numero di reads mappate sulla variabile i nell'esperimento j, con $1 \leq i \leq G$, $1 \leq j \leq m$. Questa misura è un numero intero non negativo, in contrasto con i valori continui ottenuti dai microarrays. Per loro stessa definizione, dal punto di vista statistico i counts rappresentano una somma di eventi aleatori indipendenti (la mappatura di ogni read sui geni). Possono quindi essere descritti da una variabile aleatoria che segue una determinata distribuzione statistica. I due modelli di distribuzione più utilizzati per descrivere i counts sono il modello di Poisson e il modello Binomiale Negativo.
L'intero processo di quantificazione dell'espressione è riassunto nella Figura 1.4.

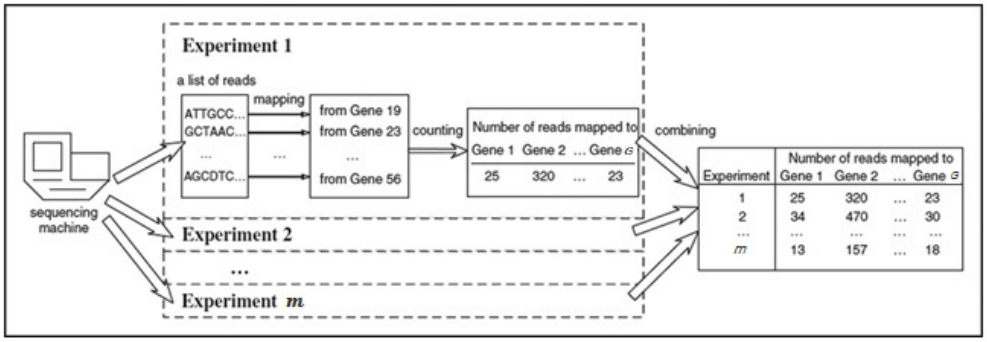


Figura 1.4: Procedimento per usare dati di RNA-Seq per analisi comparative.

Capitolo 2

Analisi

Una volta ottenuta la matrice delle conte grezze, si passa alla fase di analisi della differenziale espressione. Prima però, è necessario applicare delle trasformazioni ai dati al fine di risolvere alcune problematiche che sono intrinseche del tipo di dato con cui si sta lavorando e del modo in cui questo è stato ottenuto. I passi per l'analisi di una matrice di conte di RNA-Seq possono variare a seconda delle esigenze dei dati, ma in linea generale i passaggi principali da seguire sono:

1. Filtraggio;
2. Normalizzazione;
3. Identificazione dei geni differenzialmente espressi.

2.1 Filtraggio

Sebbene teoricamente si vogliano tenere tutti i geni nell'analisi, nella pratica può essere ragionevole filtrare quei geni che sono scarsamente espressi. Questo perché la stima dell'espressione di un gene è meno affidabile nel caso di geni con conte basse e quindi questi geni possono rappresentare una fonte di disturbo che influenza negativamente la sensibilità e la specificità nella maggior parte dei metodi di analisi di differenziale espressione. Lo scopo del filtraggio è quindi quello di rimuovere i geni che, a causa della loro scarsa intensità o variabilità complessiva, è improbabile che portino informazione riguardo il fenotipo d'interesse. Il filtraggio può essere fatto secondo diversi criteri, ad esempio, eliminando quei geni la cui somma totale delle conte su tutti i campioni, indipendentemente dal gruppo di trattamento, è inferiore a 10.

Inoltre si è visto che la perdita di potenza dovuta all'aggiustamento per test multipli può essere ridotta se i geni che hanno scarse o nulle possibilità di essere rilevati come DE vengono omessi dall'essere testati, purché il criterio di omissione sia indipendente dalla statistica test sotto l'ipotesi nulla [3]. L'idea è quella di tenere il numero di test più basso possibile, ma contemporaneamente tenere i geni d'interesse nel sottoinsieme di geni selezionati. Se geni realmente differenzialmente espressi sono sovra-rappresentati tra quelli selezionati nel filtraggio, il *false discovery rate* (FDR) [4] associato ad una certa soglia del test statistico sarà più basso a causa del filtraggio [5].

2.2 Normalizzazione

Lo scopo della normalizzazione è quello di rimuovere gli effetti sistematici, dovuti alla tecnologia, che si presentano nei dati, in modo da assicurare che gli artefatti tecnici abbiano un minimo impatto sui risultati.
Alcune fonti di variazione sistematica sono ereditate dalla procedura di Next Generetion Sequencing. Per esempio, la variazione nella composizione dei nucleotidi tra regioni genomiche implica che il coverage delle reads, ovvero il numero medio delle reads che "coprono" una base nota di riferimento, potrebbe non essere uniforme lungo il genoma. Inoltre, a parità di livello di espressione, saranno associate più reads ad un gene lungo piuttosto che ad un gene corto. Nelle analisi di differenziale espressione, dove i geni sono testati individualmente, questi *"within-sample"* biases sono ignorati dato che si assume che influenzino tutti i campioni in modo simile [6].
Altri tipi di non uniformità si verificano tra i campioni di un esperimento di RNA-Seq (*"between-sample"* biases). La prima fonte di variazione tra i campioni è la *profondità di sequenziamento* (o *dimensione della libreria*) che è il numero totale di reads mappate in un esperimento e che tipicamente è differente per differenti campioni. Questo comporta che le conte osservate non sono direttamente comparabili tra i campioni. Per esempio, se gli esperimenti 1 e 2 usano lo stesso campione biologico (in questo modo, ogni variabile è ugualmente espressa nei due esperimenti), ma l'esperimento 1 ha un milione di reads in totale, mentre l'esperimento 2 ha due milioni di reads in totale, allora è probabile che $K_{i2} \cong 2K_{i1}$, per qualunque i. Non si vuole quindi confondere tale effetto con la vera differenziale espressione.

Il modo più semplice di risolvere il problema della diversa dimensione delle librerie è quello di riscalare le conte in modo da ottenere dimensioni di libreria equivalenti per tutti i campioni. Tuttavia, questo tipo di normalizzazione non è in genere sufficiente. Infatti, anche se le dimensioni delle librerie sono identiche, è possibile che pochi geni altamente espressi catturi-

no la maggior parte delle reads sequenziate in un esperimento, lasciando solo poche reads distribuite per i geni restanti [7]. La presenza di questi pochi geni altamente espressi reprime così le conte di tutti gli altri geni che possono sembrare sottoespressi nel caso di una comparazione con un campione dove le reads sono più equamente distribuite, cosa che può portare ad avere un sacco di geni che sono indicati come differenzialmente espressi quando invece non lo sono (si veda Figura 2.1).

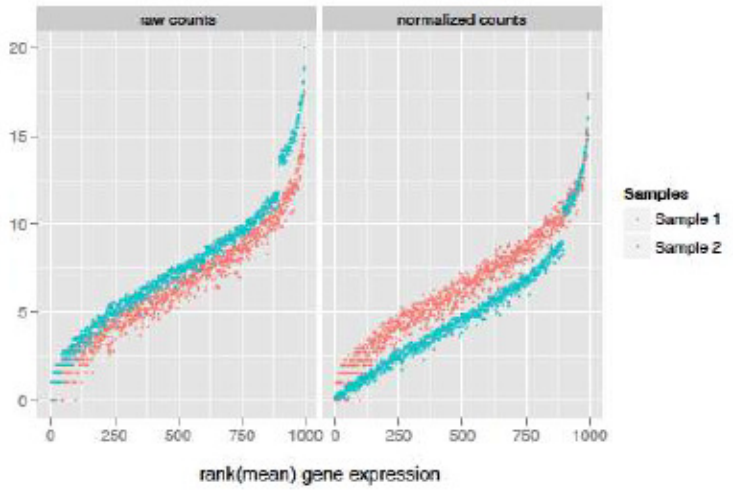


Figura 2.1: Distribuzione delle conte grezze di due campioni e delle conte normalizzate in base alla profondità di sequenziamento.

Per tenere conto di questa difficoltà e per tentare di rendere le conte comparabili tra campioni, sono stati proposti schemi di normalizzazione più complessi. Le normalizzazioni più comuni sono le *globali* che includono la stima di fattori di normalizzazione campione specifico che sono usati per riscalare le conte osservate. Esistono diversi metodi per calcolare questi fattori globali tra cui il Total Count (TC) [7], l'Upper Quartile (UQ) [7], il Median (Med) [8], la normalizzazione median-of-ratio implementata nel pacchetto DESeq di Bioconductor [9], la normalizzazione implementata da samr nel pacchetto samr di Bioconductor [10] e la Trimmed Mean of M values (TMM) implementata nel pacchetto edgeR di Bioconductor [11]. Usando questi metodi di normalizzazione, la somma delle conte normalizzate tra tutti i geni non sono necessariamente uguali tra i campioni (come sarebbe stato se solo la dimensione della libreria fosse usata per la normalizzazione), ma lo scopo è invece quello di rendere le conte normalizzate per i geni non differenzialmente

espressi simili tra i campioni (TMM, median-of-ratio e samr) oppure quello di rendere le distribuzioni delle conte simili tra campioni sulla base di un singolo quantile (TC, Med, UQ).
In alternativa ai metodi globali si possono considerare metodi *non globali* che operano trasformazioni più complesse come la Quantile (Q) [12, 13], la normalizzazione Reads Per Kilobase per Million mapped reads (RPKM) [14] o la Transcript Per Million (TPM) [15].

Particolare attenzione è stata data alla TPM al fine di verificare se questo tipo di normalizzazione incide sulle prestazioni fornite dai metodi d'inferenza per l'analisi della differenziale espressione considerati in questo lavoro. La TPM è anche il formato che molti software di quantificazione restituiscono al posto delle conte grezze, quindi è importante capire se si possono usare questi numeri al posto delle conte grezze con tutti i test.
Di seguito si riporta una breve descrizione dei metodi usati, ma per maggiori dettagli si vedano i riferimenti.

TMM: Si basa sull'ipotesi che la maggior parte dei geni non sia DE. Per prima cosa viene scelto un campione come riferimento, r, successivamente per ciascuno dei campioni rimanenti, j, viene calcolato un fattore TMM come una media pesata dei log rapporti tra il campione in considerazione e il riferimento, $M_i(j,r)$, dopo l'esclusione dei geni più espressi e dei geni con i log rapporti più grandi, G^*.

$$TMM(j,r) = \frac{\sum_{i \in G^*} w_i(j,r) M_i(j,r)}{\sum_{i \in G^*} w_i(j,r)}$$

In base all'ipotesi che i DE siano pochi, il fattore TMM dovrebbe essere vicino a 1. Se ciò non fosse, il valore fornisce una stima del fattore di correzione che deve essere applicato alla dimensione della libreria (e non alle conte grezze) per soddisfare le ipotesi. Per ottenere le conte normalizzate delle reads, questi fattori di normalizzazione sono riscalati per la media delle dimensioni delle librerie normalizzate. Le conte normalizzate si ottengono dividendo le conte grezze con questi fattori di normalizzazione riscalati.

median-of-ratio: Si basa sull'ipotesi che la maggior parte dei geni non sia DE. Il fattore di scala per un certo campione viene calcolato come la mediana, rispetto ai geni, del rapporto tra le conte grezze di quel campione e la media geometrica tra gli esperimenti del corrispettivo gene.

$$s_j = \underset{i:K_i^R \neq 0}{\text{median}} \frac{K_{ij}}{K_i^R} \quad \text{con} \quad K_i^R = \left(\prod_{j=1}^{m} K_{ij} \right)^{1/m}$$

L'idea è che i geni non DE dovrebbero avere conte simili tra gli esperimenti, quindi il rapporto dovrebbe essere circa 1. Assumendo che la maggior parte

dei geni non sia DE, il fattore così calcolato fornisce una stima del fattore di correzione che dovrebbe essere applicato a tutte le conte di un esperimento per soddisfare le ipotesi. Per ottenere le conte normalizzate è sufficiente dividere le conte grezze per il corrispettivo fattore.

samr: Le conte grezze sono divise per il numero totale di reads mappate nel corrispettivo esperimento per un certo insieme S di geni e moltiplicate per la media delle conte tra i campioni per quei geni appartenenti a S. L'insieme S contiene quei geni che sembrano non essere differenzialmente espressi.

$$\hat{s}_j = \frac{\sum_{i \in S} K_{ij}}{\sum_{i \in S} K_{i\cdot}} \quad \text{con} \quad K_{i\cdot} = \sum_{j=1}^{m} K_{ij}$$

Il metodo usato da samr risulta quindi una generalizzazione della Total Count, in cui l'insieme S è costituito dalla totalità dei geni presenti nel data set. Per determinare quali geni appartengono all'insieme S si usa un procedimento iterativo che prevede: 1) la stima di un indice di bontà di adattamento (GOF) per ogni gene appartenete a S (inizialmente S comprende tutti i geni), 2) la restrizione di S a quei geni il cui $GOF \in (0, 0.5)$ e 3) l'aggiornamento di $\hat{s}_j$. Si ripetono i passi fino a convergenza. Per ottenere le conte normalizzate è sufficiente dividere le conte grezze per il corrispettivo fattore, $\hat{s}_j$.

TPM: Lo scopo di questa trasformazione è normalizzare le conte sia rispetto alle differenti dimensioni della libreria sia rispetto alla lunghezza dei trascritti, L_i, dato che ci si aspetta che a parità di livello di espressione un trascritto lungo ottenga più reads di uno corto. Per calcolare i TPM si deve dividere ciascuna conta per la lunghezza del corrispettivo gene (RPK), successivamente si sommano tutti i valori RPK in un campione e si divide questa quantità per 1.000.000 (fattore di scala "per million"). Infine ciascun valore RPK in un campione viene diviso per il corrispettivo fattore di scala "per million".

$$\text{TPM}_{ij} = \frac{\frac{K_{ij}}{L_i}}{Prof_j} * 1.000.000 \quad Prof_j = \sum_{i=1}^{G} \frac{K_{ij}}{L_i}$$

Quando si usano i TPM, la somma di tutti i TPMs in ciascun campione è la stessa. Questo facilita la comparazione delle proporzioni di reads che mappano su un gene in ciascun esperimento. Ad esempio, se per un certo gene il valore di TPM nel campione A è 3.33 e nel campione B è 3.33, allora si può dire che la stessa percentuale di reads totali è mappata su quel gene in entrambi i campioni.

È importante tenere in considerazione che questi metodi si basano sull'assunzione che la maggior parte dei geni siano equivalentemente espressi nei

campioni, e che i geni differenzialmente espressi si dividano più o meno equamente in sovra e sotto espressi [8]. Altre strategie di normalizzazione possono essere utilizzate per risolvere altre distorsioni, che nascono ad esempio dalla diversa percentuale di GC contenuta nelle reads.

2.3 Identificazione dei geni differenzialmente espressi

Lo scopo dell'analisi è quello di identificare i geni il cui livello di espressione cambia tra le condizioni in oggetto di studio. Si vorrebbe usare un test statistico per decidere se, per un dato gene, una differenza osservata nelle conte delle reads tra due condizioni è significativa, cioè se è più grande di quella che ci si aspetterebbe se fosse dovuta unicamente ad una variazione casuale.
Una componente cruciale di questo tipo di analisi è quindi la procedura statistica usata per individuare i geni differenzialmente espressi. Il campo dell'analisi di differenziale espressione per dati di RNA-Seq è ancora nella sua infanzia e nuovi metodi vengono continuamente presentati, tuttavia non c'è un generale consenso riguardo a quale metodo sia il migliore.

In questo elaborato si è proceduto ad una valutazione comparativa di 8 metodi per l'analisi della differenziale espressione di dati di RNA-Seq (due dei quali declinati in due modi diversi) al fine di individuare se esiste un metodo o un insieme di metodi che funzionano uniformemente bene in tutte le condizioni sperimentali e se è possibile dare delle linee guida per la scelta del metodo da utilizzare date le condizioni sperimentali (numerosità campionaria, eterogeneità del campione ecc.).
I metodi presi in considerazione sono: DESeq2 [16], edgeR [17], baySeq [18], EBSeq [19], ShrinkSeq [20], SAMseq [21], NOISeqBIO [22] e voom(+limma) [23].
Nel pre-processamento dei dati si è scelto di applicare la normalizzazione TMM, dove possibile, poiché questo metodo è quello che forniva i risultati più soddisfacenti rispetto a tutte le metriche usate nella valutazione in una recente comparazione di più metodi di normalizzazione [8]. Sempre in questo lavoro si è mostrato che la median-of-ratio si comporta in modo simile alla TMM e che risulta equivalentemente buona rispetto ai criteri di valutazione presi in considerazione. In un altro lavoro si è mostrato che cambiare i metodi di normalizzazione di default proposti dai vari pacchetti per la rilevazione dei geni DE con la normalizzazione TMM non comporta un significativo impatto sui risultati delle analisi [24], perciò in questo elaborato si suppone che

le eventuali differenze rilevate con i vari pacchetti per l'analisi delle differenziale espressione non siano dovute ai diversi metodi di normalizzazione, ma unicamente alle caratteristiche degli algoritmi di detection.
In Tabella 2.1 si riportano in dettaglio le versioni dei pacchetti utilizzati per le analisi e le normalizzazioni utilizzate dai diversi metodi.Per ognuno di questi metodi è stata operata anche la normalizzazione TPM.

Metodo	**Versione**	**Normalizzazioni possibili**	**Normalizzazione consigliate**	**Normalizzazione operata**
DESeq2	1.16.0	median-of-ratio	median-of-ratio	median-of-ratio
edgeR	3.18.0	TMM, UQ, RLE (simile a median-of-ratio), Nessuna (tutti i fattori sono posti a uno)	TMM	TMM
baySeq	2.10.0	Q, TMM, TC	Q	TMM
EBSeq	1.16.0	median-of-ratio	median-of-ratio	median-of-ratio
ShrinkSeq (ShrinkBayes)	2.13.4	TMM	TMM	TMM
SAMseq (samr)	2.0	samr normalization	samr normalization	samr normalization
NOISeqBIO (NOISeq)	2.20.0	RPKM, TMM, UQ, Nessuna	RPKM	TMM
voom (Limma)	3.32.2	TMM	TMM	TMM

Tabella 2.1: Pacchetti software per la rilevazione della differenziale espressione.

Nel capitolo successivo si riportano i dettagli degli otto metodi di analisi di differenziale espressione qui considerati.

Capitolo 3

Metodi per l'analisi della differenziale espressione

3.1 Introduzione generale

In questo capitolo vengono spiegati gli algoritmi e i procedimenti di stima usati dagli otto metodi per l'analisi della differenziale espressione che sono stati valutati e comparati nel presente elaborato. Per ulteriori dettagli si vedano i riferimenti.

Tutti i criteri utilizzano una matrice di conte che contiene il numero di reads che mappano su ciascun trascritto in ciascun campione dell'esperimento. Di tali metodi sette lavorano direttamente sulle conte (DESeq2, edgeR, baySeq, EBSeq, ShrinkSeq, NOISeqBIO e SAMseq), mentre uno trasforma le conte usando poi il pacchetto R limma [25], che è stato sviluppato in origine per l'analisi della differenziale espressione per dati di microarray (voom).

I metodi che lavorano direttamente sulle conte possono essere divisi in:

- parametrici: DESeq2, edgeR, baySeq, EBSeq, ShrinkSeq;
- non parametrici: SAMseq, NOISeqBIO.

I modelli parametrici, a parte ShrinkSeq, usano un modello Binomiale Negativo (NB) per tenere conto della sovra-dispersione, mentre ShrinkSeq permette all'utente di scegliere tra una varietà di distribuzioni tra cui la Binomiale Negativa e la zero-inflated NB, ovvero una distribuzione che tiene conto del gran numero di conte nulle nei campioni. DESeq2 e edgeR prevedono un classico test d'ipotesi mentre gli altri metodi parametrici utilizzano un approccio Bayesiano. I due metodi non parametrici qui valutati (SAMseq e NOISeqBIO) non assumono alcuna distribuzione particolare per i dati. Infine, l'approccio di trasformazione voom (dal pacchetto limma di R) ha lo scopo di

trovare una trasformazione delle conte per renderle più adatte all'analisi con i metodi tradizionali sviluppati per l'analisi di differenziale espressione per i microarray.

Nello studio presente, ci si è focalizzati solo su una comparazione a due gruppi, dato che in pratica è la situazione più frequente. Tuttavia, molti dei metodi valutati supportano anche disegni sperimentali più complessi.
Di seguito sono riportati i singoli metodi nel dettaglio.

3.2 DESeq2

3.2.1 Modello e normalizzazione

La conta delle reads, K_{ij}, per il gene i, con $i = 1, ..., G$, nel campione j, con $j = 1, ..., m$, è descritta con un GLM [26] della famiglia Binomiale Negativa con legame logaritmico:

$$K_{ij} \sim NB(mean = \mu_{ij}, dispersion = \alpha_i). \tag{3.1}$$

La media è considerata come una quantità, q_{ij}, proporzionale alla quantità di frammenti di cDNA per quel gene nel campione, riscalata per un fattore di normalizzazione s_{ij} che tiene conto della diversa profondità di sequenziamento tra i campioni, ovvero:

$$\mu_{ij} = s_{ij} q_{ij}$$

Di default, le costanti di normalizzazione s_{ij} sono considerate costanti all'interno del campione, $s_{ij} = s_j$, e sono stimate con il metodo median-of-ratios. Alternativamente, l'utente può fornire costanti di normalizzazione s_{ij} calcolate usando altri metodi che possono anche differire da gene a gene.

La funzione legame è data da:

$$\log_2 q_{ij} = \sum_r x_{jr} \beta_{ir} \tag{3.2}$$

dove x_{jr} sono gli elementi della matrice di disegno, mentre β_{ir} sono i coefficienti. Nel caso più semplice di una comparazione tra due gruppi, ad esempio trattati e controlli, gli elementi della matrice di disegno indicano se un campione j è trattato o no, e il GLM stimato restituisce coefficienti che indicano la forza dell'espressione del gene e il log_2 dei *fold change* tra trattati e controlli.

3.2.2 Empirical Bayes shrinkage per la stima della dispersione

Un'accurata stima del parametro di dispersione α_i, ovvero della variabilità tra i replicati, è critica per l'analisi inferenziale di differenziale espressione. Per studi con un'ampia numerosità campionaria quest'operazione non è problematica, mentre in studi con bassa numerosità la stima della dispersione per ciascun gene risulta altamente variabile e se usata direttamente può compromettere l'accuratezza del test di differenziale espressione. Una soluzione è quella di condividere l'informazione tra i geni, in particolare DESeq2 assume che geni con valore d'espressione medio simile abbiano dispersione simile. Si assuma che il parametro di dispersione α_i segua una distribuzione log-normale a priori centrata attorno a un trend che dipende dalla media delle conte delle reads normalizzate del gene:

$$\log \alpha_i \sim N(\log \alpha_{tr}(\bar{\mu}_i), \sigma_d^2) \tag{3.3}$$

Qui, α_{tr} è una funzione della media delle conte normalizzate del gene, con

$$\bar{\mu}_i = \frac{1}{m} \sum_j \frac{K_{ij}}{s_{ij}}.$$

La funzione descrive l'aspettativa della dipendenza dalla media della distribuzione a priori. σ_d è l'ampiezza di tale distribuzione, ovvero un iper-parametro che descrive quanto la vera dispersione del singolo gene si disperde attorno al trend. Per la funzione trend, gli autori hanno notato che c'è un andamento sistematico della dispersione in funzione della media [27]:

$$\alpha_{tr}(\bar{\mu}) = \frac{a_1}{\bar{\mu}} + \alpha_0 \tag{3.4}$$

La stima finale della dispersione da questo modello è ottenuta in tre passi:

1. Per prima cosa si usano le conte di ciascun gene separatamente per ottenere una stima gene-wise preliminare della dispersione α_i^{gw} attraverso massima verosimiglianza (punti neri in Figura 3.1).

2. Successivamente si stima il trend della dispersione α_{tr} (linea rossa in Figura 3.1); questo fornisce una stima accurata della dispersione attesa per geni aventi un certo valore d'espressione, ma non rappresenta le deviazioni individuali dei singoli geni dal trend complessivo.

3. Per fare ciò si comprimono le stime della dispersione gene-wise verso i valori predetti dalla curva per ottenere i valori finali della dispersione

(frecce blu). In pratica si combina la verosimiglianza con la distribuzione a priori del trend per ottenere i valori massimi a posteriori (MAP) come stime finali della dispersione.

L'approccio utilizzato è di tipo Bayesiano empirico per cui la forza della compressione dipende da:

- quanto i veri valori della dispersione siano vicini a quelli stimati;
- i gradi di libertà: man mano che la numerosità campionaria aumenta, la forza della compressione diminuisce.

La procedura di compressione aiuta quindi ad evitare potenziali falsi positivi, che possono risultare stimando scorrettamente la dispersione. Tuttavia è possibile che per ragioni biologiche o tecniche alcuni geni abbiano una dispersione straordinariamente alta rispetto agli altri geni anche se hanno un livello di espressione medio simile. In questo caso l'inferenza basata sulle stime compresse della dispersione può portare ad avere dei falsi positivi. Per evitare questo, DESeq2 usa le stime gene-wise invece di quelle compresse quando le precedenti sono più di due deviazioni standard residuali sopra la curva.

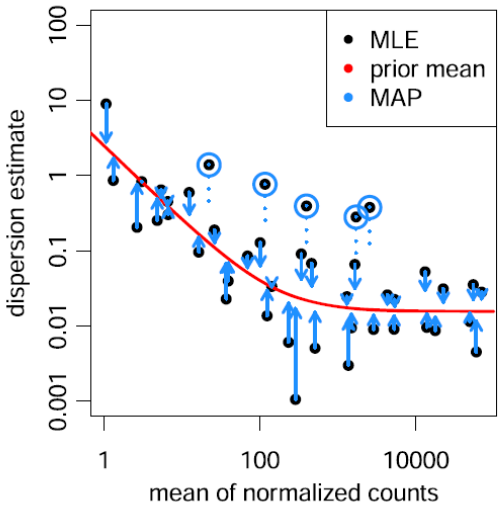


Figura 3.1: Grafico delle stime della dispersione sui valori di espressione medi per il data set Bottomly *et al.* [28] con sei campioni e due gruppi.

Di seguito si riportano in dettaglio i passi che conducono alla stima definitiva della dispersione.

Stime gene-wise della dispersione

Per ottenere una stima della dispersione gene-wise per un gene i, si comincia adattando un GLM Binomiale Negativo alle conte del gene. Questo GLM usa una stima rozza della dispersione ottenuta tramite il metodo dei momenti, basata sulle varianze e sulle medie within-group. Il GLM iniziale è necessario per ottenere un insieme iniziale di valori stimati, $\hat{\mu}_{ij}^0$. Successivamente si massimizza la verosimiglianza aggiustata Cox-Reid [29] della dispersione, condizionata ai valori stimati $\hat{\mu}_{ij}^0$ della stima iniziale, per ottenere le stime gene-wise α_i^{gw}, i.e.,

$$\alpha_i^{gw} = \underset{\alpha}{\operatorname{argmax}}\, \ell_{CR}(\alpha; \boldsymbol{\mu_{i\cdot}^0}, \boldsymbol{K_{i\cdot}})$$

con $\boldsymbol{\mu_{i\cdot}^0}$ il vettore dei valori iniziali stimati per il gene i e $\boldsymbol{K_{i\cdot}}$ il vettore che contiene le conte dei campioni per il gene i,

$$\begin{aligned} \ell_{CR}(\alpha; \boldsymbol{\mu}, \boldsymbol{K}) &= \ell(\alpha) - \frac{1}{2}\log(\det(X^t W X)) \\ \ell(\alpha) &= \sum_j \log f_{NB}(K_j; \mu_j, \alpha) \end{aligned} \tag{3.5}$$

dove $f_{NB}(k; \mu, \alpha)$ è la densità di probabilità della distribuzione Binomiale Negativa con media μ e dispersione α, mentre il secondo termine rappresenta l'aggiustamento della distorsione di Cox-Reid. Tale aggiustamento corregge la distorsione negativa della stima della dispersione dovuta al fatto che si è usata la massima verosimiglianza per ottenere le stime $\hat{\mu}_{ij}^0$ (analogo alla correzione di Bessel nella solita formula della varianza campionaria [30]). È ottenuta a partire dall'informazione di Fisher per i valori stimati, che qui è calcolata come $\det(X^t W X)$, dove W è la matrice diagonale dei pesi ottenuta tramite l'algoritmo dei minimi quadrati pesati iterati standard. Dato che la funzione legame è $\mathrm{g}(\mu) = \log(\mu)$ e la sua funzione di varianza è $V(\mu; \alpha) = \mu + \alpha\mu^2$, gli elementi della matrice diagonale W_i sono dati da:

$$w_{jj} = \frac{1}{\mathrm{g}'(\mu_j)^2 V(\mu_j)} = \frac{1}{\frac{1}{\mu_j} + \alpha}.$$

Trend della dispersione

Una curva parametrica della forma $\alpha_{tr}(\bar{\mu}) = \frac{a_1}{\bar{\mu}} + \alpha_0$ è stimata regredendo le stime della dispersione gene-wise α_i^{gw} sulle medie delle conte normalizzate, $\bar{\mu}_i$. La distribuzione campionaria delle stime della dispersione gene-wise attorno al vero valore α_i può essere fortemente distorta, e perciò non si usa la

regressione ai minimi quadrati ordinaria, ma piuttosto una regressione GLM della famiglia gamma. In aggiunta, dispersioni outlier possono distorcere la stima e quindi si usa una strategia per escludere tali outliers.

Più in dettaglio, gli iper-parametri a_1 e α_0 di $\alpha_{tr}(\bar{\mu}) = \frac{a_1}{\bar{\mu}} + \alpha_0$ sono ottenuti adattando un GLM della famiglia gamma iterativamente. A ciascuna iterazione, i geni con un rapporto tra la dispersione e i valori stimati fuori dal range $[10^{-4}, 15]$ sono lasciti fuori fino a che la somma dei *log fold change* (LFC) al quadrato dei nuovi coefficienti sui vecchi coefficienti è meno di 10^{-6} [27]. La parametrizzazione (3.4) si basa su quanto riportato da alcuni articoli riguardo la dipendenza decrescente tra la dispersione e la media in molti data set [31, 32, 33, 34, 9]. Bisogna però fare attenzione a distinguere la vera dipendenza sottostante dagli effetti delle distorsioni sulle stime, che possono dare l'illusione di una dipendenza della dispersione dalla media. Consideriamo una variabile casuale distribuita come una binomiale negativa con media μ e dispersione α. La sua varianza $v = \mu + \alpha\mu^2$ ha due componenti, $v = v_P + v_D$, la componente Poisson $v_P = \mu$ indipendente da α, e la componente di sovradispersione $v_D = \alpha\mu^2$. Quando μ è piccola, $\mu \lesssim \frac{1}{\alpha}$, la componente Poisson domina, nel senso che $\frac{v_P}{v_D} = \frac{1}{(\alpha\mu)} \gtrsim 1$, e i dati osservati forniscono poca informazione sul valore di α. Perciò la varianza campionaria di uno stimatore per α sarà grande quando $\mu \lesssim \frac{1}{\alpha}$ e questo porta alla comparsa della distorsione. Questo fenomeno può far sorgere l'apparente dipendenza di α da μ.

Stima di σ_d^2

Come osservato in [33], tipicamente per i dati di RNA-Seq è possibile adattare un'a priori log-normale alla distribuzione della dispersione osservata. Per risolvere le difficoltà computazionali legate al dover lavorare con un'a priori non coniugata, gli autori del metodo si sono basati sul seguente concetto: i residui logaritmici del trend stimato, $\log \alpha_i^{gw} - \log \alpha_{tr}(\bar{\mu}_i)$, derivano da due contributi, cioè la diffusione della vera dispersione logaritmica attorno al trend , data dall'a priori con varianza σ_d^2, e dalla distribuzione campionaria dei logaritmi degli stimatori della dispersione, con varianza σ_{lde}^2. La varianza campionaria di uno stimatore di dispersione è approssimativamente un χ^2 scalato con $m-p$ gradi di libertà, con m il numero dei campioni e p il numero dei coefficienti. La varianza del logaritmo di una variabile casuale distribuita secondo un χ_f^2 è data dalla funzione trigamma ψ_1, $Var(\log X^2) = \psi_1(f/2)$ con $X^2 \sim \chi_f^2$ [35]. Perciò, $\sigma_{lde}^2 \approx \psi_1((m-p)/2)$, ovvero la varianza campionaria del logaritmo di una varianza o di uno stimatore di dispersione è approssimativamente costante tra i geni e dipende solo dai gradi di libertà del modello.

In questo modo la varianza dell'a priori σ_d^2 è ottenuta sottraendo la varianza campionaria attesa da una stima della varianza dei residui logaritmici, s_{lr}^2:

$$\sigma_d^2 = \max\left\{s_{lr}^2 - \psi_1((m-p)/2), 0.25\right\}.$$

La varianza dell'a priori σ_d^2 è limitata da una soglia minima di 0.25 così che le stime della dispersione non siano completamente compresse verso $\alpha_{tr}(\bar{\mu}_i)$ se la varianza dei residui logaritmici è minore della varianza campionaria attesa. Per evitare l'inflazione di σ_d^2 causata da outliers (ovvero geni non ben catturati dall'a priori), si usa uno stimatore robusto per la deviazione standard s_{lr}^2 dei residui logaritmici,

$$s_{lr} = \underset{i}{\text{mad}}(\log \alpha_i^{gw} - \log \alpha_{tr}(\bar{\mu}_i)) \tag{3.6}$$

dove mad sta per la deviazione assoluta dalla mediana, divisa per il fattore di scala $\Phi^{-1}(3/4)$.

Stima finale della dispersione

Viene formata un'a posteriori logaritmica per la dispersione a partire dalla verosimiglianza logaritmica aggiustata di Cox-Reid (3.5) e dall'a priori logaritmica (3.3), e si usa il suo massimo (MAP) come stima finale della dispersione

$$\alpha_i^{MAP} = \underset{\alpha}{\text{argmax}}(\ell_{CR}(\alpha; \boldsymbol{\mu_{i\cdot}^0}, \boldsymbol{K_{i\cdot}}) + \Lambda_i(\alpha)) \tag{3.7}$$

dove $\Lambda_i(\alpha) = \frac{-(\log\alpha - \log\alpha_{tr}(\bar{\mu}_i))^2}{(2\sigma_d^2)}$ è, a meno di una costante additiva, il logaritmo della densità dell'a priori (3.3). Di nuovo, viene usano un algoritmo di backtracking line search per operare l'ottimizzazione.

3.2.3 Test d'ipotesi per la differenziale espressione

Una difficoltà comune nell'analisi di dati di RNA-Seq è l'elevata varianza delle stime dei LFC per i geni con conte basse. DESeq2 risolve questo problema comprimendo le stime dei LFC verso zero. Di nuovo si applica una procedura Bayesiana empirica: per prima cosa si adatta un GLM ordinario per ottenere le stime di massima verosimiglianza (MLEs) per i LFCs (in questa fase vengono usate le stime finali della dispersione per ciascun gene) e poi si stima una distribuzione normale centrata in zero per la distribuzione osservata dei MLEs su tutti i geni. In particolare, viene postulata un'a priori normale centrata in zero per i coefficienti β_{ir} del modello (3.2)

che rappresentano i LFCs (tipicamente tutti i coefficienti eccetto l'intercetta β_{i0}):

$$\beta_{ir} \sim N(0, \sigma_r^2).$$

Questa distribuzione viene usata come a priori sui LFC per un secondo round di stima GLM e infine si considerano le stime MAP come stime finali dei LFC. Dopo che i GLM sono stimati per ciascun gene, si testa se ciascun coefficiente del modello differisce significativamente da zero. DESeq2 usa un test di Wald: la stima compressa del LFC β_{ir} è divisa per il suo standar error stimato $SE(\beta_{ir})$, il che risulta in una statistica z che è comparata con la distribuzione Normale standard. Gli standard errors stimati sono le radici quadrate degli elementi sulla diagonale della matrice di covarianza stimata, Σ_i , ovvero $SE(\beta_{ir}) = \sqrt{\Sigma_{i,rr}}$ dove, per un GLM con a priori normale sui coefficienti, la matrice di covarianza dei coefficienti risulta [36, 37]:

$$\Sigma_i = Cov(\boldsymbol{\beta_i}) = (X^t W X + \boldsymbol{\lambda} I)^{-1}(X^t W X)(X^t W X + \boldsymbol{\lambda} I)^{-1}$$

dove $\boldsymbol{\lambda}$ è un vettore i cui elementi sono $\lambda_r = \frac{1}{\sigma_r^2}$.
I p-value del test di Wald di un sottoinsieme di geni che passano una fase di filtraggio indipendente sono aggiustati usando la procedura di Benjamini Hochberg [4].

3.2.4 Filtraggio indipendente

DESeq2 usa la media dell'espressione di ciascun gene tra tutti i campioni come criterio di filtraggio e omette tutti i geni la cui media delle conte normalizzate è sotto una certa soglia derivata dall'aggiustamento per test multipli; di default tale soglia è scelta per massimizzare il numero di geni trovati in base all'FDR specificato dall'utente. Il filtraggio riduce la perdita di potenza dovuta all'aggiustamento per test multipli e non compromette la distribuzione della statistica test poichè, sotto l'ipotesi nulla, questa è marginalmente indipendente dalla statistica di filtraggio [3].

3.3 edgeR

3.3.1 Modello

Per semplicità di notazione consideriamo inizialmente un singolo gene. Sia K_{rj} la conta osservata per la classe r e per il campione j, per un particolare gene i. Assumiamo una comparazione a due gruppi per cui $r = 1, 2$ mentre

$j = 1, ..., m$. Assumiamo una distribuzione Binomiale Negativa per le conte K_{rj} in particolare:

$$K_{rj} \sim NB(\mu_{rj}, \alpha)$$

dove α è la dispersione. Si ha che $E(K_{rj}) = \mu_{rj}$ e $Var(K_{rj}) = \mu_{rj}(1 + \mu_{rj}\alpha)$. Sia q_r la vera abbondanza relativa di un certo trascritto nell'RNA di classe r. Allora $\mu_{rj} = s_{rj}q_r$ dove s_{rj} è la dimensione della libreria per il *j-esimo* campione. Per constatare una differenza nell'abbondanza relativa di un trascritto tra le due condizioni, viene effettuato un test la cui ipotesi nulla è $H_0 : q_1 = q_2$ contro l'alternativa $H_1 : q_1 \neq q_2$ e questo è ripetuto per ciascun trascritto.

3.3.2 Stima della dispersione comune

Le opzioni date dal modello sono due, una dispersione comune o una dispersione gene-specifica. La verosimiglianza condizionata per un singolo trascritto è ottenuta condizionandosi alle somme delle conte di ciascuna classe, dato che la somma di due variabili casuali NB identicamente distribuite è ancora una NB. Il condizionamento ha l'effetto di rimuovere il parametro di disturbo q, e ciò è una generalizzazione della massima verosimiglianza ristretta (REML). Se la dimensione della libreria s_{rj} è uguale all'interno di ciascuna classe, la log-verosimiglianza condizionata di α per un singolo trascritto, dato $z_r = \sum_{j=1}^{m_r} K_{rj}$, è:

$$\ell_i(\alpha) = \sum_{r=1}^{2} \Big[\sum_{j=1}^{m_r} \log\Gamma(k_{rj} + \alpha^{-1}) + \log\Gamma(m_r\alpha^{-1}) - \log\Gamma(z_r + m_r\alpha^{-1}) - m_r \log\Gamma(\alpha^{-1}) \Big] \tag{3.8}$$

Lo stimatore della dispersione comune massimizza la verosimiglianza comune $\ell_C(\alpha) = \sum_{i=1}^{G} \ell_i(\alpha)$ dove G è il numero di trascritti.

Nelle situazioni reali in cui le dimensioni delle librerie non sono uguali, le conte non sono identicamente distribuite, e l'argomento condizionante non vale esattamente. Gli autori usano la *quantile adjustment* per aggiustare le conte osservate in alto o in basso a seconda che la corrispondente dimensione della libreria sia superiore o inferiore alla media geometrica (chiamato qCML che sta per *quantile adjusted conditional maximum likelihood*). Questa trasformazione genera delle pseudo-conte che sono approssimativamente identicamente distribuite e che possono essere inserite nell'equazione (3.8). Le log-verosimiglianze condizionate per ciascun trascritto sono sommate tra loro e il tutto viene massimizzato rispetto ad α, al fine di ottenere una stima comune.

3.3.3 Stima moderata della dispersione attraverso la verosimiglianza pesata

In campioni molto piccoli l'assunzione di dispersione comune offre una stima buona rispetto alle stime tag-wise che sono poco stabili. Tuttavia non è generalmente vero che tutti i trascritti hanno la medesima dispersione, il che suggerisce che l'inferenza può essere migliorata. Invece di forzare i trascritti ad avere una dispersione comune, gli autori hanno proposto di comprimere la dispersione tag-wise (denotata con α_i, con i che denota i trascritti) verso il valore corrispondente alla dispersione comune α. Per fare ciò viene adoperata una verosimiglianza pesata e i pesi vengono scelti in modo da approssimare una soluzione empirica Bayesiana, che non è direttamente realizzabile a causa del fatto che la distribuzione Binomiale Negativa non appartiene alla famiglia esponenziale e non esistono a priori coniugate per α. La log-verosimiglianza condizionata pesata (WL) per α_i è definita come una combinazione pesata di verosimiglianze individuali e comuni:

$$\mathrm{WL}(\alpha_i) = \ell_i(\alpha_i) + w\,\ell_C(\alpha_i) \tag{3.9}$$

dove w è il peso dato alla verosimiglianza comune [38]. In WL la verosimiglianza comune gioca lo stesso ruolo che l'a priori per α_i giocherebbe in un modello gerarchico Bayesiano, con w la precisione dell'a priori. Se $w = 0$ in (3.9), allora otteniamo le stime tag-wise qCML. D'altra parte se si sceglie w sufficientemente grande, il contributo da qualunque log-verosimiglianza individuale è contrastato dalla verosimiglianza comune e il risultato è una dispersione comune. Tra questi due estremi si ha uno schema di stima dove le stime tag-wise sono da qualche parte tra le stime individuali e quella comune.

3.3.4 Selezione di w in modo da approssimare una regola Bayesiana empirica approssimata

Si vuole selezionare un w appropriato in modo che la stima sia adattiva. Se l'evidenza suggerisce che le dispersioni non sono molto diverse tra loro, w dovrebbe essere scelto abbastanza grande da spingere tutti i trascritti ad essere compressi pesantemente verso la stima comune. Tuttavia se c'è evidenza di dispersioni variabili, w dovrebbe essere selezionato in modo da comprimere poco. Per capire la strategia di selezione di w, supponiamo che gli stimatori individuali qCML, $\hat{\alpha}_i$, siano normalmente distribuiti con media α_i e varianza nota τ_i^2, e assumiamo un modello gerarchico:

$$\hat{\alpha}_i|\alpha_i \sim N(\alpha_i, \tau_i^2),\ \alpha_i \sim N(\alpha_0, \tau_0^2),\ i = 1, ..., G.$$

Lo stimatore Bayesiano della media a posteriori di α_i sarebbe:

$$\hat{\alpha}_i^B = E(\alpha_i|\hat{\alpha}_i) = \frac{\frac{\hat{\alpha}_i}{\tau_i^2} + \frac{\alpha_0}{\tau_0^2}}{\frac{1}{\tau_i^2} + \frac{1}{\tau_0^2}}$$

In pratica, gli iperparamentri α_0 e τ_0^2 sono ignoti, ma possono essere stimati dalla distribuzione marginale di α_i per ottenere un approccio Bayesiano empirico. La strategia è scegliere w in modo che WL coincida con EB. Sotto questo idealistico modello normale, lo stimatore di massima WL è:

$$\hat{\alpha}_i^{WL} = \frac{\frac{\hat{\alpha}_i}{\tau_i^2} + w\sum_{i=1}^G \frac{\hat{\alpha}_i}{\tau_i^2}}{\frac{1}{\tau_i^2} + w\sum_{i=1}^G \frac{1}{\tau_i^2}}$$

Questo coincide con $\hat{\alpha}_i^B$ se α_0 è uguale allo stimatore della dispersione comune

$$\alpha_0 = \hat{\alpha}_0 = \frac{\sum_{i=1}^G \frac{\hat{\alpha}_i}{\tau_i^2}}{\sum_{i=1}^G \frac{1}{\tau_i^2}}$$

e

$$\frac{1}{w} = \sum_{i=1}^G \frac{\tau_0^2}{\tau_i^2} \tag{3.10}$$

Rimane solo trovare uno stimatore per τ_0^2. Sotto il modello normale, $\frac{(\hat{\alpha}_i - \alpha_0)^2}{(\tau_i^2 + \tau_0^2)} \sim \chi_1^2$, così uno stimatore consistente per τ_0 si ottiene risolvendo

$$\sum_{i=1}^G \left[\frac{(\hat{\alpha}_i - \alpha_0)^2}{(\tau_i^2 + \tau_0^2)} - 1\right] = 0 \tag{3.11}$$

Questa regola per scegliere w non è direttamente disponibile, perché gli stimatori qCML $\hat{\alpha}_i$ sono lontani dall'essere normalmente distribuiti e non hanno varianza nota. Per superare questa difficoltà, sfruttiamo il fatto che la statistica score (ovvero la derivata della log-verosimiglianza) converge alla normalità più rapidamente di quanto fanno gli stimatori di massima verosimiglianza. Inoltre l'equazione di stima (3.11) può essere scritta in termini di score di verosimiglianza $S_i(\alpha) = \partial\ell_i(\alpha)/\partial\alpha$ e dell'informazione attesa $I_i = E(J_i)$, $J_i = -\partial^2\ell_i(\alpha)/\partial\alpha^2$, funzioni di α_i.

L'algoritmo di stima risulta:

1. Si trova lo stimatore per la dispersione comune $\hat{\alpha}_0$, che massimizza ℓ_C.
2. Si calcola $S_i(\hat{\alpha}_0)$ e $I_i(\hat{\alpha}_0)$ per ciascun trascritto.

3. Si stima τ_0 risolvendo

$$\sum_{i=1}^{G} \left[\frac{S_i^2}{I_i(1 + I_i\tau_0^2)} - 1 \right] = 0$$

   Se $\sum S_i^2/I_i < G$ allora $\tau_0 = 0$.

4. Si pone

$$\frac{1}{w} = \tau_0^2 \sum_{i=1}^{G} I_i$$

5. Si ottengono gli stimatori mi massima verosimiglianza pesati $\tilde{\alpha}_i$ massimizzando $\mathrm{WL}(\alpha_i)$.

Questo algoritmo è in accordo con (3.10) e (3.11), ma è più generalmente applicabile perché usa unicamente quantità valutate in $\hat{\alpha}_0$ per stimare τ_0. Le informazioni attese I_i sono difficili da calcolare direttamente, ma possono essere approssimate bene usando le informazioni osservate J_i. Per ogni valore dato di α_i, I_i dovrebbe essere quasi direttamente proporzionale alle conte totali $z_1 + z_2$. Quindi si opera una regressione lineare con intercetta zero di J_i sul totale delle pseudo-conte , e si usano i valori predetti per rappresentare I_i. L'algoritmo può essere applicato a qualunque trasformazione di α. Gli autori hanno trovato conveniente implementare l'algoritmo in scala $\delta = \alpha/(\alpha + 1)$ perché δ assume valori strettamente limitati.

Se le dispersioni sono uguali tra loro (tutti gli $\alpha_i = \alpha$) allora $E(S_i^2) = I_i$, in questo modo τ_0^2 sarà stimato vicino a zero e quindi w sarà grande. Se, tuttavia, le dispersioni sono differenti tra loro, allora $E(S_i^2)$ non sarà nullo e S_i^2 sarà più grande di I_i in media, e questo forza τ_0^2 ad essere più grande di zero e meno peso sarà dato alla verosimiglianza comune. Il fatto che $E(S_i^2) = I_i$ sotto l'ipotesi nulla è un risultato esatto, che non si affida sulla normalità asintotica, e che assicura che l'algoritmo abbia un buon comportamento anche quando la dimensione campionaria è piccola.

3.3.5 Test statistico per la differenziale espressione

Per testare la differenza nell'espressione tra le due condizioni, si usa un test di Wald [39] che semplicemente divide $\hat{q}_2 - \hat{q}_1$ per il suo standard error stimato. È però possibile anche usare il test esatto sviluppato dagli autori del metodo [40]. Questo test esatto si basa sul metodo di *quantile adjustment*. Usando le pseudo-conte, si sfrutta il fatto che la somma di due variabili casuali NB identicamente distribuite è ancora una NB. Condizionandosi alla pseudo-somma totale (una variabile casuale NB), si può calcolare la probabilità di

osservare conte tanto o più estreme rispetto a quelle osservate, il che da come risultato un p-value esatto.

3.4 baySeq

Per baySeq, l'utente definisce un insieme di modelli, ciascuno dei quali è essenzialmente una partizione dei campioni in gruppi, dove si assume che i campioni nello stesso gruppo condividano gli stessi parametri della distribuzione sottostante. Con un'impostazione empirica Bayesiana, baySeq stima la probabilità a posteriori di ciascun modello per ciascun gene nel data set. Informazioni provenienti dall'intero insieme di geni sono usate per formare una distribuzione a priori empirica per i parametri del modello NB.

Nel formare un insieme di modelli per i dati, si considerano quali modelli sono biologicamente probabili. Nel caso più semplice di una comparazione a due gruppi, si hanno dati di conta provenienti da alcuni campioni sia nella condizione A che nella condizione B. Si supponga di avere m_r replicati biologici per ciascuna classe $A_1, ..., A_{m_r}$ e $B_1, ..., B_{m_r}$. Nella maggior parte dei casi, è ragionevole supporre che alcune reads non siano affette dalle condizioni sperimentali A e B. I valori di conta per ciascun campione in queste reads condivideranno gli stessi parametri sottostanti. Tuttavia, alcune reads possono essere influenzate delle differenti condizioni sperimentali A e B e per queste reads i dati dai campioni $A_1, ..., A_{m_r}$ condivideranno lo stesso insieme di parametri sottostanti, e lo stesso accadrà per i dati dai campioni $B_1, ..., B_{m_r}$, ma questi due insiemi di parametri non saranno uguali. Si possono così trattare i modelli come insiemi di campioni non sovrapposti. Il primo modello, quello di non differenziale espressione, è così definito dall'insieme di campioni $\{A_1, ..., A_{m_r}, B_1, ..., B_{m_r}\}$. Il secondo modello, quello di differenziale espressione tra la condizione A e B è definito dagli insiemi $\{A_1, ..., A_{m_r}\}$ e $\{B_1, ..., B_{m_r}\}$. La stessa impostazione è estendibile a disegni più complessi.

3.4.1 Modello

Si supponga di avere dati di conta provenienti da un insieme di m campioni $\mathcal{A} = \{A_1, ..., A_m\}$, tali che i dati osservati per un particolare gene, i, sono rappresentati da $(K_{i1}, ..., K_{im})$ dove K_{ij} è la conta per un particolare gene i nel campione j. Per ciascun campione A_j, si ha anche il fattore di normalizzazione della dimensione della libreria s_j.

Per ciascun gene si possono considerare i dati come

$$D_i = \{(K_{i1}, ..., K_{im}), (s_1, ..., s_m)\}$$

Ora si consideri un certo modello M su questi dati, definito dagli insiemi $\{E_1, ..., E_b\}$. Se, in questo modello, i campioni A_j e A_h sono nello stesso insieme E_q, allora questi avranno gli stessi parametri della distribuzione sottostante θ_q. Si può definire l'insieme $U = \{\theta_1, ..., \theta_b\}$. Per semplicità di notazione, si definiscono i dati associati all'insieme E_q come $D_{qi} = \{(K_{ij} : A_j \in E_q), (s_j : A_j \in E_q)\}$. Dato un modello M per i dati, allora la quantità d'interesse per ciascun gene i è la probabilità a posteriori del modello M dati i dati D_i, che è

$$\mathbb{P}(M|D_i) = \frac{\mathbb{P}(D_i|M)\mathbb{P}(M)}{\mathbb{P}(D_i)} \tag{3.12}$$

Per prima cosa, si può cercare di calcolare $\mathbb{P}(D_i|M)$ considerando la verosimiglianza marginale

$$\mathbb{P}(D_i|M) = \int \mathbb{P}(D_i|U, M)\mathbb{P}(U|M)\mathrm{d}U \tag{3.13}$$

3.4.2 Approssimazione di $\mathbb{P}(D_i|M)$

Ci sono molte possibili distribuzioni che possono essere usate per $D_i|U, M$ e per $U|M$. Per tenere conto dell'extra variabilità introdotta dai replicati biologici si può assumere che i dati provengano da una Binomiale Negativa. Nel caso in cui le dimensioni delle librerie siano uguali, sotto l'assunzione di distribuzione Binomiale Negativa, è possibile sviluppare un test esatto per la verosimiglianza di osservare i dati data la non differenziale espressione. Il problema delle dimensioni delle librerie non uguali può essere risolto usando metodi numerici in un contesto empirico Bayesiano, che permettono di mantenere i dati reali, usando la dimensione delle librerie come un fattore di normalizzazione.

Si consideri un campione A_j appartenente all'insieme E_q con dimensione della libreria s_j. Si assuma ora che le conte K_{ij} del gene i in questo campione siano distribuite come una Binomiale Negativa, con media $\mu_q s_j$ e dispersione α_q, dove $\theta_q = (\mu_q, \alpha_q)$. La parametrizzazione può essere così definita

$$\mathbb{P}(K_{ij}; s_j, \alpha_q, \mu_q) = \frac{\Gamma(K_{ij} + \alpha_q^{-1})}{\Gamma(\alpha_q^{-1})K_{ij}!} \left(\frac{1}{1 + s_j\mu_q\alpha_q}\right)^{\alpha_q^{-1}} \left(\frac{s_j\mu_q}{\alpha_q^{-1} + s_j\mu_q}\right)^{K_{ij}}$$

Sfortunatamente non ci sono coniugate che possono essere applicate. Tuttavia, se è possibile definire una distribuzione empirica per U allora si può stimare $\mathbb{P}(D_i|M)$ numericamente.

Per prima cosa si assume che i $\theta_q \in U$ siano indipendenti tra loro. Allora

$$\mathbb{P}(D_i|M) = \int \mathbb{P}(D_i|U,M)P(U|M)\mathrm{d}U = \prod_q \int \mathbb{P}(D_{qi}|\theta_q)\mathbb{P}(\theta_q)\mathrm{d}\theta_q$$

Questa assunzione riduce la dimensionalità dell'integrale e così migliora l'accuratezza dell'approssimazione numerica dell'integrale.

Successivamente si suppone che per ciascun $\theta_q \in U$ si abbia un insieme di valori Θ_q che sono campionati dalla distribuzione di θ_q. Allora si può derivare l'approssimazione

$$\begin{aligned}\mathbb{P}(D_i|M) \approx \prod_q \frac{1}{|\Theta_q|} \sum_{\Theta_q} \Big[\prod_{\{j:A_j \in E_q\}} \frac{\Gamma(K_{ij}+\alpha_q^{-1})}{\Gamma(\alpha_q^{-1})K_{ij}!} \left(\frac{1}{1+s_j\mu_q\alpha_q}\right)^{\alpha_q^{-1}} \\ \left(\frac{s_j\mu_q}{\alpha_q^{-1}+s_j\mu_q}\right)^{K_{ij}} \Big]\end{aligned} \tag{3.14}$$

Il compito rimane ora quello di derivare l'insieme Θ_q dai dati.

3.4.3 Distribuzione derivata empiricamente di *U*

Si può derivare una distribuzione empirica di U esaminando l'intero data set. Per ciascun insieme di campioni E_q, si vorrebbero trovare delle stime per la medie e la dispersione della distribuzione sottostante i dati provenienti da un singolo gene, D_{qi}. La difficoltà principale sta nello stimare propriamente la dispersione.

Per esempio, si supponga che i dati da un certo gene mostrino una differenziale espressione autentica. Se il modello che si sta testando assume che non ci sia differenziale espressione, allora per questo gene la dispersione sarà sostanzialmente sovrastimata. Dato che non si sa in anticipo quale gene è veramente differenzialmente espresso e quale no, bisogna considerare la struttura di replicazione dei dati in modo da stimare propriamente la dispersione. Si definisce la struttura di replicazione considerando gli insiemi $\{F_1, ..., F_l\}$ dove $j, h \in F_t$ se e solo se A_j è un replicato di A_h.

Considerando questa struttura per i dati, si può stimare la dispersione dei dati di un gene D_i attraverso il metodo della quasi-verosimiglianza [41]. Per prima cosa si definisce $\hat{\mu}_{ti} = E\left[\frac{K_{ij}}{s_j} | \, j \in F_t\right]$, e poi si sceglie α_i in modo che

$$2\sum_t \sum_{j\in F_t} \left\{ K_{ij} \log\left[\frac{K_{ij}}{s_j\hat{\mu}_{ti}}\right] - (K_{ij}+\alpha_i^{-1}) \log\left[\frac{K_{ij}+\alpha_i^{-1}}{s_j\hat{\mu}_{ti}+\alpha_i^{-1}}\right] \right\} = m-1 \tag{3.15}$$

Prendendo questo valore come α_i è possibile ristimare i valori $\hat{\mu}_{ij}$ con il metodo della massima verosimiglianza, scegliendo i valori per $\hat{\mu}_{ij}$ che massimizzano le verosimiglianze

$$\mathbb{P}(\{K_{ij} : j \in F_t\}; s_j : j \in F_t, \alpha_i, \hat{\mu}_{ti}) = \prod_{j \in F_t} \frac{\Gamma(K_{ij} + \alpha_i^{-1})}{\Gamma(\alpha_i^{-1}) K_{ij}!} \left(\frac{1}{1 + s_j \hat{\mu}_{ti} \alpha_i} \right)^{\alpha_i^{-1}} \left(\frac{s_j \hat{\mu}_{ti}}{\alpha_i^{-1} + s_j \hat{\mu}_{ti}} \right)^{K_{ij}}$$

per ciascun t.

Si procede a iterare le stime di α_i e di $\hat{\mu}_{ij}$ fino a convergenza.

Questa procedura fornisce il valore di α_i. Bisogna poi stimare la media della distribuzione sottostante i dati D_{qi}, cioè per l'insieme di campioni in E_q, cosa che può essere fatta facilmente fissando il valore acquisito per α_i e stimando la media μ_{qi} con il metodo della massima verosimiglianza, scegliendo il valore di μ_{qi} che massimizza la verosimiglianza

$$\mathbb{P}(D_{qi}, \alpha_i, \mu_{qi}) = \prod_{\{j : A_j \in E_q\}} \frac{\Gamma(K_{ij} + \alpha_i^{-1})}{\Gamma(\alpha_i^{-1}) K_{ij}!} \left(\frac{1}{1 + s_j \mu_{qi} \alpha_i} \right)^{\alpha_i^{-1}} \left(\frac{s_j \mu_{qi}}{\alpha_i^{-1} + s_j \mu_{qi}} \right)^{K_{ij}}$$

per ciascun q.

Si può così formare l'insieme $\Theta_q = \{(\mu_{qi}, \alpha_i)\}$ ripetendo questa procedura per tutti i q, e si può poi calcolare $\mathbb{P}(D_i|M)$ a partire dall'Equazione (3.14).

Questo metodo di stima della dispersione assume che la dispersione di un gene sia costante per insiemi differenti di campioni. Dove ci si aspetta che la dispersione sia sostanzialmente differente tra insiemi di replicati, ci potrebbero essere vantaggi nello stimare le dispersioni individualmente per ciascuno dei differenti insiemi di campioni in ciascun modello, pur considerando la struttura di replicazione all'interno di questi insiemi. Questo può essere facilmente fatto restringendo i dati (e la corrispondente struttura di replicazione) a D_{qi} quando si stima la dispersione nell'Equazione (3.15). Gli autori hanno verificato che non ci sono sostanziali differenze tra questi approcci.

3.4.4 Stima delle probabilità a priori di ciascun modello

Sono disponibili varie opzioni quando si considerano le probabilità a priori di ciascun modello $\mathbb{P}(M)$ richieste nell'Equazione (3.12). Nel caso in cui sia possibile, stimarle da altre fonti risulta la soluzione ottima. Tuttavia, in molti casi non è possibile fornire una stima ragionevole delle probabilità a priori. In questi casi gli autori hanno proposto una modifica del metodo suggerito da

Smyth [25] per la stima delle proporzioni dei geni differenzialmente espressi in esperimenti di microarray.

Si comincia scegliendo (basandosi idealmente su una conoscenza a priori del modello) alcuni valori p da usare come probabilità a priori per il modello M al fine di stimare la probabilità a posteriori $\mathbb{P}(M|D_i)$ per l'*i-esimo* gene. Ad ogni iterazione si può derivare una nuova stima

$$p^{'} = E[\mathbb{P}(M|D_i)]_i$$

per la probabilità a priori del modello M. Iterando fino a convergenza, si acquisiscono le stime delle probabilità a priori per ciascun modello. In pratica, si è visto che la scelta iniziale di p non ha un effetto sostanziale sui valori ai quali l'algoritmo converge.

3.4.5 Fattore di scala $\mathbb{P}(\boldsymbol{D_i})$

Infine, si considera il fattore di scala $\mathbb{P}(D_i)$ nell'Equazione (3.12). Dato che il numero di possibili modelli M su $\mathcal{A}$ è finito, sebbene potenzialmente grande, il fattore di scala $\mathbb{P}(D_i)$ può essere determinato sommando su tutti i possibili M, date appropriate a priori $\mathbb{P}(M)$. In pratica, il numero di modelli può essere limitato considerando solamente quelli che sono biologicamente plausibili, o imponendo alcune distribuzioni sul numero degli insiemi in M in maniera simile all'approccio di Lönnstedt et al. [42] per l'analisi della varianza nei dati di microarrray.

3.5 EBSeq

EBSeq è un approccio empirico Bayesiano che può essere usato per identificare i geni e le isoforme differenzialmente espressi in un esperimento di RNA-Seq.

Prima dell'inferenza per l'identificazione delle isoforme differenzialmente espresse, l'espressione di ciascuna isoforma deve essere stimata attraverso l'allineamento delle reads. Per geni con una singola isoforma, questo procedimento è piuttosto semplice in quanto tutte le reads che mappano su quel gene sono usate per stimare l'espressione dell'isoforma. Per i geni con isoforme multiple, la stima dell'espressione è più complessa, poiché le reads che mappano sugli esoni comuni a più isoforme, devono essere allocate in modo consistente con il livello di espressione di ciascuna isoforma, cosa che determina vari stadi di incertezza in tali stime di espressione. È importante sottolineare, per l'inferenza a livello di isoforma, che EBSeq adatta diretta-

mente l'incertezza della stima dell'espressione delle isoforme modellando la variabilità differenziale osservata nei distinti gruppi di isoforme.

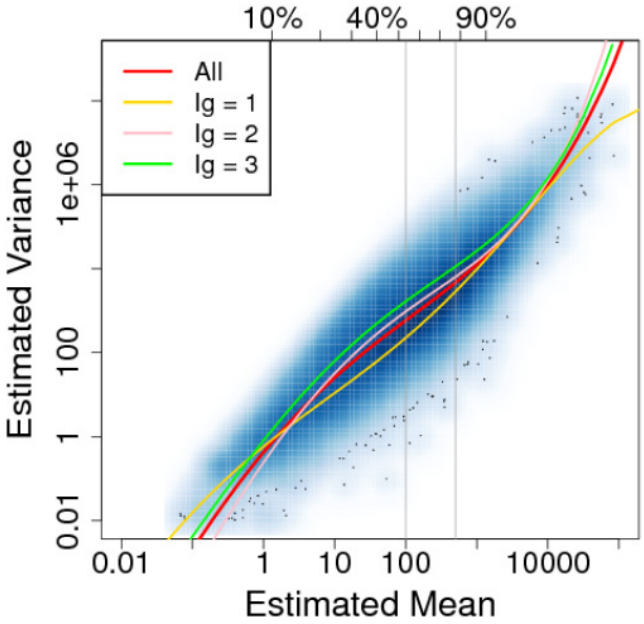


Figura 3.2: Varianza empirica contro media per ciascuna isoforma per i dati in [19]. Una spline che modella tutte le isoforme è raffigurata in rosso, mentre slines che modellano le isoforme nei gruppi $I_g = 1$, $I_g = 2$ e $I_g = 3$ sono rappresentate rispettivamente in giallo, rosa e verde.

Si consideri la Figura 3.2, dove si è fatto uno scatter plot della varianza contro la media per tutte le isoforme usando i dati di RNA-Seq provenienti da [19]. Viene mostrato l'adattamento di tre sottogruppi, definiti dal numero di isoforme appartenenti al gene genitore. Un'isoforma di un gene i è assegnata al gruppo $I_g = k$ con $k = 1, 2, 3$, se il numero totale delle isoforme del gene i è k (il gruppo $I_g = 3$ contiene tutte quelle isoforme provenienti da geni aventi tre o più isoforme). Come mostrato in Figura 3.2, rispetto alla curva totale c'è una diminuzione della variabilità nel gruppo $I_g = 1$, ed un aumento della variabilità negli altri, a causa del relativo aumento dell'incertezza inerente alla stima del livello di espressione delle isoforme quando sono presenti più isoforme di un determinato gene.

Se non si tiene conto di questo, si verifica una riduzione della potenza nell'identificare le isoforme del gruppo $I_g = 1$ (dato che le vere varianze in questo gruppo sono più basse, in media, di quelle derivate dall'insieme

completo delle isoforme), così come un aumento dei falsi positivi nei gruppi $I_g = 2$ e $I_g = 3$ (dato che la vera varianza è più alta, in media, di quelle derivate dall'insieme completo).

EBSeq modella direttamente la differenza nelle variabilità come una funzione di I_g , fornendo un approccio potente per l'inferenza a livello di isoforma. Il modello è anche utile per identificare i geni DE.

3.5.1 Modello

EBSeq richiede conte per i geni o stime dell'espressione delle isoforme; il modello generale è stato sviluppato per l'analisi delle isoforme. Il modello assume che la conta attesa per l'isoforma g del gene i nel campione j si distribuisce come una Binomiale Negativa, K_{i_gj}, dove $i = 1, ..., G$, $j = 1, ..., m$ e $g = 1, ..., N_i$; N_i denota il numero di isoforme del gene i. In particolare si assume che entro la condizione r, $K^r_{i_gj}|n_{i_gj}, p^r_{i_g} \sim NB(n_{i_gj}, p^r_{i_g})$ ovvero

$$\mathbb{P}(K^r_{i_gj}|n_{i_gj}, p^r_{i_g}) = \binom{K^r_{i_gj} + n_{i_gj} - 1}{K^r_{i_gj}} (1 - p^r_{i_g})^{K^r_{i_gj}} (p^r_{i_g})^{n_{i_gj}}$$

dove $n_{i_gj} = n_{i_g0} s_j$ è il numero di successi e $p^r_{i_g}$ è la probabilità di successo secondo la definizione classica di distribuzione Binomiale Negativa (numero di *fallimenti* precedenti il *successo n-esimo* in un processo di Bernulli di parametro p). s_j rappresenta la dimensione della libreria del campione j e può essere definita come il numero totale di reads o ottenuta attraverso uno dei metodi di normalizzazione precedentemente citati; n_{i_g0} è un parametro specifico dell'isoforma ed è comune a tutte le condizioni. Sulla base di questa impostazione la media e la varianza sono date da: $\mu^r_{i_gj} = n_{i_gj}(1 - p^r_{i_g})/p^r_{i_g}$ e $(\sigma^r_{i_gj})^2 = n_{i_gj}(1 - p^r_{i_g})/(p^r_{i_g})^2$.

Si assume una distribuzione a priori su $p^r_{i_g}$: $p^r_{i_g}|a, b^{I_g} \sim Beta(a, b^{I_g})$. L'iperparametro a è condiviso da tutte le isoforme, mentre b^{I_g} dipende da I_g, e adatta le differenze sistematiche nelle variabilità tra i gruppi I_g.

Quando sono disponibili reads di due condizioni biologiche diverse, identificare le isoforme differenzialmente espresse corrisponde ad identificare quelle isoforme per le quali $\mu^{r_1}_{i_g} \neq \mu^{r_2}_{i_g}$. Dato che n_{i_g0} è comune ad entrambe le condizioni, questo è analogo ad identificare quelle isoforme per cui $p^{r_1}_{i_g} \neq p^{r_2}_{i_g}$. Sotto l'ipotesi nulla (EE), i dati $K^{r_1,r_2}_{i_g} = K^{r_1}_{i_g}, K^{r_2}_{i_g}$ derivano dalla distribuzione predittiva a priori $f^{I_g}_0(K^{r_1,r_2}_{i_g})$:

$$f_0^{I_g}(K_{i_g}^{r_1,r_2}) = \left[\prod_{j=1}^{m} \binom{K_{i_gj} + n_{i_gj} - 1}{K_{i_gj}}\right] \frac{Beta(a + \sum_{j=1}^{m} n_{i_gj}, b^{I_g} + \sum_{j=1}^{m} K_{i_gj})}{Beta(a, b^{I_g})} \tag{3.16}$$

Sotto l'ipotesi alternativa (DE), $K_{i_g}^{r_1,r_2}$ deriva dalla distribuzione predittiva a priori $f_1^{I_g}(K_{i_g}^{r_1,r_2})$:

$$f_1^{I_g}(K_{i_g}^{r_1,r_2}) = f_0^{I_g}(K_{i_g}^{r_1}) f_0^{I_g}(K_{i_g}^{r_2}) \tag{3.17}$$

Sia Z_{i_g} una variabile latente che vale $Z_{i_g} = 1$ quando l'isoforma i_g è DE e $Z_{i_g} = 0$ quando l'isoforma i_g è EE; $Z_{i_g} \sim Bernulli(\rho)$ dove ρ denota la probabilità a priori di essere DE. La distribuzione marginale delle conte è così modellata da una mistura di distribuzioni

$$(1-\rho) f_0^{I_g}(K_{i_g}^{r_1,r_2}) + \rho f_1^{I_g}(K_{i_g}^{r_1,r_2}) \tag{3.18}$$

La probabilità a posteriori di essere DE per l'isoforma i_g è ottenuta con la regola di Bayes:

$$\frac{\rho f_1^{I_g}(K_{i_g}^{r_1,r_2})}{(1-\rho) f_0^{I_g}(K_{i_g}^{r_1,r_2}) + \rho f_1^{I_g}(K_{i_g}^{r_1,r_2})} \tag{3.19}$$

3.5.2 Stima dei parametri

Le stime delle medie e delle varianze specifiche per le isoforme sono ottenute attraverso il metodo dei momenti, mentre le stime dei quattro iperparametri globali $(a, b^{I_g=1}, b^{I_g=2}, b^{I_g=3}\ )$ si ottengono attraverso algoritmo EM [43].

Si denotino $\mu_{i_g0}^r$ e $(\sigma_{i_g0}^r)^2$ come la media e la varianza per l'isoforma g del gene i sotto dimensione della libreria standard. Allora $\mu_{i_g0}^r = \frac{1}{s_j}\mu_{i_gj}^r$ per qualunque campione j all'interno della condizione r. Si assuma che ci siano m_r campioni per la condizione r. Si possono ottenere stimatori non distorti $\hat{\mu}_{i_g0}^r = \frac{1}{m_r}\sum_{j\ in\ r} \frac{1}{s_j}\hat{\mu}_{i_gj}^r$ dove $\hat{\mu}_{i_gj}^r = K_{i_gj}^r$.

Dato che $(\sigma_{i_g0}^r)^2 = \frac{1}{s_j}(\sigma_{i_gj}^r)^2$ per qualunque campione j appartenete alla condizione r, si può ottenere lo stimatore $(\hat{\sigma}_{i_g0}^r)^2 = \frac{1}{m_r}\sum_{j\ in\ r}\frac{1}{s_j}(\hat{\sigma}_{i_gj}^r)^2$, che è non distorto condizionatamente a $\mu_{i_g0} = \hat{\mu}_{i_g0}$ dove $(\hat{\sigma}_{i_gj}^r)^2 = (K_{i_gj}^r - s_j\hat{\mu}_{i_g0}^r)^2$.

Siano $\hat{\mu}_{i_g0} = \frac{\hat{\mu}_{i_g0}^{r1} + \hat{\mu}_{i_g0}^{r2}}{2}$ e $\hat{\sigma}_{i_g0}^2 = \frac{(\hat{\sigma}_{i_g0}^{r1})^2 + (\hat{\sigma}_{i_g0}^{r2})^2}{2}$ allora lo stimatore di n_{i_g0} si ottiene da $\hat{n}_{i_g0} = \frac{\hat{\mu}_{i_g0}^2}{\hat{\sigma}_{i_g0}^2 - \hat{\mu}_{i_g0}}$. In questo elaborato $\hat{s}_j$ è stato stimato usando la normalizzazione median-of ratio, mentre se non si ha una stima per ρ lo si può porre pari a 0.5.

3.6 ShrinkSeq

Si consideri un modello lineare generalizzato (Bayesiano) e si indichi con $i = 1, ..., G$ l'indicatore delle variabili e con $j = 1, ..., m$ quello dei campioni. Allora

$$K_{ij} =^d F_{\mu_{ij},\gamma_i} \quad \mu_{ij} = \mathrm{g}^{-1}(\eta_{ij}) \quad \eta_{ij} = \beta_{i0} + \sum_{r=1}^{R} \beta_{ir} x_{jr} \tag{3.20}$$

dove μ_{ij} rappresenta la media della distribuzione F, g una funzione legame, x_{jr} è il valore della *r-esima* covariata per il campione j, e $\boldsymbol{\gamma_i} = (\gamma_{i1}, ..., \gamma_{iK})$ sono parametri non inclusi nella regressione su η_{ij}, usati per modellare la sovra-dispersione o la zero-inflation.

Per dati di RNA-Seq, F spesso rappresenta una generalizzazione della distribuzione di Poisson, come ad esempio la Poisson-Gamma (NB) o la sua versione zero-inflated. Nella parte di regressione è inoltre permessa la presenza di effetti casuali gaussiani. In un esperimento a due gruppi, l'inferenza si focalizza di solito su un coefficiente, cioè β_{i1}, ma sono possibili impostazioni di regressione più generali.

I parametri a livello gerarchico più basso sono dotati di a priori:

1. In tutti i casi l'intercetta della regressione, β_{i0}, è dotata di un a priori piatta $\beta_{i0} =^d N(0, 100)$.

2. Per i parametri di sovra-dispersione o per gli effetti casuali (rispettivamente α_i e τ_i^2) si usa spesso un'a priori informativa in modo da effettuare una compressione dei parametri legati alla dispersione, il che porta a stime più stabili.

3. Al parametro di interesse principale è applicata un'a priori informativa in modo da sistemare la correzione di molteplicità.

Sia denotato un parametro corrispondente ad un'a priori informativa con θ_i (e.g. $\theta_i = \beta_{i1}$) e sia denotata con $\pi_{\boldsymbol{\vartheta}}(\theta)$ l'a priori parametrica di θ_i per $i = 1, ..., G$, dove $\boldsymbol{\vartheta}$ è il vettore di iper-parametri ignoti (parametri dell'a priori). Il vettore $\boldsymbol{\vartheta}$ dipende dalla forma parametrica di $\pi_{\boldsymbol{\vartheta}}(\theta)$ e dal tipo di parametro. Per esempio, $N(\mu, \sigma^2)$ dipende da $\boldsymbol{\vartheta} = (\mu, \sigma^2)$. Si denoti inoltre l'insieme di tutti i vettori di iper-parametri ignoti con A, in modo che $\boldsymbol{\vartheta} \in A$. INLA [44] permette di stimare un modello (3.20) per un fissato valore di A. Prima di discutere la scelta delle a priori, spiegheremo come $\boldsymbol{\vartheta}$ è stimato per a priori parametriche conformi a INLA.

3.6.1 Modello

Di default ShrinkSeq usa una Binomiale Negativa zero-inflated per modellare i dati forniti da un esperimento di RNA-Seq. Tener conto dell'alto numero di conte nulle nei campioni può essere utile: l'elevata sovra-dispersione per le variabili con conte basse può essere causata dal fatto che non si tiene conto della "zero inflation" in modelli NB.
In questo caso vale la seguente parametrizzazione: la densità di $K_{ij} =^d NB(\mu_{ij}, \phi_i)$ è $\mathrm{g}(k_{ij}) = \binom{k_{ij}+n_i-1}{n_i-1} p_{ij}^{n_i}(1-p_{ij})^{k_{ij}}$, con $n_i = \frac{1}{\alpha_i} = \exp(-\phi_i)$ dove ϕ_i è il logaritmo del parametro di dispersione. Per modellare la zero-inflation, sia h la densità di $K_{ij} =^d ZI - NB(\mu_{ij}, w_{0i}, \phi_i)$ dove w_{0i} è il parametro di zero-inflation comune. Allora

$$h(k_{ij}) = w_{0i}\delta_0 + (1 - w_{0i})\mathrm{g}(k_{ij}). \tag{3.21}$$

La regressione coinvolge solo la seconda componente di (3.21) legando il logaritmo di μ_{ij} con le covariate $x_{j1}, ..., x_{jR}$.

In esperimenti con numerosità campionaria limitata i parametri di dispersione sono generalmente difficili da stimare, quindi c'è un generale consenso riguardo i benefici derivanti dal comprimere i parametri legati alla dispersione. ShrinkSeq permette priori parametriche (misture) sia su w_{0i} che su $\alpha_i = \exp(\phi_i)$, ad esempio una mistura di una Dirac con massa a zero e di una distribuzione log-normale può essere utile per α_i:

$$\alpha_i = \exp(\phi_i) =^d q_0\delta_0 + (1 - q_0)\ell N(\mu, \sigma^2)$$

Gli iper-parametri di tale a priori verranno poi stimati con la *procedura iterativa congiunta* sotto riportata. Nel modello ZI-NB sia il parametro di dispersione α_i sia il parametro di zero-inflation w_{0i} hanno la capacità di sovra-disperdere la distribuzione di Poisson, sebbene attraverso meccanismi molto diversi. La compressione di tali parametri è valida perché rende più robusta l'inferenza sul parametro di regressione d'interesse, ovvero β_{i1}. In generale si è notato che l'effetto della compressione di w_{0i} è inferiore rispetto a quello di α_i per cui in genere si preferisce quest'ultimo.

3.6.2 Stima delle a priori

Stima congiunta degli iper-parametri

Per stimare gli elementi di A viene proposto un approccio empirico Bayesiano. Inizialmente ci si focalizza su un singolo $\boldsymbol{\vartheta} \in A$ e si assume che $A^- = A \setminus \{\boldsymbol{\vartheta}\}$ sia noto. Si assuma un'a priori comune $\pi_{\boldsymbol{\vartheta}}(\theta)$ per tutti gli θ_i e si denoti l'a posteriori di θ_i condizionatamente ai dati $\boldsymbol{K_i} = (K_{i1}, ..., K_{im})$

e ad A come $\pi_A(\theta|\boldsymbol{K_i})$. Quindi l'a posteriori può dipendere anche dagli iper-parametri in A oltre che da quelli in $\boldsymbol{\vartheta}$. Assumiamo che $\boldsymbol{K_i}$, $i = 1, ..., G$, siano campioni indipendenti con densità $f_A(k)$. Sia $f_A(k)$ che $\pi_A(\theta|\boldsymbol{K_i} = k)$ possono dipendere da i attraverso modelli o covariate diversi ma per chiarezza eliminiamo l'indicatore. Allora

$$\begin{aligned}\pi_{\boldsymbol{\vartheta}}(\theta) &= \int \pi_A(\theta|k) f_A(k) \mathrm{d}\mu(k) \\ &\approx \pi_A^{Emp}(\theta) = \frac{1}{G}\sum_{i=1}^{G} \pi_A(\theta|\boldsymbol{K_i}) = \frac{1}{G}\sum_{i=1}^{G} \pi_{\{\boldsymbol{\vartheta}\}\cup A^-}(\theta|\boldsymbol{K_i})\end{aligned} \tag{3.22}$$

dove $\pi_A(\theta|\boldsymbol{K_i})$ è l'a posteriori di θ_i dato il suo corrispondente $\boldsymbol{K_i}$. Quindi la stima di $\boldsymbol{\vartheta}$ può essere implementata usando software come INLA che calcola le posteriori marginali dati certi modelli per l'a priori e per i dati, sostituisce le a posteriori nel giusto posto di (3.22), e trova il valore di $\boldsymbol{\vartheta}$ per il quale (3.22) vale. Se $|A| > 1$ e A^- è ignoto, allora (3.22) diventa un sistema di equazioni rispetto agli elementi di A.

Per trovare tutti i $\boldsymbol{\vartheta} \in A$, gli autori hanno proposto un algoritmo iterativo. Per stimare $\boldsymbol{\vartheta}$, si applica una procedura "tipo EM": si inizializzano tutti i $\boldsymbol{\vartheta} \in A$, si calcolano le a posteriori dati i valori correnti, si ristimano tutti i $\boldsymbol{\vartheta}$ e si itera.

Per prima cosa esponiamo la ristima di un singolo $\boldsymbol{\vartheta}$. Sia $A^{(\ell)}$ la stima corrente di A. Allora la nuova stima di massima verosimiglianza $\boldsymbol{\vartheta}^{ML,(\ell+1)}$ è

$$\boldsymbol{\vartheta}^{ML,(\ell+1)} = \underset{\boldsymbol{\vartheta}}{\operatorname{argmax}}\, \mathcal{L}(z_{A^{(\ell)}}; \boldsymbol{\vartheta})$$

dove $\mathcal{L}(z_{A^{(\ell)}}; \boldsymbol{\vartheta}) = \sum_{s=1}^{S} \log(\pi_{\boldsymbol{\vartheta}}(z_{s,A^{(\ell)}}))$ è la log-verosimiglianza di $z_{A^{(\ell)}}$, dove $z_{A^{(\ell)}}$ con $s = 1, ..., S$ è un grande insieme di S campioni indipendenti da $\pi_{A^{(\ell)}}^{Emp}(\theta)$, che è la mistura empirica delle a posteriori di tutti i θ. Quindi, ML è usato atipicamente, perché $z_{s,A^{(\ell)}}$ non è un'osservazione. Invece serve per approssimare una mistura empirica con una forma parametrica specifica.

La ristima degli iper-parametri, $\boldsymbol{\vartheta} \in A$, è svolta separatamente per ciascuna a priori. L'a posteriori marginale di un parametro, tuttavia, può dipendere dall'a priori di altri. Perciò, è richiesta un ristima congiunta delle a posteriori, cosa che è svolta da INLA. Sia B il numero delle a priori informative e $\boldsymbol{\vartheta}_b^{(\ell)}$ il *b-esimo* elemento di $A^{(\ell)}$, ovvero le stime alla iterazione ℓ. Allora, la *procedura iterativa congiunta* per stimare tutti i $\boldsymbol{\vartheta}_b \in A$ prevede:

1. inizializzare $\ell = 0$ e $\boldsymbol{\vartheta}_b^{(0)}$, $b = 1, ..., B$;

2. applicare INLA per stimare le a posteriori $\pi_{A^\ell}(\theta|\boldsymbol{K_i})$;

3. usare MLE per ottenere $\boldsymbol{\vartheta}_b^{(\ell+1)}$, $b = 1, ..., B$;

4. reiterare a partire dal passo (2) fino a convergenza.

Successivamente estenderemo questo algoritmo.

Refinimento marginale delle a posteriori sotto un'a priori alternativa

L'approssimazione numerica in INLA permette un'efficiente integrazione dell'a posteriori congiunta per ottenere le a posteriori marginali. Tuttavia, la procedura iterativa sopra presentata richiede l'uso di priori parametriche conformi ad INLA (o ad altre metodologie Bayesiane complete). Spesso, però, esiste un parametro di particolare interesse. In un esperimento con numerosità campionaria piccola, la sua a priori può avere un effetto considerevole sull'a posteriori e quindi sull'inferenza. Così, può risultare desiderabile rifinire la sua posteriori marginale usando un'a priori più adatta e flessibile. Di seguito mostreremo come rifinire una posteriori marginale, ottenuta dalla procedura iterativa congiunta, quando si cambia un'apriori particolare e lasciando le altre inalterate.

Siano $\pi_{\boldsymbol{\vartheta}_b^*}(\theta)$ e $\pi_{A^*}(\theta|\boldsymbol{K_i})$ rispettivamente l'a priori e l'a posteriori di θ_i, dati gli iper-parametri A^*, dove θ_i corrisponde alla *b-esima* componente di A^*, $\boldsymbol{\vartheta}_b^*$. Gli elementi di A^* risultano dalla *procedura iterativa congiunta*, eccetto per $\boldsymbol{\vartheta}_b^*$ che può essere scelto differentemente. Sia posto $A_{-b}^* = A^* \setminus \boldsymbol{\vartheta}_b^*$. Sotto una nuova a priori $\pi^{'}(\theta)$, la formula seguente fornisce la ristima dell'a posteriori:

$$\pi^{'}_{A^*_{-b}}(\theta|\boldsymbol{K_i}) \propto \pi_{A^*}(\theta|\boldsymbol{K_i})\frac{\pi^{'}(\theta)}{\pi_{\boldsymbol{\vartheta}_b^*}(\theta)} \tag{3.23}$$

La costante di proporzionalità è calcolata attraverso normalizzazione usando l'integrazione. Numericamente, (3.23) può essere problematica quando $\pi_{\boldsymbol{\vartheta}_b^*}(\theta)$ è limitato. Pertanto si consiglia di calcolare $\pi_{A^*}(\theta|\boldsymbol{K_i})$ sotto un'a priori più ampia di quella risultante dalla procedura iterativa congiunta. Per esperienza si è visto che un'a priori con deviazione standard due o cinque volte grande lavora bene, con risultati molto simili in questa gamma.

L'Equazione (3.23) è il cuore della *procedura iterativa marginale*:

1. inizializzare $\ell = 0$ e $\pi^{'}(\theta) = \pi^{'\ell}(\theta) = \pi^{'0}(\theta)$;

2. applicare (3.23) per calcolare l'a posteriori $\pi^{'\ell}_{A^*_{-b}}(\theta|\boldsymbol{K_i})$;

3. stimare la nuova a priori $\pi^{'\ell+1}(\theta)$;

4. reiterare dal passo (2) fino a convergenza. Si raccomanda di inizializzare $\pi^{'}(\theta)$ con $\pi_{\boldsymbol{\vartheta}_b^*}(\theta)$ (e saltare il passo (2) una volta, perché le a posteriori sono note).

Il passo (3) richiede la stima di una nuova a priori e a questo punto si hanno due alternative:

- a priori non parametriche;
- a priori parametriche.

La scelta del tipo di priori, non parametrico o parametrico (e la sua forma), è importante.

Le a priori non parametriche forniscono la massima flessibilità e adattabilità, un vantaggio per il parametro di principale interesse, a causa delle conseguenze per l'inferenza. Nella procedura iterativa marginale la stima di $\pi^{'\ell+1}(\theta)$ è ottenuta a partire dalla mistura empirica delle a posteriori correnti, $\pi^{'\ell,Emp}_{A^*_{-b}}(\theta)$, definita analogamente alla (3.22), applicando uno stimatore Kernel density con kernel Gaussiano su un grande campione ottenuto da questa mistura. Per la scelta delle a priori non parametriche vengono offerte due alternative con crescente stabilità, in particolare nelle code: stima sotto le restrizioni di unimodalità e di log-concavità [45].

Le a priori misture di parametriche, che permettono un punto di massa, possono essere utili per modellare effetti non differenziali. La *procedura marginale iterativa* sopra riportata è usata per stimare la mistura degli iperparametri adattandoli ad un campione della mistura empirica delle a posteriori correnti, usando un algoritmo EM. Tuttavia risulta computazionalmente più efficiente il metodo della *massimizzazione diretta* che massimizza direttamente la verosimiglianza marginale: sia $\pi^{'}_{\boldsymbol{\vartheta}'}(\theta)$ una priori parametrica arbitraria con iper-parametri $\boldsymbol{\vartheta}^{'}$ e sia $f_{\boldsymbol{\vartheta}'}(\boldsymbol{K_i}) = f_{\boldsymbol{\vartheta}',A^*_{-b}}(\boldsymbol{K_i})$ la verosimiglianza marginale dati l'a priori per θ_i e gli iper-parametri delle altre a priori. Infine sia $f_{\boldsymbol{\vartheta}'}(\boldsymbol{K}) = \prod_{i=1}^{G} f_{\boldsymbol{\vartheta}'}(\boldsymbol{K_i})$ il prodotto delle verosimiglianze marginali, allora $f_{\boldsymbol{\vartheta}'}(\boldsymbol{K})$ è massimizzato per

$$\tilde{\boldsymbol{\vartheta}} = \underset{\boldsymbol{\vartheta}'}{\operatorname{argmax}} \sum_{i=1}^{G} \log \left[\int \pi_{A^*}(\theta|\boldsymbol{K_i}) \frac{\pi^{'}_{\boldsymbol{\vartheta}'}(\theta)}{\pi_{\boldsymbol{\vartheta}_b^*}(\theta)} \mathrm{d}\theta \right]$$

Nel caso della scelta delle a priori parametriche, naturali estensioni dell'a priori gaussiana sono la Dirac –Gaussian prior [46] e la Gaussian–Dirac–Gaussian mixture prior [47] rispettivamente:

$$\pi(\beta) = p_0\delta_0 + (1-p_0)N(\beta; 0, \tau^2) \tag{3.24}$$

$$\pi(\beta) = p_{-1}N(\beta; \mu_{-1}, \tau_{-1}^2) + p_0\delta_0 + p_1 N(\beta; \mu_1, \tau_1^2) \tag{3.25}$$

dove δ_0 è la Dirac mass on 0 e $N(\beta; \mu, \tau^2)$ denota la densità di una gaussiana con parametri (μ, τ^2), $p_0 = 1 - p_{-1} - p_1$, e $\mu_{-1} < 0$ e $\mu_1 > 0$. In aggiunta, sono fornite le implementazioni della priori Gamma–Dirac-reverse Gamma mixture [47] e della priori Dirac-central Laplace mixture.

In genere si consiglia di usare a priori parametriche quando l'ipotesi nulla da testare è $H_0 : \beta = 0$. Simulazioni hanno mostrato che tale scelta conduce a stime di (B)FDR [48] piuttosto accurate, mentre a priori non parametriche sono da preferirsi quando l'ipotesi nulla è intervallare. La combinazione di procedure congiunte iterative e di rifinimenti marginali fornisce posteriori marginali di un parametro d'interesse sotto un'a priori flessibile rispettando le dipendenze sugli altri parametri.

3.6.3 Test d'ipotesi per la differenziale espressione

Le ipotesi sono

$$\begin{aligned} H_{0i} &: \beta_i \leq \Delta \ (Null) \\ H_{1i} &: \beta_i > \Delta \ (Alternative) \end{aligned} \tag{3.26}$$

con $\beta_i = \beta_{i1}$ e Δ imposto a priori. Si definisce $\pi_{0i} = \mathbb{P}(H_{0i}|\boldsymbol{K_i})$ e $\pi_{1i} = \mathbb{P}(H_{1i}|\boldsymbol{K_i}) = 1 - \pi_{0i}$. Tipicamente, quelle variabili per cui $\pi_{0i} \leq t$ con t piccolo sono d'interesse. Notare che β_i può anche essere un contrasto. In (3.26), non c'è alcuna ragione particolare di porre $\Delta = 0$. Valori positivi possono essere utili per evitare di individuare effetti non rilevanti come statisticamente significativi, sebbene piccoli.

3.7 SAMseq

3.7.1 Statistica di Wilcoxon

Per la variabile i, siano $K_{i1}, ..., K_{im}$ le conte appartenenti sia alla Classe 1 che alla Classe 2. Si supponga che la Classe r contenga m_r campioni, con $r = 1, 2$ e $m_1 + m_2 = m$. Sia $C_r = \{j : \textit{campione } j \textit{ appartiene alla classe } r\}$, $r = 1, 2$. Se le profondità di sequenziamento di tutti gli m esperimenti sono le stesse, allora $K_{ij_1} > K_{ij_2}$ indica che l'espressione della variabile i è più alta nell'esperimento j_1 che nel j_2. Sia $R_{ij}(K)$ il rango di K_{ij} in $K_{i1}, ..., K_{im}$. Allora, il test statistico di Wilcoxon a due campioni (anche chiamato "Mann–Whitney statistic") è

$$T_i = \sum_{j \in C_1} R_{ij}(K) - \frac{m_1(m+1)}{2} \tag{3.27}$$

Assumiamo che non ci siano ties tra $K_{i1}, ..., K_{im}$. Il termine costante è posto a $-\frac{m_1(m+1)}{2}$ invece di $-\frac{m_1(m_1+1)}{2}$ (la definizione classica) in modo che $E[T_i] = 0$ quando la variabile i non è differenzialmente espressa. Un valore assoluto di T_i elevato è una forte evidenza di differenziale espressione della variabile i, e un valore positivo/negativo di T_i indica se la variabile i è sovraespressa/sottoespressa nella Classe 1. La statistica di Wicoxon dipende solo dai ranghi ed è non parametrica.

3.7.2 Strategia di ricampionamento

La statistica (3.27) ha senso solo se le profondità di sequenziamento dei campioni sono le stesse. Altrimenti $K_{i1}, ..., K_{im}$ non sono comparabili. Sfortunatamente, nei data sets reali, le profondità di sequenziamento di campioni differenti sono spesso molto diverse. Un'idea per risolvere questo problema potrebbe essere semplicemente scalare ciascuna conta K_{ij} per la profondità di sequenziamento del campione j. Tuttavia, si è scoperto che questa tecnica funziona male, dato che non produce conte con l'appropriata quantità di variazione. Per questo motivo si è scelto di usare una strategia di ricampionamento.

Supponiamo che le profondità di sequenziamento degli esperimenti siano $s_1, ..., s_m$. Si assume che queste siano note. Si denoti $s_{min} = \min_{j=1,...,m} s_j$ e $j_{min} = \operatorname{argmin}_{j=1,...,m} s_j$. Vale a dire che il $j_{min} - esimo$ esperimento ha la più piccola profondità di sequenziamento s_{min}. Si tiene l'intera lista delle reads generate dall'esperimento j_{min} immutata, e si accorciano le liste generate dagli altri esperimenti, in modo che anch'essi abbiano profondità di sequenziamento s_{min}. Per fare ciò, ciascuna read viene selezionata casualmente con probabilità s_{min}/s_j. Dopo la selezione, il numero di reads mappate per la variabile i nell'esperimento j è

$$K_{ij}^{'} \sim Binomiale(K_{ij}, s_{min}/s_j). \tag{3.28}$$

Questo metodo viene chiamato *"down sampling"*.

Si può vedere la procedura in un'altra maniera. Se $K_{ij} \sim Poisson(s_j q_{ij})$, dove q_{ij} è l'espressione della variabile i nelle'esperimento j, e si genera $K_{ij}^{'}$ dalla (3.28), allora $K_{ij}^{'} \sim Poisson(s_{min} q_{ij})$. In questo caso l'esperimento j dopo il *down sampling* ha la profondità di sequenziamento attesa s_{min}. La

statistica di Wilcoxon per i dati *down sampled* può essere definita come

$$T_i^{'} = \sum_{j \in C_1} R_{ij}(K^{'}) - \frac{m_1(m+1)}{2}, \tag{3.29}$$

dove $R_{ij}(K^{'})$ è il rango di $K_{ij}^{'}$ in $K_{i1}^{'}, ..., K_{im}^{'}$.

Secondo gli autori, il metodo del *down sampling* lavora bene, ma può essere inefficiente quando s_{min} è piccolo, dato che troppe reads vengono scartate. In questi casi, si ricampiona ciascun esperimento ad una profondità di sequenziamento, che è la media geometrica delle profondità di sequenziamento di tutti gli esperimenti. Più dettagliatamente, sia $\bar{s} = (\prod_{j=1}^{m} s_j)^{1/m}$ e si ricampiona usando

$$K_{ij}^{'} \sim Poisson\left(\frac{\bar{s}}{s_j} K_{ij}\right). \tag{3.30}$$

Questo metodo di campionamento è chiamato *"Poisson sampling"*. Vale la pena notare che anche se K_{ij} segue una distribuzione di Poisson, non vale lo stesso per $K_{ij}^{'}$. Infatti quest'ultimo ha come valore atteso $\bar{s}q_{ij}$, ma la varianza è inflazionata per un fattore $\bar{s} = s_j + 1$. Tuttavia, si è visto che questa inflazione non danneggia significativamente la performance del metodo. Generalmente è impossibile generare una $Poisson(\bar{s}q_{ij})$ da una $Poisson(s_j q_{ij})$ per qualunque valore ignoto di q_{ij} se $\bar{s} > s_j$.
Comparando il *down sampling* con il *Poisson sampling* sotto vari schemi di simulazione, gli autori hanno trovato che i due metodi danno risultati molto simili quando $s_{max}/s_{min} < 10$, mentre negli altri casi il *Poisson sampling* è significativamente migliore del *down sampling*.

Nell'Equazione (3.29), si assume che non ci siano ties tra $K_{i1}^{'}, ..., K_{im}^{'}$. Tuttavia, dato che sono tutti numeri interi, alcuni ties si possono verificare. Per risolvere il problema, a ciascuna conta si è aggiunto un piccolo numero casuale, i.e.

$$K_{ij}^{'} \leftarrow K_{ij}^{'} + \epsilon_{ij},$$

dove $\epsilon_{ij} \sim i.i.d.\, Uniforme(0, 0.1)$, $1 \leq i \leq G, 1 \leq j \leq m$.

Sopra si è assunto che le profondità di sequenziamento $s_1, ..., s_m$ siano note. In pratica possono essere stimate accuratamente con diversi metodi come la TMM, la median-of-ratio, la quantile o la normalizzazione implementata nel pacchetto samr di Bioconductor. Quest'ultima è quella usata in questo elaborato.

3.7.3 Ricampionamenti multipli

La strategia di ricampionamento sopra riportata permette di applicare la statistica di Wilcoxon, ma ha due inconvenienti. Per prima cosa, sono

usati solo sottoinsiemi dei dati e quindi molte reads sono scartate durante la procedura di ricampionamento. Secondo, il ricampionamento, così come l'aggiunta di piccoli numeri per risolvere i ties, porta casualità ai risultati, cosa che può essere un problema per le variabili con conte basse. Questi due inconvenienti possono abbassare la potenza del metodo. Per minimizzare queste limitazioni, si ripete il ricampionamento D volte ($D > 1$) e si fa la media. Cioè, se il rango di K'_{ij} in $K'_{i1}, ..., K'_{im}$ nel ricampionamento d è $R_{ij}(K'^{d})$, si usa la statistica

$$T_i^*(two-class) = \frac{1}{D}\sum_{d=1}^{D}\left(\sum_{j\in C_1} R_{ij}(K'^{d}) - \frac{m_1(m+1)}{2}\right). \tag{3.31}$$

Questa strategia di ricampionamento multiplo aumenta la potenza della statistica di Wilcoxon definita in (3.29) riducendo la sua varianza. Nei dati simulati, si è trovato che $D = 20$ è una valore grande abbastanza per fornire un valore stabile di T_i^* e guadagnare sufficiente potenza.

3.8 NOISeqBIO

NOISeqBIO combina l'impostazione non parametrica di NOISeq con un approccio empirico Bayesiano ispirato a [49]. Questo metodo assume che i geni possano essere classificati in due differenti popolazioni: geni con espressione invariante tra le due condizioni e geni la cui espressione cambia tra le condizioni. In NOISeqBIO, viene definita una statistica Z per valutare il cambio nell'espressione e la distribuzione di probabilità di Z può essere descritta come una mistura di due distribuzioni: una per i geni che cambiano tra le condizioni e un'altra per i geni invarianti. Questa distribuzione mistura f può essere scritta come: $f(z) = p_0 f_0(z) + p_1 f_1(z)$, dove p_0 è la probabilità che un gene abbia lo stesso livello di espressione in entrambe le condizioni, i.e. il rapporto di geni non differenzialmente espressi e il totale, e $p_1 = 1 - p_0$ è la probabilità di un gene di essere differenzialmente espresso tra le condizioni, i.e. il rapporto tra DE e il totale. f_0 e f_1 sono rispettivamente, le densità di Z per i non DE e per i DE. Se una delle due distribuzioni può essere stimata, si può calcolare la probabilità che un gene appartenga a uno dei due gruppi. L'algoritmo consiste dei seguenti tre passi:

1. Calcolare la statistica Z per la differenziale espressione;
2. Stimare il punteggio nullo Z_0;
3. Ottenere la probabilità di DE.

3.8.1 Calcolo della statistica Z

Per NOISeqBIO si adoperano le stesse statistiche usate in NOISeq, ovvero il log-rapporto ($M_i = \log_2(\bar{k}_1/\bar{k}_2)$) e le differenze ($D_i = \bar{k}_1 - \bar{k}_2$) dei valori di espressione medi per le due condizioni. La ragione per cui si usano queste due statistiche è per ottenere misure più affidabili del cambio nel livello di espressione tra le due condizioni, dato che i *fold change* per le variabili con conte basse possono essere erronei e lo stesso accade per la differenza nelle espressioni tra due condizioni nel caso di conte alte. Come in NOISeq, gli zeri nei dati di espressione sono rimpiazzati con un valore piccolo più grande di zero per evitare indeterminazioni quando si calcolano le statistiche. In NOISeqBIO, M e D sono corretti per la variabilità biologica: $M_i^* = \frac{M_i}{a_0+\hat{\sigma}_M}$ e $D_i^* = \frac{D_i}{a_0+\hat{\sigma}_D}$, dove $\hat{\sigma}_M$e $\hat{\sigma}_D$ sono gli standard errors di M_i e di D_i, rispettivamente, e sono calcolati come segue:

- $\hat{\sigma}_M^2 = Var(\log_2(\bar{k}_1/\bar{k}_2)) = Var(\log_2(\bar{k}_1) - \log_2(\bar{k}_2)) = Var(\log_2(\bar{k}_1)) + Var(\log_2(\bar{k}_2))$, assumendo che $\bar{k}_1$ e $\bar{k}_2$ siano indipendenti. Viene usato il metodo delta (i.e. un'approssimazione per serie di Taylor) per stimare $Var(\log_2(K)) \approx \left(\frac{1}{E(K)\log(2)}\right)^2 Var(K)$. Per ciascuna condizione r, si stima $E(\bar{k}_r) = \bar{k}_r$ e $Var(\bar{k}_r) = S_r^2/m_r$. Quindi $\hat{\sigma}_M \approx \frac{1}{\bar{k}_1^2\log(2)^2}\frac{S_1^2}{m_1} + \frac{1}{\bar{k}_2^2\log(2)^2}\frac{S_2^2}{m_2}$.

- $\hat{\sigma}_D^2 = Var(\bar{k}_1 - \bar{k}_2) = \frac{S_1^2}{m_1} + \frac{S_2^2}{m_2}$

a_0 è calcolato come un dato percentile di tutti i valori in $\hat{\sigma}_M^2$ o $\hat{\sigma}_D^2$, rispettivamente, come in [49] (gli autori suggeriscono di prendere il *90esimo* percentile). Infine, per definire la statistica Z si combinano M_i^* e D_i^* nel modo seguente: $Z = \frac{M_i^*+D_i^*}{2}$.

3.8.2 Stima di Z_0

Sia $\mathbb{K}_r$ la matrice di espressione genica per ciascuna condizione sperimentale r ($r = 1, 2$) di dimensione $G \times m_r$, dove G è il numero dei geni e m_r è il numero dei replicati biologici nella condizione r. Si assume che le matrici $\mathbb{K}_r$ siano state precedentemente normalizzate e che i geni non espressi tra tutti i replicati per entrambe le condizioni siano stati rimossi secondo un criterio di filtraggio definito dall'utente.
Al fine di calcolare in seguito la densità nulla f_0 è necessario prima stimare i valori di Z per quei geni che non cambiano in base alle due condizioni (Z_0). Per fare ciò si permutano le etichette dei campioni tra $\mathbb{K}_1$ e $\mathbb{K}_2$ b volte, e si

calcola la statistica Z come sopra. Si ottiene una matrice con b colonne e G righe e Z_0 è generato mettendo insieme tutti i suoi valori.
Quando meno di cinque replicati per condizione sono disponibili, questa distribuzione nulla è scadente dato che il numero di possibili permutazioni è basso. In questi casi si prende in prestito informazione da geni simili. I geni sono raggruppati in base ai loro valori di espressione con un algoritmo di clustering (*k-means*). Per ciascun cluster t, si considerano i valori d'espressione di tutti i geni nel cluster, i_t, come osservazioni entro la condizione corrispondente e poi si mescola questa sottomatrice $b \times i_t$ volte. Per ciascuna permutazione, si calcola Z_0. Quando $i_t \geq 1000$, il cluster è diviso ulteriormente in sub-clusters.

3.8.3 Probabilità di differenziale espressione

Il passo successivo è quello di stimare f_0/f dai punteggi Z_0 e Z. In NOISeqBIO si stimano separatamente f_0 e f usando uno stimatore Kernel Density (KDE) con kernel Gaussiano.
Dato un gene con un punteggio z, la probabilità a posteriori di essere differenzialmente espresso $p_1(z)$ può essere derivata dalla regola Bayesiana come: $p_1(z) = \frac{p_1 f_1(z)}{f(z)} = 1 - \frac{p_0 f_0(z)}{f(z)}$. Per p_0, si considera un limite superiore di $p_0 \leq \min_Z\{f(Z)/f_0(Z)\}$ per evitare valori negativi di p_1.

3.9 voom

3.9.1 Log-counts permillion

I dati di RNA-Seq consistono in una matrice di conte di reads K_{ij}, con $i = 1, ..., G$ geni e $j = 1, ..., m$ campioni. Si indichi con n_j il numero totale di reads mappate per il campione j:

$$n_j = \sum_{i=1}^{G} K_{ij}$$

Il numero di reads osservate per un dato gene è proporzionale non solo al livello di espressione del gene, ma anche alla lunghezza del trascritto e alla profondità di sequenziamento della libreria. Dividere ciascuna conta per la corrispondente dimensione della libreria (in milioni) fornisce le *counts per million* (cpm), una semplice misura di abbondanza delle reads che può essere comparata tra librerie con differenti dimensioni. Si definisce il valore *log-*

counts per million (log-cpm) per ciascuna conta come:

$$y_{ij} = \log_2 \left(\frac{K_{ij} + 0.5}{n_j + 1.0} \times 10^6 \right)$$

I log-cpm sono trattati analogamente ai valori di log-intensità degli esperimenti di microarray, con la differenza che non si può assumere che i valori di log-cpm abbiano varianze costanti.
Al numeratore è stato aggiunto 0.5 in modo da evitare di considerare logaritmi di zero, e da ridurre la variabilità dei log-cpm per geni con espressione bassa. Al denominatore è stato aggiunto 1 in modo da assicurare che $(K_{ij} + 0.5)/(n_j + 1.0)$ sia strettamente compreso tra 0 e 1.

3.9.2 Proprietà dei log-cpm

Le distribuzioni di probabilità delle conte sono naturalmente eteroschedastiche, con varianza più grande per le conte grandi.
Si indichi con $\mu = E[K]$ il valore atteso di una conta date le condizioni sperimentali, e si supponga che :

$$Var[K] = \mu + \alpha\mu^2$$

dove α è il parametro di dispersione. Se K è grande, allora il valore in log-cpm dell'osservazione è:

$$y \approx \log_2(K) - \log_2(n) + 6\log_2(10)$$

dove n è la dimensione della libreria. Si noti che l'analisi è condizionata a n, così n è trattata come una costante. Segue che $Var[y] \approx Var[\log_2(K)]$. Se anche μ è grande, allora:

$$\log_2(K) \approx \mu + \frac{K - \mu}{\mu}$$

per il teorema di Taylor [50], allora:

$$Var[y] \approx \frac{Var[K]}{\mu^2} = \frac{1}{\mu} + \alpha.$$

Il primo termine $\left(\frac{1}{\mu}\right)$ deriva dalla variabilità tecnica associata al sequenziamento, e gradualmente decresce con la dimensione attesa delle conte, mentre la variabilità biologica (α) rimane più o meno costante. Si può concludere che i valori in log-cpm generalmente mostrino un trend media-varianza che decresce dolcemente con la dimensione delle conte, e che la trasformazione log-cpm approssimativamente rimuova il trend tra la varianza delle conte di RNA-Seq in funzione della dimensione delle conte per i geni con conte più grandi.

3.9.3 Modellazione della varianza a livello osservazionale

Nelle applicazioni di RNA-Seq, le dimensioni delle conte possono variare considerevolmente da campione a campione per lo stesso gene. Campioni diversi possono essere sequenziati a profondità diverse, così dimensioni di conte diverse possono essere piuttosto differenti anche se i valori di cpm sono gli stessi. Per questa ragione si vuole modellare la relazione media-varianza dei log-cpm a livello di osservazione individuale, invece di applicare una stima della variabilità a livello di gene a tutte le osservazioni di uno stesso gene. Una difficoltà tecnica sta nel fatto che si vuole predire le varianze di osservazioni individuali sebbene, per definizione, non ci siano replicati a livello osservazionale da cui poter stimare le varianze.
La strategia che propongono è quella di stimare in modo non parametrico la relazione media-varianza del logaritmo delle conte delle reads e di usare questa relazione per predire la varianza di ciascun valore in log-cpm. L'inverso della varianza predetta è poi trasformato in un peso associato a ciascun valore in log-cpm. Quando i pesi sono incorporati in un modello lineare, la relazione media-varianza per i valori in log-cpm è effettivamente eliminata.

3.9.4 Stima di w_{ij}

L'algoritmo procede come segue.
Per prima cosa, si adattano modelli lineari gene-wise ai valori in log-cpm normalizzati, prendendo in considerazione il disegno sperimentale, le condizioni di trattamento, i replicati e così via. Si assume che:

$$E[y_{ij}] = \lambda_{ij} = x_j^T \beta_i$$

dove x_j è un vettore di covariate e β_i è un vettore di coefficienti ignoti che rappresentano i $\log_2$-*fold changes* tra le condizioni sperimentali. In termini matriciali:

$$E[y_i] = X\beta_i$$

dove y_i è il vettore dei valori in log-cpm per il gene i e X è la matrice di disegno con le x_j come righe. L'interesse centrale è testare se uno o più dei β_{ij} sono uguali a zero.

Il modello lineare sopra riportato è adattato, tramite minimi quadrati, ai valori in log-cpm y_{ij} per ciascun gene. Questo genera una deviazione standard residuale, s_i, per ciascun gene (Figura 3.3a). Vengono anche calcolati le stime dei coefficienti di regressione $\hat{\beta}_i$, i valori stimati $\hat{\lambda}_{ij} = x_j^T \hat{\beta}_i$ e i log-cpm medi $\bar{y}_i$ per ciascun gene. I log-cpm medi sono convertiti in log-conte medie con:

$$\tilde{K} = \bar{y}_i + \log_2(\tilde{n}) - \log_2(10^6)$$

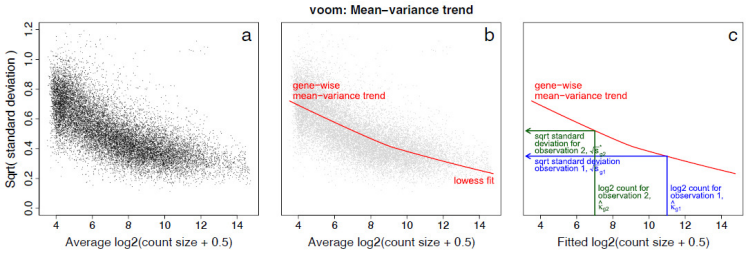


Figura 3.3: **(a)** Scatter plot delle radici quadrate delle deviazioni standard residuali contro le log-conte medie. **(b)** Curva LOWESS (regressione locale pesata) mostra la relazione funzionale tra le medie gene-wise e le varianze. **(c)** Il trend media-varianza permette a ciascuna osservazione di mappare un valore della radice quadrata delle deviazioni standard usando il suo valore predetto per le log-conte.

dove $\tilde{n}$ è la media geometrica delle dimensioni delle librerie più uno.

Si adatta poi un trend robusto alle deviazioni standard residuali in funzione delle log-conte medie per ciascun gene (Figura 3.3b). Per ottenere un trend media-varianza liscio, si adatta una curva LOWESS [51] alla radice quadrata delle deviazioni standard $s_i^{1/2}$ come una funzione della media delle log-conte $\tilde{K}$. Le radici quadrate delle deviazioni standard sono usate perché sono approssimativamente distribuite simmetricamente. La curva LOWESS è statisticamente robusta [52] e fornisce una linea di tendenza attraverso le deviazioni standard. La curva LOWESS definisce una funzione lineare a tratti $lo()$ interpolando la curva sui valori ordinati di $\tilde{K}$.

Successivamente i valori stimati dei log-cpm $\hat{\lambda}_{ij}$ sono convertiti in conte stimate:

$$\hat{\mu}_{ij} = \hat{\lambda}_{ij} + \log_2(n_j + 1) - \log_2(10^6)$$

Il trend è poi interpolato per prevedere la deviazione standard di ogni singola osservazione, $lo(\hat{\mu}_{ij})$, basata sulla sua dimensione della conta stimata, $\hat{\mu}_{ij}$ (Figura 3.3c). Il valore della funzione $lo(\hat{\mu}_{ij})$ è la radice quadrata del valore predetto della deviazione standard dei log-cpm, y_{ij}.

Infine l'inverso del quadrato della deviazione standard predetta per ciascuna osservazione diventa un peso associato a quella osservazione, $w_{ij} = lo(\hat{\mu}_{ij})^{-4}$.

I valori di log-cpm y_{ij} e i pesi associati w_{ij} sono poi inseriti nella modellazione lineare standard di limma e nella pipeline di analisi differenziale empirica Bayesina. La maggior parte delle funzioni di limma sono disegnate per accettare pesi quantitativi, il che permette di attuare analisi simili a quel-

le fatte per i microarray mentre si tiene conto della relazione media-varianza dei valori in log-cpm a livello osservazionale. La pipeline di limma include la modellazione lineare per analizzare complessi esperimenti con fattori di trattamento multipli o pesi quantitativi per tener conto delle variazioni nella precisione tra differenti osservazioni, e metodi statistici empirici Bayesiani per prendere in prestito forza tra i geni. Prendere in prestito informazione tra i geni è una caratteristica fondamentale dei metodi statistici genome-wide, poiché consente variazioni specifiche del gene, ma contemporaneamente permette che l'inferenza sia affidabile anche con numerosità campionarie piccole.

3.10 False Discovery Rate

Il *false discovery rate* (FDR) [53] è un metodo di concettualizzazione del controllo del tasso di errore di I tipo quando sono condotti test multipli. La procedura di controllo dell'FDR, in questo contesto, prevede il controllo della proporzione attesa di falsi positivi tra i geni che sono rilevati come differenzialmente espressi. Tale sistema prevede un controllo dell'errore di I tipo meno restrittivo rispetto alle procedure basate sul *family wise error rate* (FWER), come la correzione di Bonferroni [54], che controllano la probabilità di avere almeno un falso positivo tra tutti i test fatti. La procedura di controllo dell'FDR ha una potenza superiore, al costo di un aumento del tasso di errori del I tipo.
L'FDR non è una quantità che può essere calcolata, ma va stimata dato che, generalmente, tra i geni rilevati come significativamente DE non si conosce quanti di questi siano realmente DE e quanti no.

In questo elaborato si sono considerati diversi metodi per l'inferenza della differenziale espressione che prevedono modi diversi per la stima dell'FDR. In particolare DESeq2, edgeR e voom, che restituiscono un p-value grezzo per ciascun test, adottano la procedura di correzione di Benjamini Hochberg [4], NOISeqBIO usa il *local false discovery rate* (lfdr)[49] definito come uno meno la probabilità a posteriori di essere DE, SAMseq usa un approccio permutazionale [10], mentre baySeq, EBSeq e SkrinkSeq usano il *Bayesian false discovery rate* (BFDR)[47, 48] definito come il valore atteso di lfdr condizionato al fatto che lfdr sia minore di una certa soglia.

In fase di analisi, per determinare la lista dei geni differenzialmente espressi per ogni metodo si sono selezionati quei geni il cui FDR è risultato inferiore a 0.05. Tuttavia, per evitare biases dovuti al metodo di stima dell'FDR adottato dai vari approcci, e per validarne unicamente l'algoritmo di stima della differenziale espressione, si è scelto di considerare anche le liste dei *top500*.

Per ciascun metodo le liste dei *top500* sono state costruite ordinando i valori assoluti della statistica test in ordine decrescente e successivamente selezionando unicamente i primi 500 geni. Tali geni quindi risultano essere i più differenzialmente espressi.

Capitolo 4

Simulazioni

I metodi per l'analisi della differenziale espressione, nella maggior parte dei casi, sono stati applicati a data sets simulati, per i quali si possono controllare le impostazioni e dove si conoscono quali geni sono realmente differenzialmente espressi.

In questo elaborato si sono scelte due strategie di simulazione:

1. Simulazione parametrica;

2. Simulazione non parametrica.

La simulazione parametrica prevede di generare le conte per ogni gene a partire da una distribuzione Binomiale Negativa, poiché negli esperimenti di RNA-Seq reali i dati si presentono sotto forma di conte ed esibiscono un trend media-varianza non lineare. Tuttavia limitare le analisi alla sola simulazione parametrica sembrava restrittivo: molti dei metodi per l'analisi della differenziale espressione si basano infatti su un modello Binomiale Negativo e generare i dati a partire dalla stessa distribuzione rischia di avvantaggiare tali metodi. Si è scelto, quindi, di simulare i dati anche in modo non parametrico per non favorire alcun metodo per l'analisi della differenziale espressione.
Di seguito si riportano nel dettaglio le procedure di simulazione.

4.1 Simulazione parametrica

Le conte per ogni gene sono state simulate da una distribuzione Binomiale Negativa, con media e dispersione stimati da dati di RNA-Seq reali, seguendo il seguente approccio.

4.1.1 Stima dei parametri

Per ciascun trascritto è stata stimata un coppia di parametri (media=$\hat{\mu}_i$, dispersione=$\hat{\alpha}_i$) a partire dai $m = 144$ replicati biologici del data set Kidney [55]. La media è stata stimata tramite massima verosimiglianza: la funzione di log-verosimiglianza per un insieme di osservazioni identicamente distribuite da una NB, $k_1, ..., k_m$, è [56]

$$\begin{aligned}\ell(\mu, \alpha | k_1, ..., k_m) &= \sum_{j=1}^{m} \log \mathbb{P}(K_j = k_j | \mu, \alpha) \\ &= \sum_{j=1}^{m} \log \Gamma(k_j + 1/\alpha) - m \log \Gamma(1/\alpha) - \sum_{j=1}^{m} \log \Gamma(k_j + 1) \\ &+ \sum_{j=1}^{m} k_j \log \left(\frac{\mu\alpha}{1 + \mu\alpha} \right) - \frac{m}{\alpha} \log(1 + \mu\alpha)\end{aligned} \tag{4.1}$$

Considerando l'Equazione 4.1 si può dimostrare che la stima di massima verosimiglianza di μ_i è data dalla classica formula per la stima non distorta di una media,

$$\hat{\mu}_i = \frac{1}{m} \sum_{j=1}^{m} K_{ij}^{Kidney}.$$

Per quanto riguarda la stima del parametro di dispersione, $\hat{\alpha}_i$, si è provato a calcolare una stima di massima verosimiglianza attraverso la massimizzazione numerica della funzione di log-verosimiglianza, ma gli algoritmi di ottimizzazione non riuscivano a convergere. Come alternativa si sono considerate le stima gene-wise della dispersione prodotte dai pacchetti DESeq2 e edgeR. Per ogni trascritto la stima della dispersione è stata calcolata facendo una media aritmetica delle stime prodotte dai due pacchetti. Si è considerata la medesima dispersione per entrambi i gruppi di trattamento.

4.1.2 Algoritmo di simulazione dei dati

Per creare la matrice delle conte simulate è stata usata la funzione *sim.counts* del pacchetto ssizeRNA di Bioconductor versione 1.2.9 [57]. Questa funzione simula le conte per un esperimento di RNA-Seq a due gruppi di trattamento a partire da una distribuzione Binomiale Negativa. In input sono stati forniti i vettori delle medie e delle dispersioni stimate precedentemente e il *fold change*, che in questo caso è stato fissato pari a 2.5. Questa funzione permette di definire quanti geni e quanti campioni per gruppo di trattamento

si vogliono simulare, la percentuale di differenzialmente espressi desiderata, così come la distribuzione tra up e down regolati.

L'algoritmo definisce casualmente la posizione dei geni differenzialmente espressi nella matrice finale delle conte simulate e per ogni trascritto seleziona casualmente una coppia media-dispersione da quelle fornite in input. Ad ogni elemento della matrice delle conte, K_{ij}, vengono quindi associati due parametri (μ_i e α_i) che dipendono da quale coppia media-varianza è associata a quel trascritto i. La media dipende inoltre dallo stato di espressione del gene. Per quei geni che saranno differenzialmente espressi la media dei campioni corrispondenti al secondo gruppo di trattamento viene moltiplicata o divisa per il fold change a seconda se il gene è up o down regolato, generando così una nuova media μ_{ij}. Successivamente per ogni elemento della matrice viene creato un valore casuale a partire da una distribuzione $NB(\mu_{ij}, \alpha_i)$. Il processo viene ripetuto per quei trascritti che non soddisfano la condizione di controllo: alle conte viene applicata la trasformazione Counts per Million (cpm) e i trascritti che devono essere ricreati sono quelli per cui la somma del numero dei campioni che presentano valore di cpm superiore a 2 è inferiore a 3.

4.2 Simulazione non parametrica

Per la simulazione non parametrica si è utilizzato il pacchetto SimSeq di Bioconductor [58] che simula una matrice di conte di un esperimento di RNA-Seq campionando le colonne da un data set di partenza sufficientemente grande, scambiando poi le conte individuali all'interno di ogni gene e infine aggiustando per un fattore di correzione per creare la differenziale espressione.

In questo caso si sono scelti due diversi data sets reali da cui simulare le conte poiché, essendo SimSeq un metodo non parametrico è possibile che le matrici simulate siano influenzate molto dalle caratteristiche dei data sets reali. Si è operata un'analisi esplorativa applicando uno dei metodi considerati più stabili (DESeq2) ai data sets reali al fine di comprenderne, in linea generale, le peculiarità. Il primo data set, Ovary, risulta poco eterogeneo con pochi geni differenzialmente espressi (64 su 6193 geni testati), viceversa il secondo data set, Kidney, mostra molta eterogeneità al suo interno con più della metà dei geni DE (14145 su 20531 geni testati).
La scelta di questi due tipi di data sets risulta fondamentale in fase di validazione e confronto dei metodi di differenziale espressione al fine di capire quali sono quelli che danno risultati affidabili anche in condizioni difficili, ovvero quando c'è poca differenziale espressione o in condizioni anomale, ovvero

quando la maggior parte dei geni è differenzialmente espressa. Per maggiori dettagli sui data sets di partenza si rimanda al Capitolo 5.

4.2.1 Notazione

Sia $\boldsymbol{Y}$ la matrice delle conte delle reads di un esperimento di RNA-Seq, e sia y_{ijr} una singola conta in $\boldsymbol{Y}$, dove $i = 1, ..., G$ è l'indicatore dei geni, $j = 1, ..., m_r$ indica i campioni all'interno di ciascun gruppo di trattamento e $r = 1, 2$ indica i due gruppi di trattamento. Si assuma che sia m_1 che m_2 siano relativamente grandi e si consideri $\boldsymbol{Y}$ come il data set di partenza.

4.2.2 Scelta dei fattori di normalizzazione

Per simulare uno sbilanciamento nella profondità di sequenziamento, si modellano le medie dei livelli di espressione, dato un certo gene i e un certo trattamento r, come se avessero una media comune q_{ir} che è alterata da un fattore di normalizzazione moltiplicativo specifico per ciascun campione s_{jr} tale che,

$$E[y_{ijr}] = q_{ir}s_{jr}$$

Ci sono molti metodi proposti per il calcolo dei fattori di normalizzazione moltiplicativi, in questo elaborato si è scelto la TMM.

4.2.3 Algoritmo di simulazione dei dati

L'algoritmo vuole i seguenti input: un data set di dati di RNA-Seq di partenza $\boldsymbol{Y}$ con due gruppi di trattamento indipendenti; un vettore s di fattori di normalizzazione calcolati con un elemento per ciascuna colonna del data set sorgente; il numero di geni non differenzialmente espressi (EE) G_0 e il numero di DE G_1 nella matrice simulata dove $G_0 + G_1 \leq G$ e il numero di colonne m in ciascuno dei due gruppi di trattamento nella matrice simulata dove $m \leq \min\{m_1, \lfloor m_2 \rfloor\}$ dove $\lfloor \cdot \rfloor$ è la funzione parte intera.
L'output dell'algoritmo SimSeq è una matrice di conte di RNA-Seq, $\boldsymbol{K}$, con G_0 geni EE e G_1 geni DE con m colonne per ciascuno dei due gruppi indipendenti di trattamento.

Sia $\mathcal{G} \equiv \{1, 2, ..., G\}$ l'insieme degli indici di tutti i geni in $\boldsymbol{Y}$.
Il seguente algoritmo descrive la procedura di simulazione:

1. Per ciascun gene $i \in \mathcal{G}$, si calcola un p-value dal Wilcoxon Rank Sum test, ovvero un test per la differenziale espressione.

2. Dato l'insieme dei p-value calcolati, si calcola il *local false discovery rate* (lfdr) per ciascun gene [59, 60] usando il pacchetto *fdrtool*.

3. Un vettore di pesi di probabilità di campionamento $\boldsymbol{w}$ viene calcolato come uno meno il lfdr per ciascun gene i riscalati per sommare a uno.

4. Si selezionano casualmente G_1 geni per essere DE da $\mathcal{G}$ senza reinserimento in base al vettore dei pesi di probabilità di campionamento $\boldsymbol{w}$ e si denota questo insieme come $\mathcal{G}_1$.

5. Si selezionano casualmente G_0 geni per essere EE da $\mathcal{G} \setminus \mathcal{G}_1$ senza reinserimento in base ai pesi di campionamento e si denota questo insieme come $\mathcal{G}_0$. Sia $\mathcal{G}^* \equiv \mathcal{G}_0 \cup \mathcal{G}_1$ l'insieme di tutti i geni EE e DE scelti nel passi 1 e 2.

6. Si seleziona casualmente una colonna di $\boldsymbol{y}$ senza reinserimento dal primo gruppo di trattamento di $\boldsymbol{Y}$. Si seleziona $\boldsymbol{y}$ in base all'insieme $\mathcal{G}^*$ per creare la colonna $\boldsymbol{k_1}$. Si assegna $\boldsymbol{k_1}$ al gruppo di trattamento simulato 1.

7. Si seleziona casualmente una colonna senza reinserimento da entrambi i gruppi di trattamento in $\boldsymbol{Y}$ e si denotano queste due colonne come $\boldsymbol{y_1}$ e $\boldsymbol{y_2}$. Siano s_1 e s_2 i loro corrispondenti fattori di normalizzazioni moltiplicativi da $\boldsymbol{s}$.

8. Si selezionano le due colonne $\boldsymbol{y_1}$ e $\boldsymbol{y_2}$ in base all'insieme di geni $\mathcal{G}^*$.

9. Si crea la colonna $\boldsymbol{k_2}$ nel modo seguente. Per ciascun gene $i \in \mathcal{G}^*$ sia

$$k_{2i} = \begin{cases} y_{1i} & \text{se } i \in \mathcal{G}_0 \\ \lfloor y_{2i} * s_1/s_2 + 0.5 \rfloor & \text{se } i \in \mathcal{G}_1 \end{cases}$$

   dove $\lfloor \cdot \rfloor$ è la funzione parte intera, in modo che $y_{2i} * s_1/s_2$ è arrotondato all'intero più vicino. Sia $\boldsymbol{k_2}$ il vettore le cui entrate sono $\{k_{2i} : i \in \mathcal{G}^*\}$. Si assegna $\boldsymbol{k_2}$ al gruppo di trattamento simulato 2.

10. Si ripetono i passi 6-9 un numero totale di m volte con le colonne campionate senza reinserimento tra ciascuna iterazione.

È possibile operare una modifica all'algoritmo che permette di lavorare con un data set di partenza con un disegno di trattamento appaiato, come in Kidney (si veda Figura 4.1). In questo caso si richiede che $2m \leq \min\{m_1, m_2\}$. Nel passo 1 si usa il Wilcoxon Signed Rank test invece del Wilcoxon Rank Sum test. Si modifica il passo 6 in modo che una coppia di colonne derivanti da una stessa unità sperimentale sia selezionata senza reinserimento, e si fa in modo che la prima colonna della coppia sia $\boldsymbol{y}$. Poi al passo 7, una coppia

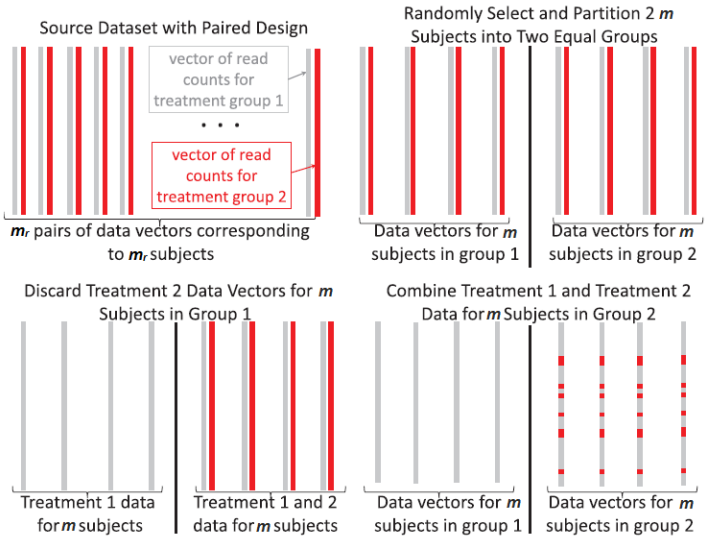


Figura 4.1: Illustrazione dell'algoritmo SimSeq per un data set di RNA-Seq di partenza con disegno di trattamento appaiato. Viene creato un data set simulato con m campioni per ciascuno dei due gruppi di trattamento indipendenti.

di colonne derivanti da un'altra unità sperimentale viene selezionata senza reinserimento e si fa in modo che la prima colonna nella coppia sia $\boldsymbol{y_1}$ e la seconda sia $\boldsymbol{y_2}$. Il procedimento prosegue come al solito nei passi successivi.

Capitolo 5

Dati

In questo capitolo si riportano i dettagli riguardo i data sets utilizzazi per le simulazioni, gli esperimenti di simulazione e un excursus sulla malattia oggetto di studio.

5.1 Kidney

Una parte degli esperimenti di simulazione si basa sul data set di dati di RNA-Seq KIRC ottenuto dal progetto *The Cancer Genome Atlas* [55] e disponibile nel pacchetto SimSeq di Bioconductor [58]. I dati sono stati sequenziati usando la piattaforma di analisi Illumina HiSeq 2000 RNA Sequencing Version 2 e la stima delle conte grezze per ciascun gene è stata calcolata usando il software RSEM [61]. I dati sono scaricabili da The Cancer Genome Atlas: `https://tcga-data.nci.nih.gov/tcga/`. La versione del data set KIRC usata è unc.edu_KIRC.IlluminaHiSeq_RNASeqV2.Level_3.1.5.0.

Il data set KIRC include 20531 geni e 72 coppie di colonne appaiate per ciascun individuo affetto da carcinoma a cellule renali (*Kidney Renal Clear Cell Carcinoma* (KIRC)): una proveniente da una regione tumorale del corpo e l'altro da una regione circostante non tumorale. Come già ricordato nel capitolo precedente, da un'analisi esplorativa condotta sull'intero data set utilizzando DESeq2 è risultato che 14145 geni su 20531 sono rilevati come differenzialmente espressi. Questo numero non ha validità in sé, ma ci permette di trarre alcune conclusioni sulle caratteristiche del data set che sembra essere molto eterogeneo, considerando che DESeq2 è un metodo piuttosto stabile [62]. In questo elaborato ci si riferisce al data set KIRC con il nome Kidney e viene usato unicamente come base per le simulazioni sia parametriche che non.

5.2 Esperimenti di simulazione con Kidney

Nel caso delle simulazioni non parametriche il data set Kidney è stato usato come matrice di partenza da dare in input al pacchetto SimSeq per la generazione di una matrice di conte simulate, mentre nel caso delle simulazioni parametriche Kidney è stato usato per ottenere le stime dei parametri di media e dispersione che sono state poi passate al pacchetto ssizeRNA [57] che simula le conte di un esperimento di RNA-Seq a partire da una distribuzione Binomiale Negativa.
In ciascuno dei nostri esperimenti di simulazione, sono stati simulati 10 data sets ognuno contenente un totale di 5000 geni di cui 4500 EE e 500 DE equamente distribuiti tra sovra-espressi e sotto-espressi. Si sono studiate tre scelte differenti per la dimensione campionaria: per ciascun gruppo di trattamento si sono simulati $m_r = 5$, $m_r = 10$ e $m_r = 20$ campioni. Di seguito indicheremo con $Kidney(sim)$ m_r *campioni* i data sets simulati a partire da Kidney tramite SimSeq con $m_r = 5, 10, 20$ campioni per gruppo di trattamento, mentre $Kidney(NB)$ m_r *campioni* saranno i data sets simulati da una Binomiale Negativa i cui parametri sono stati stimati da Kidney con $m_r = 5, 10, 20$ campioni per gruppo di trattamento.

5.3 Ovary

Ci si riferisce al data set oggetto di studio in questo elaborato con il nome di Ovary e verrà usato come base per le simulazioni, ma soprattutto come caso di studio reale su cui validare quanto scoperto con le simulazioni. Il data set Ovary contiene dati di cancro all'ovaio ottenuti con la tecnologia di sequenziamento Illumina HiSeq2000 (strand specific e paired end). La profondità media di sequenziamento per campione è di 80M di reads. Dopo l'allineamento e la quantificazione si è ottenuta una matrice con 19303 geni (26464 trascritti). I campioni provengono dalla biobanca della ASL Spedali civili di Brescia e sono 28 campioni di pazienti con tumore all'ovaio provenienti da biopsie ovariche prelevate alla diagnosi (quindi naive ad ogni trattamento). I campioni sono stati selezionati per essere da un punto di vista anatomo-patologico molto simili: sono tutti carcinomi di istotipo sieroso di alto grado e di stadio III/IV. I pazienti differiscono invece per la risposta al trattamento chemioterapico: 14 soggetti sono definiti "Sensibili" e 14 "Resistenti". La definizione di resistenza e sensibilità è determinata dal tempo intercorso tra la fine della terapia e la comparsa di una recidiva: se < 6 mesi la paziente è resistente, se > 12 la paziente è sensibile. In fase di analisi del data set reale lo scopo dello studio è valutare la presenza di

marcatori molecolari in grado di stratificare alla diagnosi i pazienti in base alla loro futura risposta alla terapia. Questo permetterebbe di bilanciare la terapia in modo personalizzato per ottenere una prognosi migliore.

5.4 Esperimenti di simulazione con Ovary

Il data set Ovary è stato usato unicamente nel caso delle simulazioni non parametriche come matrice di partenza da dare in input al pacchetto SimSeq per la generazione di una matrice di conte simulate di un esperimento di RNA-Seq. In particolare è stato utilizzato un suo sottoinsieme contenente 6193 trascritti e 28 campioni selezionando unicamente quei trascitti che in fase di allineamento sono risultati perfettamente identici a quelli presenti nei database.
Poiché la numerosità campionaria di Ovary è piuttosto limitata è stato possibile condurre un unico esperimento di simulazione in cui sono stati simulati 10 data sets con numerosità campionaria pari a 5 campioni per gruppo di trattamento, ognuno contenente un totale di 5000 geni di cui 4500 EE e 500 DE equamente distribuiti tra sovra-espressi e sotto-espressi. Di seguito indicheremo con $Ovary(sim)$ i data sets simulati a partire da Ovary tramite SimSeq con $m_r = 5$ campioni per gruppo di trattamento, mentre $Ovary$ sarà il data set reale completo che analizzeremo a fine elaborato.

5.5 Il carcinoma ovarico

Il carcinoma ovarico è un tumore che colpisce le ovaie, due organi delle dimensioni di circa tre centimetri situati uno a destra e uno a sinistra dell'utero al quale sono connessi tramite le tube di Falloppio. Le ovaie sono deputate alla produzione di ormoni sessuali femminili e di ovociti, ovvero le cellule riproduttive femminili: ogni mese, quando la donna è fertile e non è in stato di gravidanza, le ovaie producono un ovocita che si muove verso l'utero per essere fecondato.

Il cancro all'ovaio è dovuto alla proliferazione incontrollata delle cellule cancerose nell'organismo. Nel 90% dei casi il carcinoma ovarico ha origine dalle cellule epiteliali, ovvero le cellule che ricoprono superficialmente le ovaie. Si dice quindi che il tumore è *epiteliale*. Nei restanti casi il tumore può svilupparsi dalle cellule germinali, che sono le cellule che producono gli ovociti, o dalle cellule del tessuto dello stroma gonadico, che è il tessuto di sostegno dell'ovaio. In tali casi il tumore viene detto rispettivamente *germinale* e *stromale* [63]. L'Organizzazione Mondiale della Sanità (OMS)

Stadio I	Il carcinoma è limitato a un ovaio o a entrambi. Esso si sviluppa, all'interno di una ciste ospite per poi romperne la parete ed estendersi, all'esterno nell'ovaio (vegetazioni).
Stadio II	Le cellule cancerose si estendono all'interno della cavità addominale circondata dal peritoneo, pur permanendo all'interno della pelvi (porzione inferiore del peritoneo). Possono così intaccare l'utero, le trombe, il sacco rettale e la vescica.
Stadio III	Le cellule cancerose si estendono verso l'alto, all'interno dell'addome, in direzione dell'intestino, del colon, dello stomaco e del diaframma. Una volta che il peritoneo viene attaccato dalle cellule cancerose, esso produce liquido (ascite) che si deposita dell'addome. Le cellule cancerose possono raggiungere e colonizzare i linfonodi localizzati, a livello dei vasi cardiaci, dell'aorta e della vena cava.
Stadio IV	Il cancro si diffonde al di là dell'addome e raggiunge la pleura, (tessuto che circonda i polmoni) dove produce un liquido detto pleurite e si sposta quindi verso altri organi quali i polmoni o il fegato. Si parla allora di metastasi, ovvero estensione delle cellule cancerose ad altri organi a distanza.

Tabella 5.1: Stadiazione del tumore all'ovaio secondo la classificazione FIGO

classifica i tumori ovarici secondo sei istotipi principali: sieroso, mucinoso, endometrioide, a cellule chiare, a cellule transizionali e squamoso.

Nel mondo occidentale, tra i tumori ginecologici, il carcinoma ovarico è il secondo per frequenza ed il primo come causa di morte. In Italia, secondo le stime del 2012 del Registro Tumori, il tumore dell'ovaio colpisce in media 4.490 donne ogni anno. Considerando le altre forme tumorali esso è al nono posto per frequenza, costituendo il 2,9% di tutte le diagnosi di tumore. In Europa rappresenta il 5% di tutti i tumori femminili [63]. Risulta più frequente nella popolazione caucasica, nei Paesi dell'Europa nord occidentale e negli USA, assai meno frequente nei Paesi asiatici, africani, sudamericani.

In base alla gravità e alla proliferazione, i tumori sono classificati in quattro stadi secondo il sistema FIGO (Federazione Internazionale di Ginecologia e Ostetricia). Capire lo stadio del tumore è essenziale per programmare il trattamento più appropriato. Gli stadi FIGO sono riportati nella Tabella 5.1 (si veda anche la Figura 5.1).

Secondo la FIGO negli stadi iniziali (stadio I) la sopravvivenza a cinque anni è pari all'85%; così non è negli stadi avanzati in cui la sopravvivenza a cinque anni scende al 25% [64]. La ragione di ciò è la mancanza di un metodo per l'identificazione precoce di questo tipo di tumore, infatti più dei

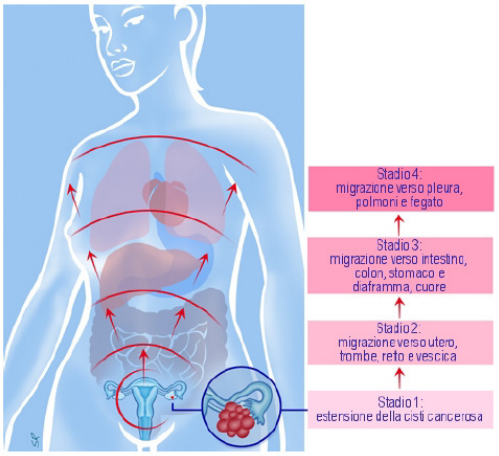


Figura 5.1: Rappresentazione schematica semplificata per visualizzare l'evoluzione e l'estensione del carcinoma ovarico agli altri organi.

due terzi dei tumori vengono diagnosticati in stadi avanzati.

5.6 Tempo computazionale richiesto

In fase di applicazione pratica dei metodi a un data set, una variabile importante di cui bisogna tenere conto è il tempo computazionale richiesto per svolgere le analisi. In questo elaborato si è dovuto limitare il numero di simulazioni per ciascun esperimento a 10, poiché il tempo di elaborazione richiesto da alcuni metodi non ci ha permesso di estendere ulteriormente le analisi. In tutto sono state generate:

- 1 x 10 matrici simulate per Ovary(sim);
- 3 x 10 matrici simulate per Kidney(sim);
- 3 x 10 matrici simulate per Kidney(NB);

per un totale di 70 matrici su ognuna delle quali sono stati applicati i 10 metodi. Le stesse analisi sono state ripetute nel caso le matrici fossero state preventivamente normalizzate con la TPM per un totale 1400 analisi di cui alcune computazionalmente molto intensive.
L'ideale sarebbe stato svolgere almeno un centinaio di simulazioni per ogni condizione sperimentale, ma il metodo che più ci ha vincolato è stato ShrinkSeq che per ogni *run* impiega un po' più di 2 ore. In tutto, solo per

ShrinkSeq, sono state necessarie circa 280 ore per completare le elaborazioni su tutte le matrici (circa 11 giorni), mentre se si fosse scelto di fare 100 simulazioni per ogni condizione sperimentale sarebbero stati necessari circa 4 mesi. Per indagare il controllo dell'errore di I tipo operato dai metodi che restituiscono un p-value nominale, si è potuto allargare il numero di simulazioni a 100 per ogni condizione sperimentale poiché ShrinkSeq non faceva parte dei metodi considerati. Un altro metodo piuttosto lento, anche se non ai livelli di ShrinkSeq, è baySeq, che per ogni *run* impiega circa 20 minuti. In generale voom richiede solo qualche secondo per operare ed è risultato il metodo che necessita di meno tempo di elaborazione. Per quanto riguarda gli altri metodi, il tempo computazionale richiesto da alcuni di essi, come SAMseq, NOISeq e gli EBSeq, dipende dalla scelta dei parametri, come il numero di iterazioni o di ricampionamenti, ma in genere non supera i 5 minuti per *run*.

Capitolo 6

Risultati

In questo capitolo si procederà ad esporre i risultati del confronto tra i diversi metodi in relazione al tipo di dato su cui si è lavorato. Usando i dati simulati, si sono studiati diversi aspetti in differenti condizioni sperimentali:

- L'abilità di controllare il tasso di errore di primo tipo. Questo è stato valutato calcolando gli errori di I tipo osservati rispetto a un determinato livello di significatività.

- Specificità e sensibilità. Si sono usati l'area (totale/parziale) sotto la curva Receiver Operating Characteristic (ROC) e gli andamenti delle curve ROC medie.

- Concordanza delle liste. Si sono costruiti dei dendrogrammi usando la correlazione di Spearman sui ranghi e un algoritmo di clustering gerarchico.

- L'abilità di controllare il False Discovery Rate. In questo caso si sono considerati il numero di falsi positivi nelle liste dei geni differenzialmente espressi rilevati per un determinato cut off e come complemento si è considerato anche il corrispettivo True Positive Rate.

Con i dati simulati si è valutata anche la capacità della normalizzazione TPM di modificare le prestazioni dei metodi d'inferenza per l'analisi della differenziale espressione.

Per i dati di RNA-Seq reali si sono comparate le liste dei geni indicati come DE tra i differenti metodi, sia in termini di numerosità che in termini di sovrapposizione. Si è analizzata inoltre la concordanza del ranking dei geni ottenuta dai differenti metodi.

6.1 Scelta dei parametri

Molti dei metodi che sono comparati in questo studio permettono all'utente di selezionare i valori di alcuni parametri, cosa che può influenzare i risultati in diversi modi. Nella maggior parte dei casi si sono scelti i valori di default forniti dalle implementazioni in modo da mettersi alla pari di un caso reale in cui è difficile determinare come alterare queste impostazioni per ottimizzare la performance dei metodi. Ciò nonostante sono state fatte alcune scelte che spiegheremo di seguito. Per informazioni più dettagliate riguardo il significato dei diversi parametri, si rimanda alle pubblicazioni originali dei diversi metodi.
In DESeq2, per stimare la relazione media-varianza, si è usata una regressione locale, mentre nell'implementazione del test si è scelto il classico test di Wald. Per edgeR si è scelto di considerare sia il caso in cui la dispersione è comune a tutti i geni (indicato come edgeR.cmn), sia il caso in cui si ha la dispersione tag-wise che viene compressa verso la dispersione comune (indicato con edgeR.tgw). Si è usato il test esatto per trovare i geni differenzialmente espressi tra le condizioni. Per baySeq è stata adoperata una numerosità campionaria pari a 5000 per stimare le distribuzioni a priori, mentre le probabilità a priori per gene di essere differenzialmente espresso per ciascun gruppo sono state fissate pari a 0,5. In EBSeq si è aumentato il numero di iterazioni a 10 per far convergere la stima degli iper-parametri; inoltre si è considerata sia la versione che tiene conto delle isoforme presenti nell'esperimento (indicata con EBSeq.iso) sia quella che non ne tiene conto (chiamata EBSeq). Per SAMseq il numero di permutazioni per stimare l'FDR è stato fissato a 500. Infine con ShrinkSeq si è scelto di utilizzare la distribuzione Binomiale Negativa zero-inflated e di applicare lo shrinkage sia al parametro di dispersione sia al coefficiente di regressione d'interesse della procedura d'inferenza. Per rendere i risultati di ShrinkSeq comparabili con quelli degli altri metodi, non si è imposta una soglia di *fold change* diversa da zero in fase di stima dell'FDR. Per brevità indicheremo NOISeqBIO con il nome di NOISeq.

6.2 Controllo dell'errore di I tipo

Per quei metodi che forniscono p-values nominali (DESeq, edgeR.cmn, edgeR.tgw e voom+limma) si è proceduto a valutare la loro capacità di controllare l'errore di I tipo ad un livello pre-specificato, in assenza di alcun gene realmente DE. Quando non ci sono geni realmente differenzialmente espressi, i p-values dovrebbero seguire approssimativamente una distribu-

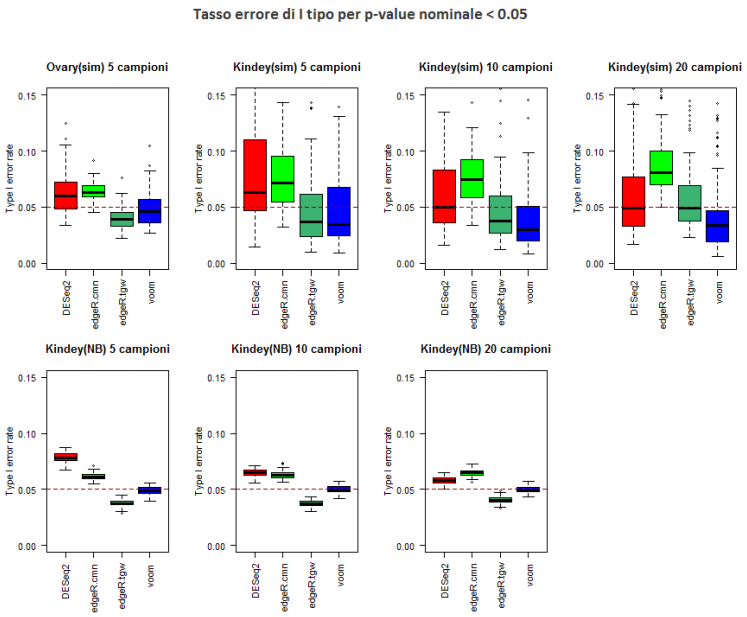


Figura 6.1: Tasso d'errore di I tipo per i quattro metodi che forniscono p-values nominali, nei differenti studi di simulazione.

zione uniforme. Se il tasso di errore di I tipo è controllato correttamente, allora la proporzione attesa di test con p-values sotto un valore nominale alfa dovrebbe essere approssimativamente alfa.

In questo specifico caso per ogni metodo di simulazione e per ogni data set di partenza si sono simulate 100 matrici ognuna contenente 5000 geni tutti EE. Si sono poi applicati i quattro metodi presi in considerazione e si sono rilevate le liste di geni DE alla soglia nominale di 0.05.
A prima vista si può notare che il più grande impatto sui risultati viene dato dal metodo di simulazione: per quanto riguarda le simulazioni non parametriche, gli alfa osservati sono molto più variabili che nel caso delle simulazioni parametriche in cui primo e terzo quartile sono molto vicini alla mediana. Ciò può essere dovuto al fatto che, nel caso parametrico, la maggior parte dei metodi considerati condivide la distribuzione sottostante con quella generatrice dei dati, tuttavia anche i risultati di voom, che non si basa sulla distribuzione Binomiale Negativa, risultano poco variabili. Quindi è verosimile ritenere che nel caso non parametrico l'algoritmo di simulazione permetta ai dati di mantenere una sorta di rumore di fondo, ereditato dalle caratteristiche del data set di partenza, che influenza i metodi e il loro controllo sull'errore di I tipo. In questo particolare caso tutte le colonne delle matrici simulate con

l'algoritmo SimSeq sono campionate dal primo gruppo di trattamento della matrice di partenza (perché tutti i geni sono EE) e le conte non vengono alterate rispetto al caso reale. Il metodo si simulazione non parametrico risente quindi della variabilità dei dati intra classe: maggiore è la variabilità, più è probabile che gli algoritmi rilevino dei falsi positivi. Viceversa, per costruzione, i dati simulati dalla distribuzione NB hanno un livello di variabilità intra classe minimo dovuto unicamente al caso, e quindi anche la probabilità di rilevare dei falsi positivi è piccola. Si tenga conto del fatto che lo scenario più vicino alle applicazioni reali è quello prodotto dalle simulazioni non parametriche. A parte questo si può vedere che comunque i risultati si attestano attorno al cut off scelto.
In Ovary(sim) tutti e quattro i metodi operano abbastanza bene: edgeR.cmn rileva il maggior numero di falsi positivi seguito da DESeq2 che però risulta più variabile, mentre edgeR.tgw è il più conservativo tra i metodi. voom produce risultati molto vicini al valore nominale del tasso d'errore di I tipo.
Nel caso di Kidney(sim) gli alfa osservati risultano molto più variabili in particolare quando la numerosità campionaria è pari a 5 per gruppo di trattamento. All'aumentare del numero di campioni la variabilità dei risultati diminuisce, mentre edgeR.cmn, che anche in questo caso è il metodo che produce più falsi positivi, peggiora leggermente. Questo probabilmente è dovuto al fatto che, all'aumentare della numerosità campionaria, l'assunzione che tutti i geni abbiano la stessa dispersione risulta sempre meno credibile.
In Kidney(sim), voom è il metodo più conservativo e sembra non risentire molto della variazione della numerosità campionaria. Con 5 campioni DESeq2 risulta il più variabile tra i metodi e restituisce una proporzione di falsi positivi leggermente superiore agli altri (tranne edgeR.cmn), mentre si riporta al livello degli altri metodi all'aumentare del numero di campioni. Anche edgeR.tgw sembra campione dipendente diventando sempre meno conservativo.
Passando alle simulazioni parametriche la variabilità dei risultati diminuisce drasticamente seppure i livelli delle mediane siano comparabili ai valori ottenuti con il metodo di simulazione non parametrico. In generale i metodi non sembrano risentire particolarmente dell'aumento della dimensione campionaria, a parte DESeq2 che presenta una riduzione del tasso di errore di I tipo. Nuovamente edgeR.tgw risulta il metodo più conservativo, mentre voom si attesta attorno al valore nominale del tasso d'errore di I tipo. DESeq risulta il metodo che produce più falsi positivi con 5 campioni per gruppo di trattamento, mentre viene sostituito da edgeR.cmn quando i campioni sono 20.

6.3 Sensibilità e specificità

In questa fase si è valutata la capacità dei dieci metodi considerati di discriminare tra veri DE e non. Per fare ciò si è calcolato un punteggio per ciascun gene e per ciascun metodo, che ci permette di ordinare i geni in base alla loro significatività o evidenza di differenziale espressione tra le condizioni. Per DESeq2, edgeR.cmn, edgeR.tgw e voom+limma che restituiscono un p-value nominale, come punteggio si è considerato $1 - p_{nom}$. Per SAMseq si è usato il valore assoluto della statistica di Wilcoxon media, mentre per baySeq, EBSeq, EBSeq.iso, NOISeq e ShrinkSeq si è usata la stima della probabilità a posteriori di differenziale espressione. Tutti questi punteggi non sono influenzati dalla direzione della differenziale espressione (sovra o sotto espressione) tra le due condizioni.
Dato un valore di soglia per una certa statistica e la conoscenza a priori di come sono stati simulati i geni, si possono definire geni veri differenzialmente espressi (veri positivi), geni falsi differenzialmente espressi (falsi positivi), geni realmente non differenzialmente espressi (veri negativi) e geni falsi non differenzialmente espressi (falsi negativi). Con queste quantità si possono calcolare i valori di sensibilità e di specificità. La sensibilità è definita come la proporzione di veri DE che sono dichiarati significativi dal test, mentre 1-specificità da la proporzione di non-DE che sono stati dichiarati DE (si veda Figura 6.2). Rappresentare graficamente le coppie (1-specificità; sen-

	DE	Non DE
Positivi	Veri +	Falsi +
Negativi	Falsi -	Veri -

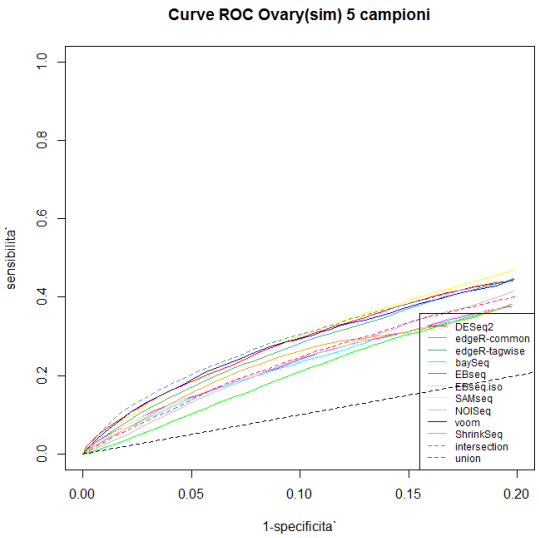
$$S = \frac{V_+}{(V_+ + F_-)}$$

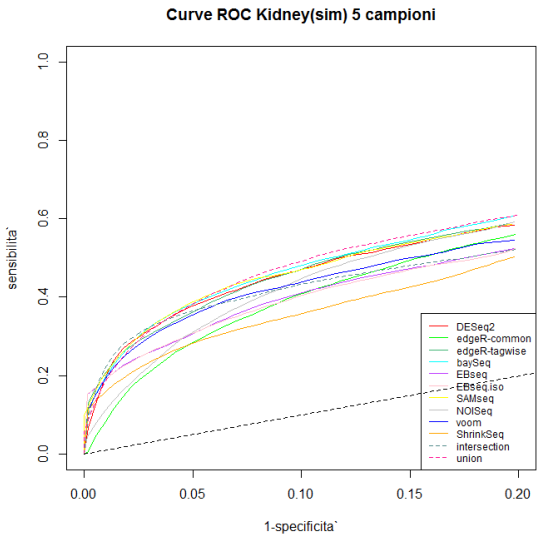
$$1 - S_p = 1 - \frac{V_-}{(V_- + F_+)} = \frac{F_+}{(V_- + F_+)}$$

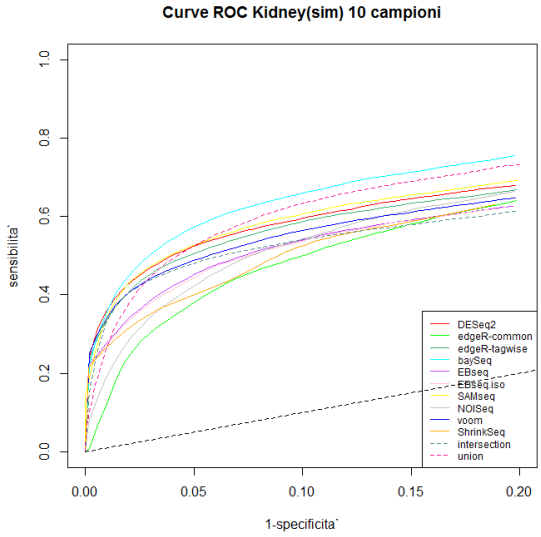
Figura 6.2: Definizione di sensibilità e 1-specificità.

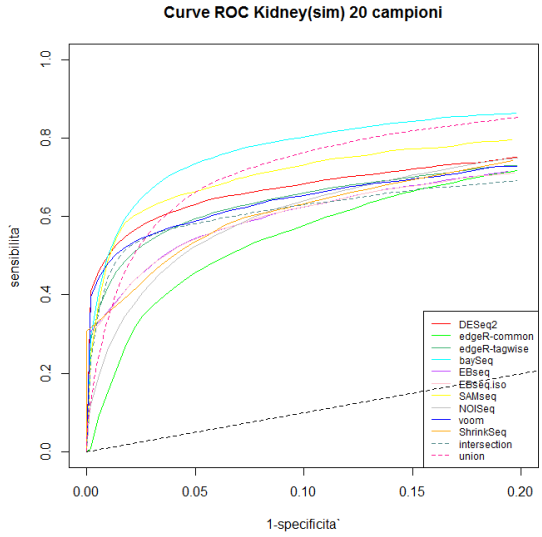
sibilità) al cambiare della soglia produce una curva chiamata Receiver Operating Characteristic curve, o più semplicemente curva ROC. I metodi che ordinano meglio i geni produrranno una curva ROC con maggiore sensibilità rispetto agli altri a parità di specificità. Esaminare le curve ROC ci da modo, quindi, di conoscere l'abilità dei metodi di ordinare i geni in ordine di differenziale espressione. Un metodo ideale ordinerebbe tutti i veri differenzialmente espressi in cima alla lista, mentre la parte inferiore della lista sarebbe composta dai geni senza cambi nel livello di espressione.

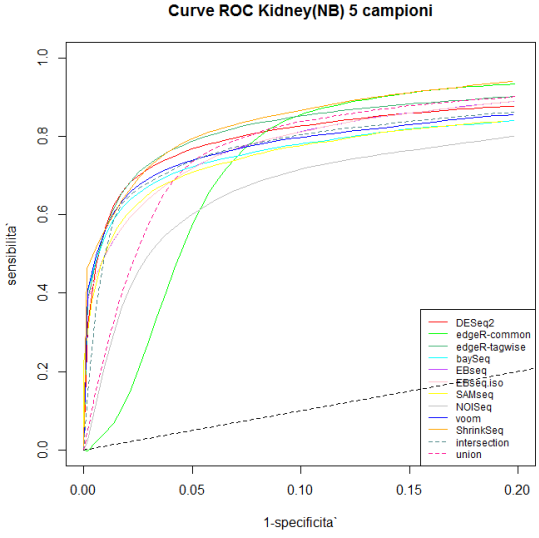
Pertanto per ogni disegno sperimentale e per ogni metodo riportiamo le curve ROC medie, ottenute mediando i valori di sensibilità e specificità

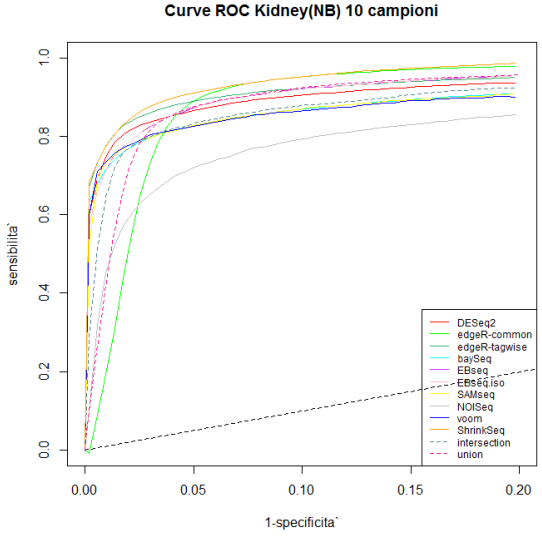

Curve ROC Ovary(sim) 5 campioni
sensibilita`
1-specificita`
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.05
0.10
0.15
0.20
DESeq2
edgeR-common
edgeR-tagwise
baySeq
EBseq
EBSeq.iso
SAMseq
NOISeq
voom
ShrinkSeq
intersection
union


Curve ROC Kidney(sim) 5 campioni
sensibilita`
1-specificita`
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.05
0.10
0.15
0.20
DESeq2
edgeR-common
edgeR-tagwise
baySeq
EBseq
EBSeq.iso
SAMseq
NOISeq
voom
ShrinkSeq
intersection
union


Curve ROC Kidney(sim) 10 campioni
sensibilita`
1-specificita`
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.05
0.10
0.15
0.20
DESeq2
edgeR-common
edgeR-tagwise
baySeq
EBseq
EBSeq.iso
SAMseq
NOISeq
voom
ShrinkSeq
intersection
union


Curve ROC Kidney(sim) 20 campioni
sensibilita`
1-specificita`
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.05
0.10
0.15
0.20
DESeq2
edgeR-common
edgeR-tagwise
baySeq
EBseq
EBSeq.iso
SAMseq
NOISeq
voom
ShrinkSeq
intersection
union


Curve ROC Kidney(NB) 5 campioni
sensibilita`
1-specificita`
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.05
0.10
0.15
0.20
DESeq2
edgeR-common
edgeR-tagwise
baySeq
EBseq
EBSeq.iso
SAMseq
NOISeq
voom
ShrinkSeq
intersection
union


Curve ROC Kidney(NB) 10 campioni
sensibilita`
1-specificita`
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.05
0.10
0.15
0.20
DESeq2
edgeR-common
edgeR-tagwise
baySeq
EBseq
EBSeq.iso
SAMseq
NOISeq
voom
ShrinkSeq
intersection
union

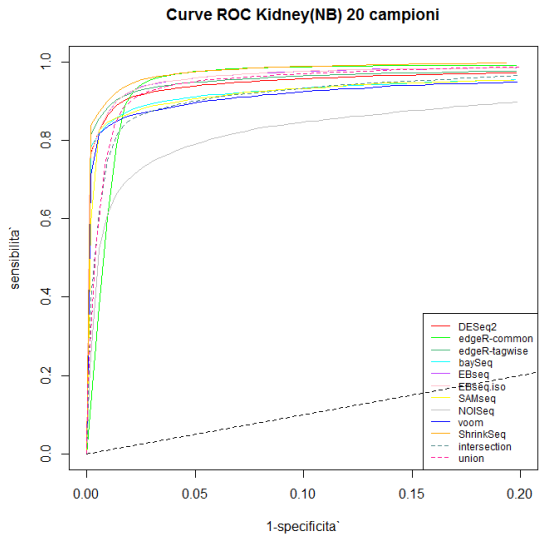


Figura 6.3: Curve ROC per i dieci metodi valutati, nei differenti studi di simulazione.

delle 10 simulazioni svolte. Si riporta unicamente una porzione della curva che ci permette di concentrare l'attenzione su quei valori di specificità che corrispondono a valori di soglia che sono più plausibili, ovvero che sceglieremo se dovessimo fissare una soglia per discriminare DE ed EE. Oltre alle curve ROC per ogni metodo, si sono create anche le curve ROC corrispondenti all'unione e all'intersezione delle liste: dato che i diversi metodi non hanno soglie in comune la strategia che si è adoperato per costruirle è la seguente.

1. A partire dalle liste ordinate della prima iterazione di ciascun metodo si sono presi i primi k geni da ciascuna lista per $k = 2, 4, 6, ..., 5000$.

2. Per ciascun k si è operata l'unione e l'intersezione di tali liste composte dai primi k geni.

3. Si sono calcolati i valore di sensibilità e specificità delle liste "unione" e "intersezione" rispettivamente, per ciascun k.

4. Per ogni iterazione si sono ripetuti i passi 1)-3).

5. Per ogni k, si sono mediati i valori di sensibilità e specificità ottenuti dalle 10 iterazioni rispettivamente.

6. Si sono rappresentati graficamente i valori così ottenuti.

Tali curve servono a darci un'idea dei seguenti aspetti. Se si uniscono due liste di geni rilevati come DE, tendenzialmente il numero di veri positivi dovrebbe aumentare così come il numero di falsi positivi, dunque si verificherebbe un miglioramento nella sensibilità e un peggioramento nella specificità. Viceversa, se si considera l'intersezione di due liste, il numero di veri positivi dovrebbe diminuire così come il numero dei falsi positivi, cosa che si traduce in una diminuzione della specificità, ma in un miglioramento della sensibilità. Il tutto si gioca sul *trade off* sensibilità-specificità dove a seconda di qual è l'entità maggiore del cambiamento di sensibilità e specificità ci si aspetta che la curva unione/intersezione migliori/peggiori rispetto ai singoli casi. In particolare, se due liste hanno un buono numero di geni realmente DE in comune, la loro unione comporta un aumento del numero di falsi positivi a fronte di un numero di veri positivi che rimane praticamente inalterato, dunque ci si aspetta che la sensibilità rimanga costante, mentre la specificità peggiori, quindi la curva ROC dell'unione dovrebbe essere peggiore delle singole curve. Viceversa accade se si considera l'intersezione, in cui pur mantenendo lo stesso numero di veri positivi i falsi positivi diminuiscono, dunque in questo caso la specificità migliora e la curva intersezione dovrebbe essere superiore alle singole curve. Se invece si considera lo scenario in cui due liste sono "quasi-complementari", ovvero entrambe rilevano una porzione di geni DE che l'altra non rileva, allora i casi si invertono. Se si opera l'unione si ha un aumento del numero dei veri positivi a fronte un aumento dei falsi positivi trascurabile, quindi la specificità rimane quasi inalterata mentre la sensibilità subisce un aumento, cosa che porta la curva ROC dell'unione ad essere migliore delle singole curve. Nel caso dell'intersezione invece, il numero dei veri positivi subisce una drastica riduzione mentre quello dei falsi positivi non dovrebbe variare molto, quindi la sensibilità peggiorerebbe e la curva ROC dell'intersezione sarebbe inferiore alle altre. Ovviamente gli scenari descritti sono solo una minima parte di quelli che potrebbero capitare, senza considerare che per valori di k diversi le situazioni si possono invertire. In questo caso la cosa migliore sarebbe selezionare i metodi migliori per disegno di simulazione e fare unione e intersezione solo su questi, ma per problemi computazionali e di tempo non si è esplorato l'insieme di tutte le possibili combinazioni. Si sono considerate invece tutte e 10 le liste, quindi ci si aspetta che tendenzialmente le curve di unione e intersezione si posizionino in mezzo all'insieme delle curve ROC.

Passiamo quindi a dare una panoramica dei vari casi (Figura 6.3): ad un primo sguardo vediamo che le curve si allontanano progressivamente dalla bisettrice avvicinandosi alla situazione ideale man mano che aumenta la differenziale espressione e la numerosità campionaria. Questa è un'indicazione che i test riescono ad ordinare meglio i geni in base alla loro differenziale

espressione quando questa è molto evidente o quando l'informazione a disposizione è superiore. Concentrandoci sulla Tabella 6.2 si può notare che le deviazioni standard delle aree parziali (pAUC) sotto le curve ROC sono maggiori nei casi di simulazioni non parametriche rispetto a quelle delle simulazioni parametriche, e che tra i metodi SAMseq è il più variabile, ma questo è dovuto al fatto che il metodo sia di natura non parametrica.
Osservando le curve ROC nel caso Ovary(sim) si può notare che SAMseq, DESeq2, voom e edgeR.tgw sono i metodi migliori e tutti molto vicini. Tali metodi risultano però inferiori rispetto alla curva dell'intersezione, dunque è verosimile supporre che in questo caso molti metodi trovino un nucleo di geni differenzialmente espressi in comune e che la loro intersezione faccia diminuire il numero dei falsi positivi. Tra i metodi peggiori si trova edgeR.cmn, gli EBSeq e baySeq che in questo caso funziona male probabilmente perché la poca differenziale espressione fa sì che i parametri sottointesi dai due modelli che considera siano molto simili e dunque anche le probabilità di DE e di EE per un gene tenderanno ad essere vicine a 0.5. Come supposto, dato che la curva dell'intersezione risulta molto buona, quella dell'unione è più scarsa della maggior parte dei singoli metodi.
Nel caso di Kidey(sim) si nota che all'aumentare del numero dei campioni le curve tendono ad allontanarsi l'una dall'altra. Questo può indicare che data l'eterogeneità del data set, all'aumentare dell'informazione i diversi metodi colgono aspetti differenti che comportano un ranking diverso delle liste dei geni. Questo sembra confermato dal fatto che, a partire da un valore di 1-specificità intorno a 0.025, la curva ROC dell'unione supera quella dell'intersezione e si porta vicino alle curve corrispondenti ai metodi migliori, cosa che conferma che le liste dei metodi sono piuttosto diversificate e rilevano geni DE che altri metodi non trovano. Quando si considerano pochi geni, invece, la curva dell'intersezione supera quella dell'unione, indicando che i veri geni con una differenziale espressione maggiore dovrebbero essere in gran parte comuni a tutte le liste. baySeq, SAMseq, DESeq2 e edgeR.tgw sembrano confermarsi tra i metodi più efficienti. Al contrario NOISeq, edgeR.cmn, gli EBSeq e ShrinkSeq sono tra i metodi più scarsi, quest'ultimo però migliora all'aumentare della numerosità.
Infine considerando le simulazioni parametriche, si vede che NOISeq in particolare non riesce a raggiungere il livello degli altri metodi. I metodi che non si fondano su una distribuzione Binomiale Negativa risultano peggiori degli altri, mentre questi ultimi risultano tutti molto vicini con ShrinkSeq che opera in maniera leggermente superiore. edgeR.cmn risulta inferiore a quasi tutte le curve fino a un valore di 1-specificità attorno a 0.05 per poi assestarsi attorno agli stessi livelli di ShrinkSeq: questo implica che, a differenza degli altri metodi che rilevano più o meno gli stessi geni per le soglie

più alte, edgeR.cmn a parità di sensibilità rileva molti più falsi positivi. In questo caso sembra quindi che ci siano tre gruppi distinti: i metodi che si fondano sulla NB, quelli che non si basano sulla NB (tranne NOISeq) e NOISeq. Ogni gruppo è caratterizzato dall'avere un ordinamento simile dei geni al loro interno, ma ordinamenti diversi tra i vari gruppi. Questo si riflette sulle curve di unione e intersezione che a differenza dei casi precedenti rimangono in mezzo al fascio di curve: di nuovo per valori di 1-specificità bassi l'intersezione è superiore all'unione e viceversa per i restanti valori, quindi i geni più differenzialmente espressi vengono messi in cima alla lista quasi da tutti, mentre all'aumentare della soglia i metodi si differenziano sulla base del loro impianto teorico.

Per avere un'informazione più pratica e per capire come si comportano i metodi ad uno stesso livello, consideriamo per ogni metodo e per ogni simulazione un punto in particolare sulla curva ROC corrispondente ai primi 500 geni più differenzialmente espressi, dato che 500 era il numero di geni che abbiamo simulato essere DE. Per ciascuna simulazione e per ciascun metodo si sono considerati i primi 500 geni in base al valore assoluto ordinato della statistica che è stata usata per creare le curve ROC; a partire da queste liste si è calcolato il numero di veri positivi e successivamente si è fatta la media dei veri positivi, arrotondandola all'intero più vicino, tra le 10 simulazioni di ogni metodo. Infine per ognuno di questi valori ottenuti si è calcolato la percentuale di falsi positivi, la percentuale di falsi negativi che numericamente in questo particolare caso coincide con 1-specificità [1], e la sensibilità. Abbiamo anche effettuato l'unione e l'intersezione delle liste dei primi 500 geni dei vari metodi per ogni simulazione, rilevando i veri positivi e il numero totale dei geni identificati come DE, abbiamo mediato queste quantità sulle 10 simulazioni e abbiamo calcolato gli stessi punteggi sopra citati (in questo caso 1-specifità e percentuale di falsi negativi non corrispondono). Dalla Tabella 6.1 si può vedere che generalmente la specificità non varia molto da metodo a metodo (in genere la differenza tra due valori è al massimo di 0.02), quindi ci si concentra su una regione della curva ROC ben definita e non troppo grande. In corrispondenza di questi valori, la sensibilità e ovviamente anche il valore assoluto dei veri positivi rispecchiano l'ordinamento dei metodi che si vede in Figura 6.3 per cui metodi che in quell'area corrispondono a curve più alte rileveranno più geni DE correttamente e quindi la sensibilità sarà maggiore. Da notare come la percentuale maggiore di falsi positivi corrisponda, nei disegni non parametrici, a edgeR.cmn, mentre in quelli parametrici,

[1] $1 - Spec = \frac{FP}{VN+FP} = \frac{FP}{4500}$ mentre $\%FN = \frac{FN}{FN+VN} = \frac{FN}{4500}$ ma $FN = FP$ perché il totale dei geni DE simulati è $VP + FN = 500$, mentre il totale dei geni rilevati come DE cioè $VP + FP = 500$ perché abbiamo considerato i primi 500 geni.

a NOISeq. Tendenzialmente all'aumentare della differenziale espressione e della numerosità campionaria si assiste a un aumento del numero di veri positivi trovati con il conseguente aumento della sensibilità e della specificità, mentre le percentuali di falsi positivi e di falsi negativi diminuiscono.

	Ovary(sim)				Kidney(sim) 5 campioni			
	VP	% FP	Sens	1-Spec	VP	% FP	Sens	1-Spec
DESeq2	133	0.73	0.27	0.082	205	0.59	0.41	0.066
edgeR.cmn	94	0.81	0.19	0.090	179	0.64	0.36	0.071
edgeR.tgw	124	0.75	0.25	0.084	213	0.57	0.43	0.064
baySeq	107	0.79	0.21	0.087	221	0.56	0.44	0.062
EBSeq	115	0.77	0.23	0.086	189	0.62	0.38	0.069
EBSeq.iso	115	0.77	0.23	0.086	185	0.63	0.37	0.070
SAMseq	136	0.73	0.27	0.081	188	0.62	0.38	0.069
NOISeq	110	0.78	0.22	0.087	190	0.62	0.38	0.069
voom	132	0.74	0.26	0.082	211	0.58	0.42	0.064
ShrinkSeq	118	0.76	0.24	0.085	196	0.61	0.39	0.068
unione	215	0.82	0.43	0.220	303	0.75	0.61	0.197
intersezione	30	0.55	0.06	0.008	83	0.11	0.17	0.002
	Kidney(sim) 10 campioni				Kidney(sim) 20 campioni			
	VP	% FP	Sens	1-Spec	VP	% FP	Sens	1-Spec
DESeq2	255	0.49	0.51	0.054	303	0.39	0.61	0.044
edgeR.cmn	214	0.57	0.43	0.064	244	0.51	0.49	0.057
edgeR.tgw	260	0.48	0.52	0.053	299	0.40	0.60	0.045
baySeq	289	0.42	0.58	0.047	349	0.30	0.70	0.034
EBSeq	232	0.54	0.46	0.060	272	0.46	0.54	0.051
EBSeq.iso	225	0.55	0.45	0.061	266	0.47	0.53	0.052
SAMseq	245	0.51	0.49	0.057	307	0.39	0.61	0.043
NOISeq	223	0.55	0.45	0.062	266	0.47	0.53	0.052
voom	256	0.49	0.51	0.054	306	0.39	0.61	0.043
ShrinkSeq	239	0.52	0.48	0.058	276	0.45	0.55	0.050
unione	362	0.70	0.72	0.186	414	0.64	0.83	0.164
intersezione	117	0.06	0.23	0.002	154	0.01	0.31	0.000
	Kidney(NB) 5 campioni				Kidney(NB) 10 campioni			
	VP	% FP	Sens	1-Spec	VP	% FP	Sens	1-Spec
DESeq2	361	0.28	0.72	0.031	410	0.18	0.82	0.020
edgeR.cmn	282	0.44	0.56	0.048	371	0.26	0.74	0.029
edgeR.tgw	367	0.27	0.73	0.030	418	0.16	0.84	0.018
baySeq	344	0.31	0.69	0.035	393	0.21	0.79	0.024
EBSeq	336	0.33	0.67	0.036	400	0.20	0.80	0.022
EBSeq.iso	336	0.33	0.67	0.036	400	0.20	0.80	0.022
SAMseq	337	0.33	0.67	0.036	393	0.21	0.79	0.024
NOISeq	294	0.41	0.59	0.046	340	0.32	0.68	0.036
voom	352	0.30	0.70	0.033	394	0.21	0.79	0.024
ShrinkSeq	370	0.26	0.74	0.029	423	0.15	0.85	0.017
unione	425	0.54	0.85	0.113	456	0.45	0.91	0.084
intersezione	163	0.06	0.33	0.002	256	0.01	0.51	0.001
	Kidney(NB) 20 campioni							
	VP	% FP	Sens	1-Spec				
DESeq2	447	0.11	0.89	0.012				
edgeR.cmn	438	0.12	0.88	0.014				
edgeR.tgw	452	0.10	0.90	0.011				
baySeq	433	0.13	0.87	0.015				
EBSeq	446	0.11	0.89	0.012				
EBSeq.iso	446	0.11	0.89	0.012				
SAMseq	431	0.14	0.86	0.015				
NOISeq	371	0.26	0.74	0.029				
voom	427	0.15	0.85	0.016				
ShrinkSeq	460	0.08	0.92	0.009				
unione	478	0.36	0.96	0.060				
intersezione	326	0.00	0.65	0.0002				

Tabella 6.1: Tabella con la cardinalità dei veri positivi, la percentuale di falsi positivi, la sensibilità e 1- specificità delle liste dei primi 500 geni per ogni metodo e per ogni disegno di simulazione.

Per quanto riguarda unione e intersezione, i valori che troviamo non si discostano da quanto previsto: se si fa l'unione la sensibilità tende a migliorare mentre la specificità peggiora, viceversa accade per l'intersezione. Se queste

soluzioni siano utili o meno dipende da quale è l'interesse della ricerca in oggetto di studio.

Per avere invece una misura della capacità di discriminazione complessiva dei metodi ci si concentra ora sull'area sotto le curve ROC (AUC). La Figura

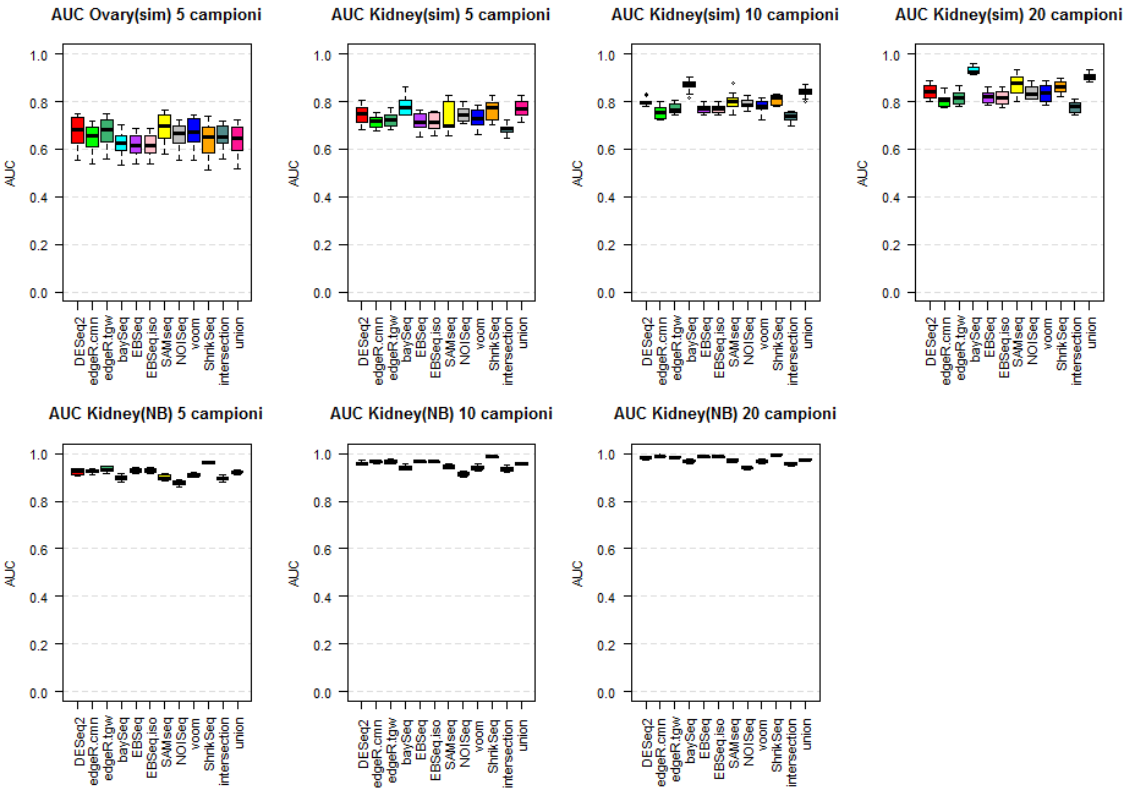


Figura 6.4: Area sotto la curva ROC (AUC) per i dieci metodi valutati e per le loro intersezioni e unioni, nei differenti studi di simulazione.

6.4 riporta i boxplot dei valori delle AUC ottenuti dalle 10 simulazioni per ogni metodo e per ogni disegno di simulazione. Anche in questo caso, come per il controllo dell'errore di I tipo, la differenza più evidente risulta quella tra i diversi metodi di simulazione. Di nuovo, infatti, le AUC nelle simulazioni non parametriche variano molto di più rispetto al caso parametrico in cui, tra una simulazione e l'altra, i risultati non si discostano più di tanto (si veda Tabella 6.2).

	Ovary(sim)				Kidney(sim) 5 campioni			
	AUC medio	sd AUC	pAUC medio	sd pAUC	AUC medio	sd AUC	pAUC medio	sd pAUC
DESeq2	0.666	0.064	0.055	0.016	0.734	0.041	0.087	0.015
edgeR.cmn	0.640	0.055	0.040	0.009	0.703	0.025	0.074	0.013
edgeR.tgw	0.665	0.061	0.052	0.015	0.712	0.027	0.087	0.014
baySeq	0.619	0.052	0.043	0.012	0.769	0.041	0.089	0.015
EBSeq	0.610	0.048	0.044	0.012	0.697	0.036	0.076	0.013
EBSeq.iso	0.610	0.048	0.044	0.012	0.696	0.036	0.076	0.013
SAMseq	0.690	0.061	0.056	0.016	0.734	0.064	0.088	0.023
NOISeq	0.652	0.053	0.045	0.011	0.735	0.033	0.080	0.015
voom	0.665	0.065	0.054	0.016	0.717	0.042	0.082	0.015
ShrinkSeq	0.591	0.071	0.045	0.017	0.695	0.040	0.068	0.015
intersection	0.652	0.049	0.056	0.015	0.684	0.023	0.081	0.011
union	0.642	0.067	0.047	0.012	0.769	0.040	0.091	0.014
	Kidney(sim) 10 campioni				Kidney(sim) 20 campioni			
	AUC medio	sd AUC	pAUC medio	sd pAUC	AUC medio	sd AUC	pAUC medio	sd pAUC
DESeq2	0.790	0.016	0.112	0.006	0.831	0.029	0.131	0.011
edgeR.cmn	0.746	0.025	0.091	0.010	0.795	0.025	0.105	0.011
edgeR.tgw	0.760	0.023	0.110	0.009	0.806	0.027	0.125	0.011
baySeq	0.859	0.029	0.123	0.013	0.923	0.016	0.151	0.009
EBSeq	0.757	0.018	0.099	0.009	0.806	0.027	0.114	0.011
EBSeq.iso	0.756	0.017	0.097	0.009	0.805	0.029	0.116	0.011
SAMseq	0.786	0.039	0.114	0.014	0.852	0.044	0.137	0.016
NOISeq	0.780	0.021	0.099	0.009	0.826	0.026	0.117	0.012
voom	0.770	0.027	0.106	0.009	0.823	0.037	0.125	0.013
ShrinkSeq	0.779	0.022	0.091	0.011	0.850	0.027	0.118	0.014
intersection	0.735	0.019	0.101	0.006	0.778	0.024	0.119	0.010
union	0.840	0.023	0.116	0.010	0.904	0.017	0.140	0.010
	Kidney(NB) 5 campioni				Kidney(NB) 10 campioni			
	AUC medio	sd AUC	pAUC medio	sd pAUC	AUC medio	sd AUC	pAUC medio	sd pAUC
DESeq2	0.913	0.012	0.156	0.004	0.949	0.007	0.174	0.003
edgeR.cmn	0.917	0.008	0.138	0.003	0.958	0.004	0.169	0.003
edgeR.tgw	0.925	0.011	0.161	0.004	0.956	0.007	0.178	0.003
baySeq	0.889	0.012	0.148	0.004	0.929	0.009	0.167	0.003
EBSeq	0.919	0.009	0.152	0.004	0.956	0.004	0.173	0.002
EBSeq.iso	0.919	0.009	0.152	0.004	0.956	0.004	0.173	0.002
SAMseq	0.899	0.011	0.142	0.003	0.929	0.008	0.167	0.003
NOISeq	0.866	0.011	0.128	0.004	0.903	0.007	0.148	0.001
voom	0.899	0.008	0.151	0.004	0.931	0.009	0.167	0.004
ShrinkSeq	0.948	0.004	0.163	0.003	0.975	0.002	0.183	0.002
intersection	0.896	0.011	0.149	0.005	0.935	0.009	0.166	0.003
union	0.923	0.006	0.150	0.003	0.957	0.004	0.172	0.003
	Kidney(NB) 20 campioni							
	AUC medio	sd AUC	pAUC medio	sd pAUC				
DESeq2	0.971	0.005	0.187	0.002				
edgeR.cmn	0.978	0.002	0.186	0.001				
edgeR.tgw	0.974	0.004	0.188	0.002				
baySeq	0.959	0.006	0.182	0.002				
EBSeq	0.976	0.003	0.190	0.001				
EBSeq.iso	0.976	0.003	0.189	0.001				
SAMseq	0.956	0.006	0.180	0.002				
NOISeq	0.928	0.004	0.161	0.002				
voom	0.957	0.007	0.180	0.003				
ShrinkSeq	0.989	0.001	0.188	0.001				
intersection	0.957	0.005	0.178	0.002				
union	0.975	0.002	0.187	0.002				

Tabella 6.2: Tabella con i valori di 1) AUC medio, 2) standard deviation delle AUC 3) pAUC medio, 4) standard deviation delle pAUC, sulle 10 simulazioni per ogni metodo e per ogni disegno di simulazione.

Emerge chiaramente inoltre come la differenziale espressione presente nei data set influenzi la capacità dei test di discriminare DE ed EE: in Ovary(sim), dove la differenziale espressione è scarsa e la numerosità campionaria ristretta, i test fanno più fatica a ordinare correttamente i geni il che si traduce in valori di AUC che variano tra 0.6 e 0.7 e che generalmente indicano che il classificatore lavora in maniere piuttosto scadente. Per Kidney(sim), dove la differenziale espressione è più accentuata, già a partire da 5 campioni, i valori delle AUC si attestano tra 0.7 e 0.8, un valore che indica che i test lavorano in modo discreto. Per campioni di dimensione maggiore tutti i metodi operano meglio, con valori di AUC nel range (0.8, 0.9), dato che suggerisce che in questi casi le capacità di discriminazione dei test sono buone. Infine per quanto riguarda le simulazioni parametriche, dove si è simulato un *fold change* costante pari a 2.5, da subito i valori delle AUC si distribuiscono tra 0.9 e 1 e l'aumento della numerosità campionaria non fa altro che comprimere questi valori verso 1, dunque in questo caso le capacità dei test sono considerate eccellenti e le liste dovrebbero essere molto vicine al modello ideale con i veri DE in cima e i restanti EE in fondo.

Si considerino ora i singoli disegni di simulazione per capire quali metodi funzionano meglio in relazione ai differenti contesti. In Ovary(sim), nonostante la scarsa numerosità, SAMseq sembra il metodo in grado di discriminare meglio DE ed EE, mentre si può notare che tutti i metodi che assumono una distribuzione Binomiale Negativa dei dati operino leggermente peggio. Anche voom che applica una trasformazione delle conte produce risultati vicini a SAMseq: sembra quindi, che in questo caso, i dati siano lontani dall'essere distribuiti secondo una Binomiale Negativa per cui i metodi che non si basano su questa distribuzione risultano avvantaggiati, anche se la numerosità non è elevata. Tra i metodi basati sulla NB come distribuzione dei dati, DESeq2 e edgeR.tgw sono i migliori e probabilmente il loro vantaggio sugli altri è dovuto al fatto di moderare la stima della dispersione, mentre baySeq e gli EBSeq, pur non discostandosi molto dagli altri metodi, funzionano complessivamente peggio.
Per quanto riguarda le simulazioni Kidney(sim) appare chiaro che il metodo che ordina meglio i geni sulla base della loro differenziale espressione sia baySeq e che il divario con gli altri metodi aumenti all'aumentare della numerosità campionaria. Come abbiamo già spiegato, baySeq stima per ogni gene due probabilità a posteriori: la probabilità che il gene sia EE, sottointendendo che tutti i campioni abbiano gli stessi parametri in comune e la probabilità che il gene sia DE, assumendo che i campioni delle due condizioni abbiano parametri diversi, ma condivisi all'interno dei rispettivi gruppi. Il data set ideale per questo metodo è un data set che mostra elevata varianza between-group e bassa within-group tra le conte dei geni DE, qualità che pro-

babilmente i data sets simulati a partire da Kidney hanno. Anche ShrinkSeq e SAMseq sembrano produrre risultati affidabili, mentre edgeR.cmn e gli EBSeq sono i peggiori. Per quanto riguarda le scarse prestazioni di edgeR.cmn, il problema è da ricercare nell'assunzione di una dispersione comune per tutti i geni che si è dimostrato essere lontana dalla realtà, mentre per gli EBSeq il problema può essere che il metodo cerca di cogliere l'incertezza generata in fase di allineamento e quantificazione dell'espressione delle isoforme, ma dato che le conte sono state simulate, tale variabilità spuria non è generata perché non c'è alcuna fase di allineamento.
Nelle simulazioni parametriche dove i dati sono stati generati a partire dalla distribuzione Binomiale Negativa con *fold change* costante per i geni DE, i metodi che si basano sulla stessa distribuzione risultano decisamente avvantaggiati nella discriminazione tra DE ed EE: SAMseq, NOISeq e voom operano leggermente peggio, ma questo non sorprende dato che generalmente, si è dimostrato che i risultati dei metodi parametrici superano quelli dei non parametrici quando l'ipotesi sottostante il modello parametrico è soddisfatta. Anche baySeq in questo caso funziona leggermente peggio degli altri metodi, mentre i risultati di ShrinkSeq sembrano leggermente superiori, anche se all'aumentare della numerosità campionaria il divario con gli altri metodi si riduce ulteriormente e quasi tutte le AUC rasentano la perfezione.

6.4 Concordanza delle liste

Per verificare se ci siano delle similitudini nell'ordine dei geni tra i vari metodi, per ogni simulazione e per ogni coppia di metodi si è calcolato il coefficiente di correlazione per ranghi di Spearman, che è una misura non parametrica di dipendenza statistica tra i ranghi di due variabili (correlazione tra ranghi). La correlazione di Spearman tra due variabili è equivalente alla correlazione di Pearson tra i ranghi di quelle due variabili, ma mentre la correlazione di Pearson valuta una relazione lineare tra le variabili, la correlazione di Spearman accerta una relazione di monotonicità. A livello pratico, il coefficiente ρ viene calcolato come:

$$\rho = 1 - \frac{6 \sum_i D_i^2}{G(G^2 - 1)}$$

dove $D_i = r_i - s_i$ è la differenza dei ranghi, r_i ed s_i sono rispettivamente il rango della prima variabile e della seconda variabile della i-esima osservazione, mentre G è il numero totale di osservazioni. Se non ci sono valori ripetuti, una perfetta correlazione di Spearman è pari a 1 e si verifica quando ciascuna

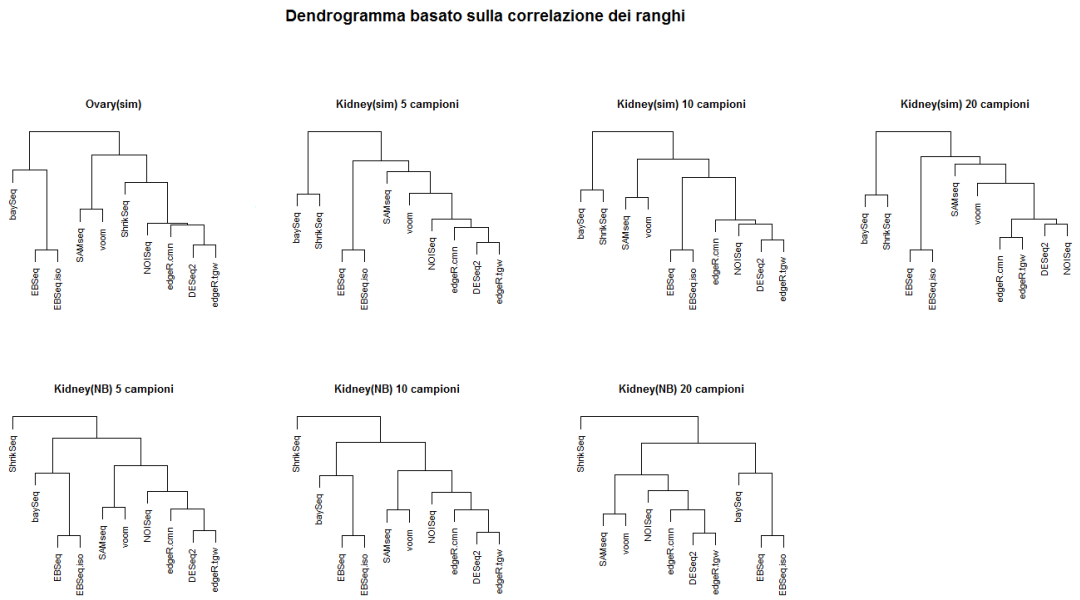


Figura 6.5: Dendrogrammi informativi della similarità nell'ordinare i geni tra i metodi per ogni disegno di simulazione.

variabile è una funzione monotona perfetta dell'altra (o perfettamente opposta per una correlazione di -1). Intuitivamente, la correlazione di Spearman tra due variabili sarà alta quando le osservazioni hanno un rango simile.
Partendo dagli stessi punteggi utilizzati per creare le curve ROC, si sono ottenuti i coefficienti di correlazione per ogni coppia di metodi e per ogni iterazione. Successivamente si sono mediati i valori corrispondenti sulle 10 iterazioni, in modo da ottenere un'unica matrice di similarità per ogni disegno di simulazione. Essendo la correlazione di Spearman una semi-metrica, si può applicare un algoritmo di *cluster analysis* gerarchico con legame completo per stimare un dendrogramma che sia informativo della similiratà di ordinamento dei geni tra i metodi. In Figura 6.5 si può vedere il risultato.
Sono presenti tre cluster principali che non variano molto a seconda del disegno di simulazione: il primo è formato da EBSeq ed EBSeq.iso che praticamente eseguono lo stesso algoritmo, il secondo è costituito da DESeq2, edgeR.cmn, edgeR.tgw e NOISeq che tendono a ordinare i geni in modo simile e l'ultimo è composto da SAMSeq e voom. Quest'ultimo cluster presenta inoltre delle similarità con il secondo. Gli ordinamenti ottenuti da baySeq e dagli EBSeq invece, non risultano particolarmente simili a nessuno degli

altri metodi considerati. ShrinkSeq è il metodo che più varia a seconda dei dati: con Ovary(sim) il suo ordinamento si avvicina a quelli dei cluster di DESeq2 e SAMseq, mentre in Kidney(sim) è più simile a baySeq, infine in Kidney(NB) costituisce un gruppo a sé stante.

6.5 Controllo FDR

In un contesto pratico, uno dei metodi più diffusi per rilevare quali sono i geni differenzialmente espressi è quello di ottenere i p-value aggiustati (o gli FDR o i BFDR) per ogni gene e fissare una soglia per tali valori al di sotto della quale un gene viene definito DE. In questo modo si vorrebbe porre un controllo sulla lista dei geni rilevati come DE, il cui tasso di falsi positivi dovrebbe essere inferiore o al più uguale al valore del cut off scelto. Si è scelto quindi di esaminare se effettivamente porre una soglia di significatività per i p-values aggiustati controlli il tasso di falsi positivi a un livello desiderato. Si è fissata la soglia per gli FDR pari a 0.05, si sono applicati i diversi metodi e si sono calcolate le liste dei geni differenzialmente espressi, come l'insieme di quei geni il cui p-value aggiustato è uguale o inferiore a 0.05. Per ogni metodo e per ogni simulazione si è calcolato l'FDR osservato come la frazione di geni rilevati significativi a questo livello che invece erano false scoperte. Come già ricordato precedentemente per DESeq2, edgeR.cmn, edgeR.tgw e voom si sono usati i p-value corretti tramite la procedura di Benjamini Hochberg, per SAMseq si è usato l'FDR prodotto tramite un approccio permutazionale, mentre per baySeq, EBSeq, EBSeq.iso e ShrinkSeq si sono considerati i BFDR. In questa analisi si è scelto di includere anche NOISeq, per il quale si è considerato l'lfdr, nonostante questo metodo non ritorni alcuna statistica che venga consigliata come stima dell'FDR.
Si consideri la Figura 6.6. Come già osservato in precedenza, la differenza più evidente negli esiti la si può vedere tra i due diversi metodi di simulazione dei dati: i risultati per i data sets simulati in modo non parametrico sono più variabili perché ereditano una sorta di "rumore" dovuto alle caratteristiche del data sets di partenza usato per la simulazione, cosa che non accade nel caso parametrico in cui il modello generatore dei dati è sempre lo stesso.
In Ovary(sim) si può osservare come il controllo dell'FDR sia molto povero per tutti i metodi. SAMseq non riesce a rilevare alcun gene come differenzialmente espresso, mentre dei geni rilevati con voom e NOISeq nessuno risulta realmente DE. I metodi che danno i risultati migliori sono edgeR.tgw e DESeq2 (seppure con esiti molto variabili tra le simulazioni) in cui però circa il 60% dei geni dichiarati DE sono falsi positivi. Sembra quindi che nel caso in cui in un data set vi sia poca differenziale espressione e scarsa numerosità

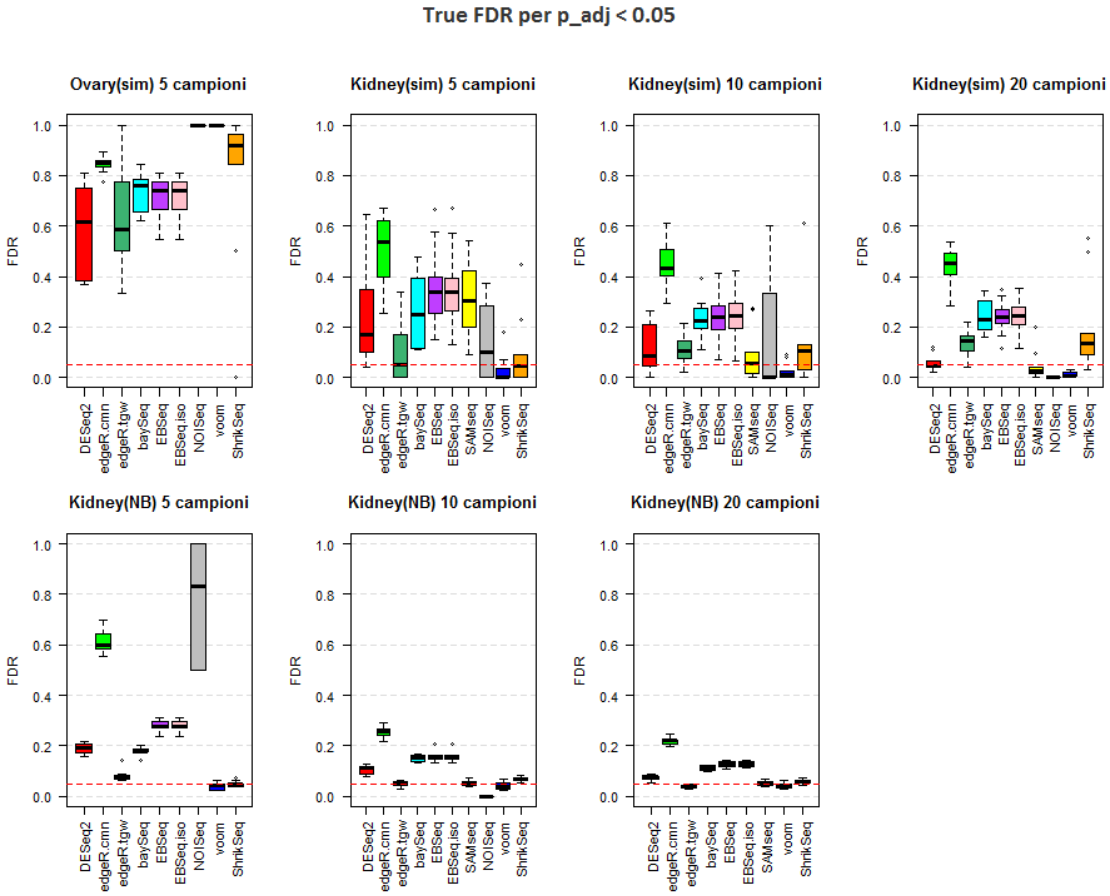


Figura 6.6: False discovery rates (FDR) osservato per una soglia di FDR imposta a 0.05, per i dieci metodi valutati, nei differenti studi di simulazione.

campionaria tutti i metodi non riescano a fornire una lista di geni DE affidabile e che la scelta migliore in questo caso siano i metodi che moderano la stima della dispersione.
Nel caso in cui sia presente una differenziale espressione più evidente, come in Kidney(sim), i risultati mostrano un netto miglioramento, infatti tutti i metodi si attestano almeno sotto il 50% di FDR. In generale, tranne per SAMseq, DESeq2 e gli EBSeq, l'aumento della numerosità campionaria sembra non influenzare molto i livelli dei risultati, ma ha un effetto decisivo sulla loro variabilità che diminuisce sensibilmente. SAMseq è l'unico metodo il cui livello di FDR migliora nettamente con l'aumentare della numerosità campionaria, ma questo non è sorprendente data la sua natura non parametrica

e l'importanza che assegna alle permutazioni dei campioni. Se si considerano i casi in cui la numerosità campionaria è più elevata voom, SAMseq e DESeq2 sembrano i metodi con la percentuale minore di falsi positivi tra i geni rilevati come DE. Anche ShrinkSeq ed edgeR.tgw si allineano attorno al valore della soglia, ma il loro livello si alza leggermente all'aumentare della numerosità campionaria. edgeR.cmn è il metodo il cui controllo dell'FDR risulta peggiore e questo è spiegabile sulla base di come viene stimata la sua dispersione e quindi la sua varianza. Come suggerito dalla teoria, l'lfdr che si è usato per NOISeq come stima dell'FDR non sembra essere paragonabile alle altre stime. Nel caso quindi di un data set con una buona differenziale espressione metodi come voom, edgeR.tgw, DESeq2, ShrinkSeq e SAMseq sembrano essere i più affidabili (tranne SAMseq quando la numerosità campionaria è troppo bassa).
Queste ultime considerazioni non variano molto se ci si sposta in un contesto di simulazione parametrica con una buona differenziale espressione. Come già sottolineato la variabilità si riduce molto rispetto ai casi in cui la simulazione è di tipo non parametrico. In contesto parametrico i metodi sembrano generalmente più sensibili alle variazioni di numerosità (tranne voom e ShrinkSeq): nel caso in cui si hanno 20 campioni per gruppo di trattamento tutti i metodi tranne edgeR.cmn hanno un valore di FDR pari o inferiore a 0.1. Nel caso con scarsa numerosità campionaria SAMseq non rileva alcun gene differenzialmente espresso, mentre già con 10 campioni si riporta su livelli ottimali. ShrinkSeq e voom si posizionano in linea con il livello 0.05 e non subiscono variazioni rilevanti con l'aumentare della numerosità. Anche in questo caso edgeR.cmn è il metodo che da il maggior numero di falsi positivi nelle liste dei geni differenzialmente espressi, anche se il suo livello migliora notevolmente rispetto al caso non parametrico e all'aumentare della numerosità. baySeq, EBSeq ed EBSeq.iso risultano, come nei casi precedenti, ad un livello intermedio, mentre edgeR.tgw e DESeq2 si attestano attorno al livello di cut off pre-scelto migliorando con l'aumentare della numerosità. Dunque in queste condizioni la maggior parte dei metodi sembra riuscire a controllare l'FDR a un livello desiderato.

In un contesto pratico tuttavia, non si è interessati unicamente a limitare il tasso di falsi positivi, ma si vuole anche trovare il maggior numero di geni realmente DE. Nell'ambito di dati genomici questo rappresenta in genere un *trade off*: più aumenta la sensibilità, più le liste si allungano e maggiore è la percentuale di falsi positivi presenti. Per verificare quindi se ci sia un metodo che coniughi un buon controllo dell'FDR con un'elevata percentuale di geni DE trovati, si è calcolato il True Positive Rate (TPR), ovvero la frazione dei geni realmente DE che sono stati rilevati come significativi (in poche parole la sensibilità). Le liste per ogni metodo e per ogni simulazione sono le stesse

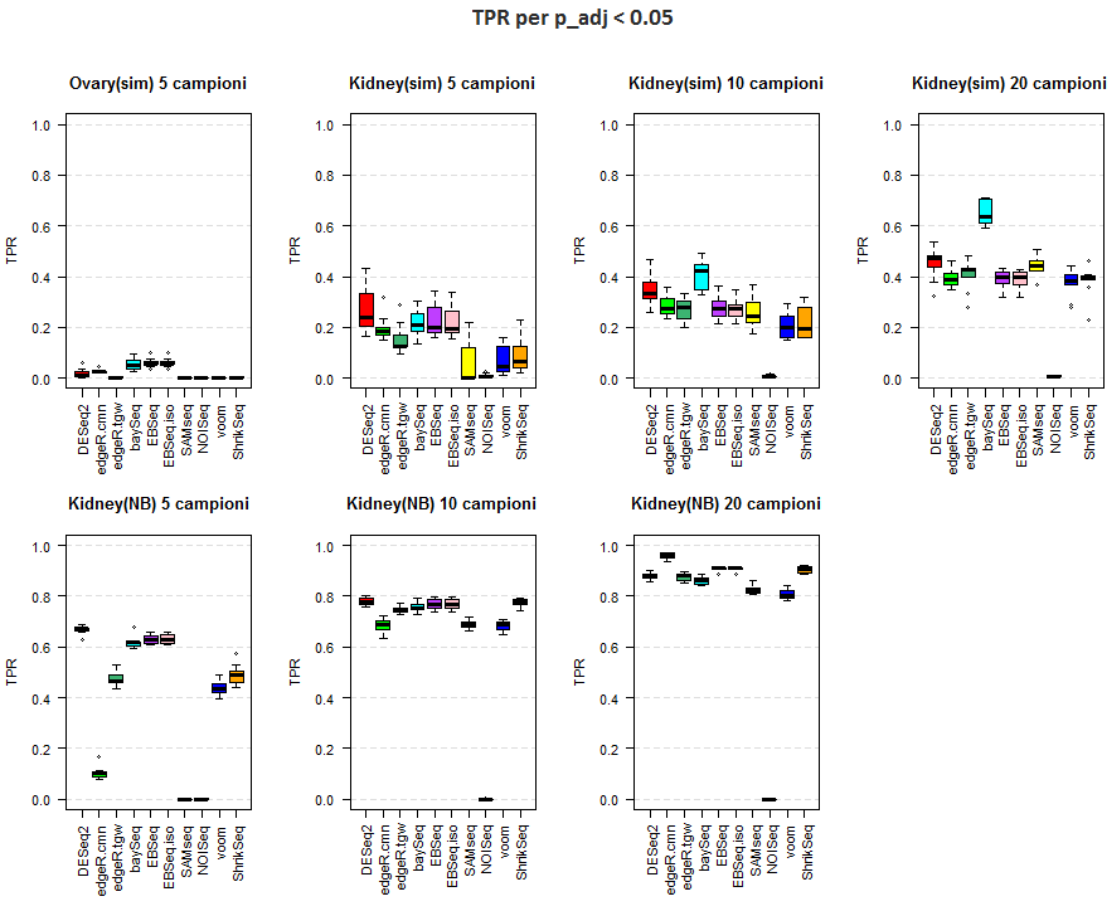


Figura 6.7: True positive rates (TPR) osservato per una soglia di FDR imposta a 0.05, per i dieci metodi valutati, nei differenti studi di simulazione.

che sono state usate per il controllo dell'FDR e che quindi comprendono quei geni la cui stima dell'FDR è inferiore a 0.05.
Si consideri la Figura 6.7. Rispetto al controllo dell'FDR in questo caso la variabilità dei risultati è più simile tra i diversi metodi di simulazione: in Ovary(sim) quasi tutti i metodi rilevano meno del 10% dei geni realmente DE. I metodi con maggiore sensibilità sono baySeq, EBSeq ed EBSeq.iso il cui controllo dell'FDR però risultava piuttosto scarso, mentre DESEq2 e edgeR.tgw che fornivano la percentuale più bassa di falsi positivi rilevano anche pochissimi geni DE correttamente.
In Kidney(sim) baySeq è di gran lunga il metodo con sensibilità maggiore, ma si deve tener conto che circa il 20% dei geni nella sua lista dei DE è un

falso positivo. Da segnalare invece DESeq2 che nonostante abbia sensibilità leggermente peggiore rispetto a baySeq riesce, all'aumentare della numerosità campionaria, a riportare l'FDR osservato a livello della soglia pre-scelta. Metodi come voom, ShrinkSeq, SAMseq ed edgeR.tgw che avevano un ottimo controllo dell'FDR, restituiscono le percentuali più basse di TPR, ma edgeR.tgw e SAMseq migliorano notevolemnte all'aumentare della numerosità campionaria portandosi tra i migliori per 20 campioni. Come DESeq2, anche edgeR.cmn va contro corrente rispetto alla teoria: pur essendo il metodo che aveva la percentuale maggiore di falsi positivi, risulta scarso anche nel trovare i veri geni DE.
Nel caso delle simulazioni parametriche le percentuali dei veri DE correttamente rilevati si alzano molto in particolare all'aumentare della numerosità campionaria dove arrivano fino quasi al 90%. Come nei casi precedenti NOISeq non rileva praticamente alcun gene DE indipendentemente dal disegno di simulazione e dalla numerosità, dunque per questo metodo è necessario considerare strategie alternative per rilevare i geni DE che non si basino sull'lfdr. Tutti i metodi vengono influenzati pesantemente dalla numerosità campionaria: SAMseq con 5 campioni per gruppo di trattamento non rileva alcun gene DE e dunque anche la percentuale di TPR è nulla, mentre già con 10 campioni si riporta a livello degli altri metodi seppure risulta tra gli ultimi come quantità di veri DE rilevati. SAMseq e voom rispettano quindi il *trade off* per cui si ha buon controllo dell'FDR, ma sensibilità più scarsa. ShrinkSeq invece mantiene una percentuale di falsi positivi nella lista dei DE vicina al 5%, ma all'aumentare della numerosità trova quasi il 90% dei geni DE. DESeq2 sembra avere un buon compromesso tra controllo dell'FDR e sensibilità, in particolare per campioni piccoli. baySeq, EBSeq ed EBSeq.iso sono tra i metodi con maggior sensibilità e maggior numero di falsi positivi. In conclusione DESeq2 per numerosità piccole sembra il metodo migliore per coniugare sensibilità elevata e bassa percentuale di falsi positivi, mentre con campioni più numerosi ci si può servire di edgeR.tgw e ShrinkSeq. Gli altri metodi tendono invece a seguire il *trade off* per cui, più o meno evidentemente a seconda dei casi, se la sensibilità è alta lo sarà anche la percentuale di falsi positivi e viceversa.

6.6 Confronto TPM e non

Negli ultimi anni si è discusso molto sulla normalizzazione *Transcripts per million* (TPM) e sulle sue capacità di correggere sia per la profondità di sequenziamento sia per la lunghezza dei trascritti, ma non si è ancora giunti ad una conclusione unanime se effettivamente apporti dei sostanziali benefici

ai metodi inferenziali. Poiché il risultato di tale normalizzazione è una conta trasformata, la domanda che gli esperti si pongono è se la normalizzazione incida sul modello generativo alla base dei metodi inferenziali, per cui ci si aspetta che la normalizzazione non modifichi sostanzialmente i risultati nel caso di metodi non parametrici, mentre con quelli parametrici, non è detto che gli esiti rimangano inalterati. Al fine di verificare se questo tipo di normalizzazione modifichi le prestazioni fornite dai metodi d'inferenza per l'analisi della differenziale espressione considerati in questo elaborato, di seguito si riportano alcune delle analisi fin qui svolte nel caso in cui i data sets siano normalizzati con la TPM con lo scopo di evidenziare le differenze rispetto al caso in cui i dati di partenza erano conte grezze.

6.6.1 Controllo dell'errore di I tipo

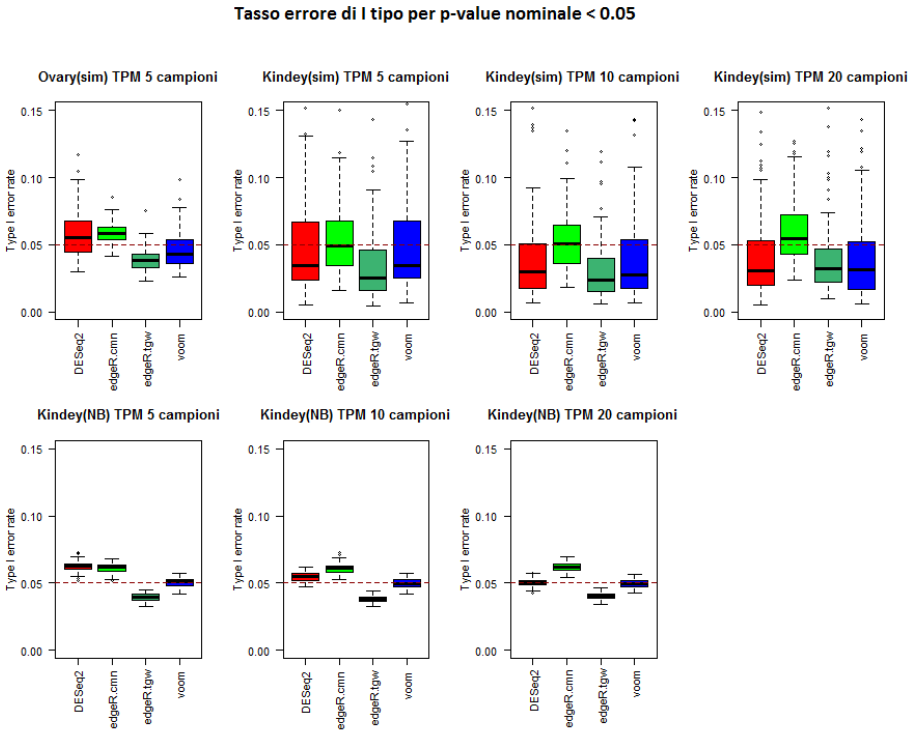


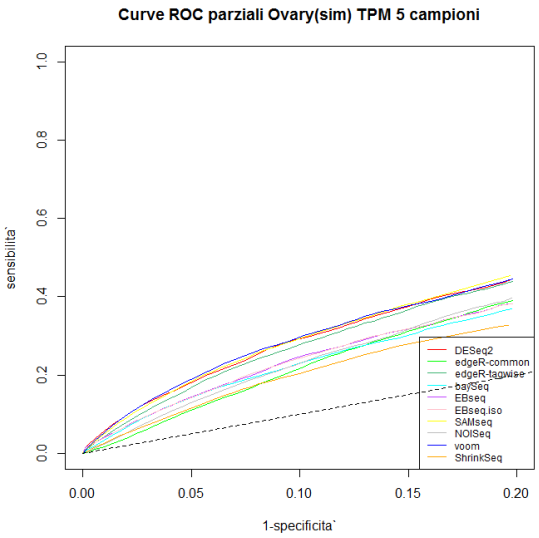
Figura 6.8: Tasso d'errore di I tipo per i quattro metodi che forniscono p-values nominali, nei differenti studi di simulazione nel caso in cui sia applicata la normalizzazione TPM.

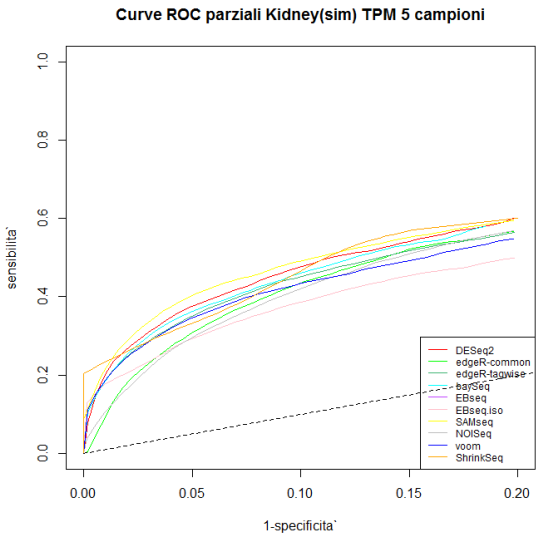
Per quanto riguarda il controllo dell'errore di I tipo (Figura 6.8), i metodi più sensibili alla normalizzazione TPM sono DESeq2 e gli edgeR, mentre voom rimane praticamente inalterato rispetto al caso precedente. Nei data sets simulati in modo non parametrico, i risultati prodotti da DESeq2 appaiono meno variabili (in particolare in Kidney(sim) 5 campioni) e si attestano sotto il 5% anche per numerosità campionarie piccole a differenza di quanto succedeva se la matrice di partenza era costituita da conte grezze. Anche edgeR.cmn si allinea subito con il cut off scelto a differenza di prima dove la percentuale di falsi positivi rimaneva sempre superiore a 0.05 anche all'aumentare della numerosità. Sembra quindi che l'effetto della normalizzazione TPM su questi metodi sia quello di renderli più conservativi e in alcuni casi meno variabili riducendo inoltre l'impatto che il numero dei campioni può avere sui risultati, che si mostrano molto simili per 5 campioni così come per 20. Per i data sets simulati parametricamente invece, i metodi non sembrano subire alcuna variazione rispetto al caso di riferimento, né nella variabilità né nel livello, né rispetto alla dipendenza dalla numerosità campionaria fatta eccezione per DESeq2 il cui livello risulta uniformemente più schiacciato e che quindi migliora il controllo dell'errore di I tipo.
In conclusione sembra che la normalizzazione TPM favorisca il controllo dell'errore di I tipo, in particolare nelle simulazione non parametriche dove probabilmente riesce a rimuovere parte di quella variabilità che deriva delle caratteristiche delle matrici di partenza usate per le simulazioni, diminuendo generalmente la percentuale di falsi positivi presenti nelle liste.

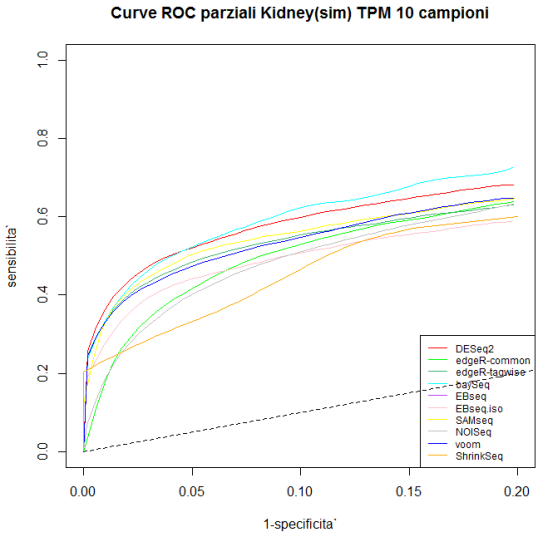
6.6.2 Sensibilità e specificità

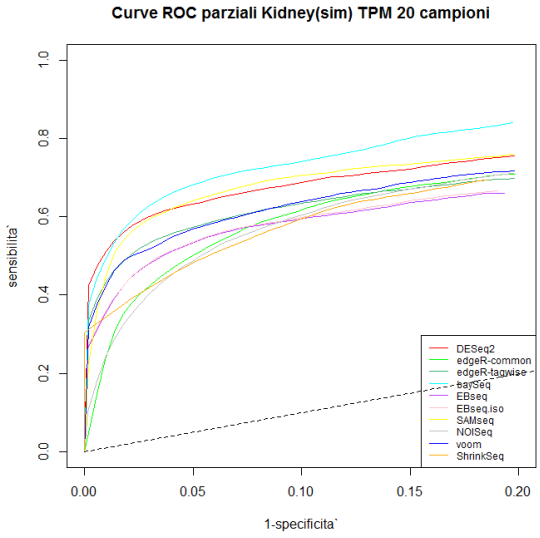
Vogliamo ora verificare se la normalizzazione TPM provoca un'alterazione nel modo in cui i metodi ordinano i geni analizzando le curve ROC (Figura 6.9): complessivamente gli andamenti non cambiano rispetto al caso di riferimento per cui a seconda del tipo di data set le curve saranno più o meno vicine alla bisettrice. Analizzando più in dettaglio il ranking dei metodi in base alle curve medie si può vedere che non coincide perfettamente con lo stesso ranking nel caso di riferimento, anche se, in generale, i metodi migliori rimangono tra i migliori, mentre i peggiori si confermano tra i peggiori. Nelle simulazioni non parametriche, ShrinkSeq è l'unico metodo che per numerosità campionarie ridotte modifica molto la sua posizione finendo tra i peggiori nel caso di Ovary(sim) e tra i migliori nel caso di Kidney(sim) 5 campioni, mentre in Kidney(NB) 20 campioni voom incrementa notevolmente le sue capacità.

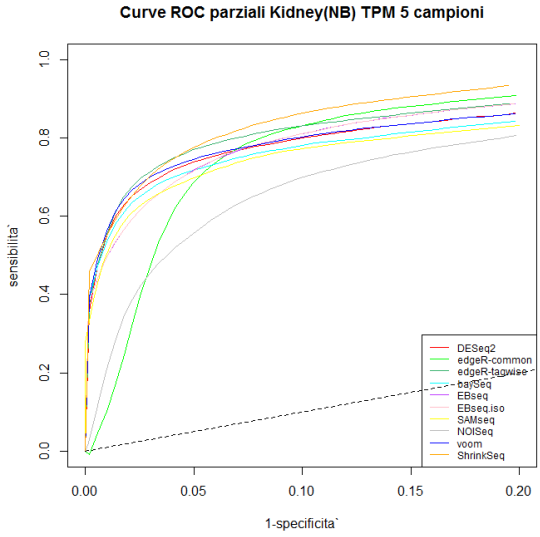
Si analizzino ora le AUC: guardando la Figura 6.10 e confrontandola con la Figura 6.4 si può vedere che i risultati rimangono praticamente inalterati

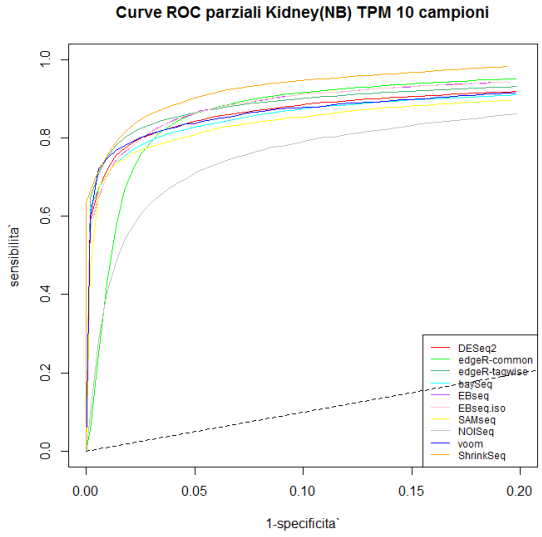

Curve ROC parziali Ovary(sim) TPM 5 campioni
sensibilita`
1-specificita`
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.05
0.10
0.15
0.20
DESeq2
edgeR-common
edgeR-tagwise
baySeq
EBseq
EBseq.iso
SAMseq
NOISeq
voom
ShrinkSeq


Curve ROC parziali Kidney(sim) TPM 5 campioni
sensibilita`
1-specificita`
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.05
0.10
0.15
0.20
DESeq2
edgeR-common
edgeR-tagwise
baySeq
EBseq
EBseq.iso
SAMseq
NOISeq
voom
ShrinkSeq


Curve ROC parziali Kidney(sim) TPM 10 campioni
sensibilita`
1-specificita`
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.05
0.10
0.15
0.20
DESeq2
edgeR-common
edgeR-tagwise
baySeq
EBseq
EBseq.iso
SAMseq
NOISeq
voom
ShrinkSeq


Curve ROC parziali Kidney(sim) TPM 20 campioni
sensibilita`
1-specificita`
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.05
0.10
0.15
0.20
DESeq2
edgeR-common
edgeR-tagwise
baySeq
EBseq
EBseq.iso
SAMseq
NOISeq
voom
ShrinkSeq


Curve ROC parziali Kidney(NB) TPM 5 campioni
sensibilita`
1-specificita`
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.05
0.10
0.15
0.20
DESeq2
edgeR-common
edgeR-tagwise
baySeq
EBseq
EBseq.iso
SAMseq
NOISeq
voom
ShrinkSeq


Curve ROC parziali Kidney(NB) TPM 10 campioni
sensibilita`
1-specificita`
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.05
0.10
0.15
0.20
DESeq2
edgeR-common
edgeR-tagwise
baySeq
EBseq
EBseq.iso
SAMseq
NOISeq
voom
ShrinkSeq

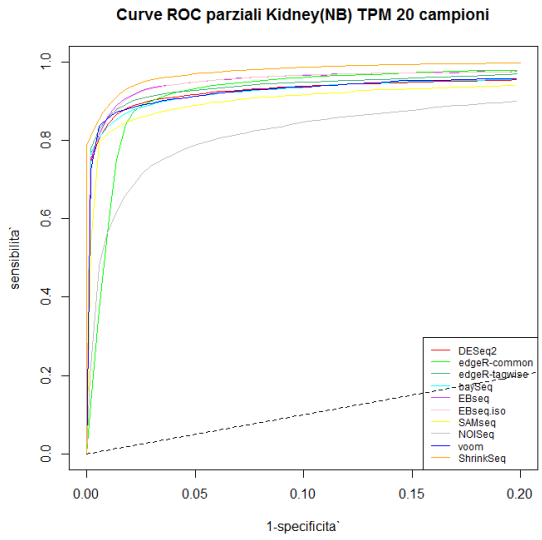


Figura 6.9: Curve ROC per i dieci metodi valutati, nei differenti studi di simulazione nel caso in cui sia applicata la normalizzazione TPM.

rispetto al caso di riferimento sia nei livelli che nella variabilità. Anche il ranking dei metodi nei diversi scenari di simulazione non cambia, quindi sembra che la TPM non modifichi la capacità dei metodi di discriminare tra DE e non. Per quantificare il cambiamento nei livelli medi di AUC e di pAUC tra i risultati prodotti usando le conte grezze e le conte normalizzate con la TPM, nella Tabella 6.3 si è operata la differenza tra i valori di AUC medi e di pAUC medi nei due scenari, calcolati sulle 10 simulazioni per ogni metodo e per ogni disegno di simulazione. Oltre a questa informazione sono riportate anche le deviazioni standard dei valori delle AUC e dei pAUC nel caso in cui i dati siano preventivamente normalizzati con la TPM: facendo un confronto con i valori riportati in Tabella 6.2 si vede che le deviazioni standard sono praticamente identiche nei due scenari e questo conferma quanto visto con i boxplot, ovvero che la variabilità non subisce modifiche sostanziali. Per quanto riguarda le differenze delle AUC e dei pAUC si può notare che tendenzialmente i valori in Tabella 6.3 sono per la maggior parte positivi e questo indica che i metodi sembrano preferire le conte grezze piuttosto che le conte normalizzate per definire un ordinamento; tali differenze sono tuttavia molto piccole rispetto all'entità dei valori delle AUC e dei pAUC rispettivamente, per cui è verosimile pensare che non ci siano molte difformità tra le due tecniche.

In conclusione sembra che complessivamente la normalizzazione TPM non

	Ovary(sim)				Kidney(sim) 5 campioni			
	diff AUC	**sd AUC**	**diff pAUC**	**sd pAUC**	**diff AUC**	**sd AUC**	**diff pAUC**	**sd pAUC**
DESeq2	0.00166	0.063	0.0009	0.016	-0.007	0.046	-0.001	0.016
edgeR.cmn	-0.00527	0.055	-0.0014	0.009	0.017	0.024	-0.004	0.014
edgeR.tgw	0.00002	0.060	0.0006	0.014	0.025	0.024	0.004	0.013
baySeq	0.00073	0.052	0.0004	0.011	0.004	0.042	0.002	0.013
EBSeq	0.00374	0.043	0.0003	0.012	0.016	0.023	0.004	0.011
EBSeq.iso	0.00311	0.043	-0.0003	0.012	0.015	0.023	0.004	0.011
SAMseq	0.00662	0.060	0.0023	0.016	-0.005	0.061	-0.003	0.023
NOISeq	0.00696	0.051	0.0019	0.009	0.013	0.035	0.004	0.014
voom	-0.00351	0.062	-0.0002	0.016	0.002	0.050	0.001	0.016
ShrinkSeq	0.01952	0.066	0.0080	0.015	-0.078	0.030	-0.020	0.021
	Kidney(sim) 10 campioni				**Kidney(sim) 20 campioni**			
	diff AUC	**sd AUC**	**diff pAUC**	**sd pAUC**	**diff AUC**	**sd AUC**	**diff pAUC**	**sd pAUC**
DESeq2	-0.001	0.019	-0.0005	0.007	-0.004	0.036	-0.001	0.013
edgeR.cmn	0.017	0.023	-0.0046	0.010	0.019	0.023	-0.007	0.012
edgeR.tgw	0.031	0.021	0.0056	0.008	0.032	0.024	0.004	0.011
baySeq	0.012	0.030	0.0073	0.013	0.014	0.015	0.009	0.004
EBSeq	0.035	0.023	0.0045	0.007	0.043	0.022	0.005	0.005
EBSeq.iso	0.035	0.023	0.0019	0.007	0.039	0.025	0.009	0.006
SAMseq	0.028	0.042	0.0074	0.015	0.023	0.049	0.005	0.004
NOISeq	0.019	0.024	0.0054	0.009	0.017	0.029	0.007	0.003
voom	-0.003	0.023	0.0020	0.008	0.005	0.037	0.004	0.004
ShrinkSeq	0.006	0.030	0.0017	0.021	0.011	0.029	0.005	0.004
	Kidney(NB) 5 campioni				**Kidney(NB) 10 campioni**			
	diff AUC	**sd AUC**	**diff pAUC**	**sd pAUC**	**diff AUC**	**sd AUC**	**diff pAUC**	**sd pAUC**
DESeq2	0.009	0.013	0.004	0.004	0.011	0.007	0.0042	0.003
edgeR.cmn	0.008	0.012	-0.005	0.004	0.009	0.005	0.0002	0.002
edgeR.tgw	0.011	0.014	0.004	0.004	0.010	0.006	0.0041	0.002
baySeq	-0.005	0.011	0.001	0.004	-0.004	0.007	-0.0008	0.003
EBSeq	0.008	0.012	0.006	0.004	0.009	0.005	0.0034	0.003
EBSeq.iso	0.008	0.012	0.004	0.004	0.008	0.005	0.0027	0.003
SAMseq	0.002	0.010	-0.005	0.004	0.007	0.008	0.0045	0.003
NOISeq	0.004	0.011	0.003	0.003	0.003	0.005	0.0015	0.002
voom	-0.004	0.009	-0.002	0.004	-0.006	0.009	-0.0017	0.003
ShrinkSeq	0.006	0.005	0.004	0.004	-0.007	0.002	0.0051	0.002
	Kidney(NB) 20 campioni							
	diff AUC	**sd AUC**	**diff pAUC**	**sd pAUC**				
DESeq2	0.011	0.006	0.0039	0.002				
edgeR.cmn	0.009	0.003	0.0055	0.002				
edgeR.tgw	0.006	0.004	0.0031	0.002				
baySeq	-0.001	0.005	-0.0004	0.002				
EBSeq	0.009	0.004	0.0031	0.001				
EBSeq.iso	0.007	0.004	0.0025	0.001				
SAMseq	0.008	0.006	0.0033	0.002				
NOISeq	0.003	0.004	0.0012	0.002				
voom	-0.006	0.008	-0.0027	0.002				
ShrinkSeq	-0.009	0.001	-0.0064	0.001				

Tabella 6.3: Tabella con i valori di 1) $\overline{AUC}_{grez} - \overline{AUC}_{TPM}$, 2) standard deviation delle AUC nel caso TPM 3) $\overline{pAUC}_{grez} - \overline{pAUC}_{TPM}$, 4) standard deviation delle pAUC nel caso TPM, sulle 10 simulazioni per ogni metodo e per ogni disegno di simulazione.

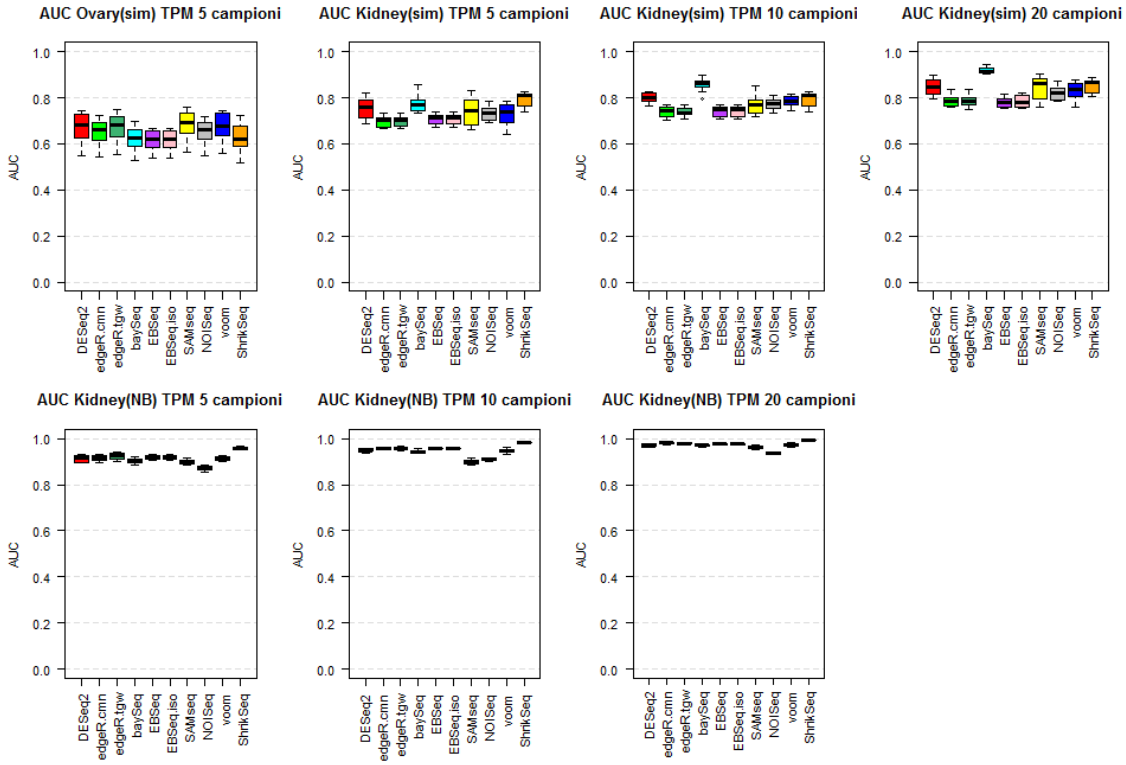


Figura 6.10: Area sotto la curva ROC (AUC) per i dieci metodi valutati, nei differenti studi di simulazione nel caso in cui sia applicata la normalizzazione TPM.

alteri le capacità dei metodi di discriminazione tra geni DE ed EE, e che sebbene i risultati siano praticamente uguali ci sia una leggerissima predilezione verso quelli prodotti a partire dalle conte grezze; l'ordinamento dei geni, in particolare dei maggiormente espressi, cambia lievemente per alcuni metodi, il che implica che il ranking dei metodi in base alle curve ROC medie non sia esattamente lo stesso nei due scenari considerati.

6.6.3 Controllo FDR

Infine si vuole verificare se la normalizzazione TPM determini delle variazioni nel tasso di falsi positivi nelle liste di geni DE prodotte da ciascun metodo, rispetto al caso di riferimento in cui le analisi sono state applicate

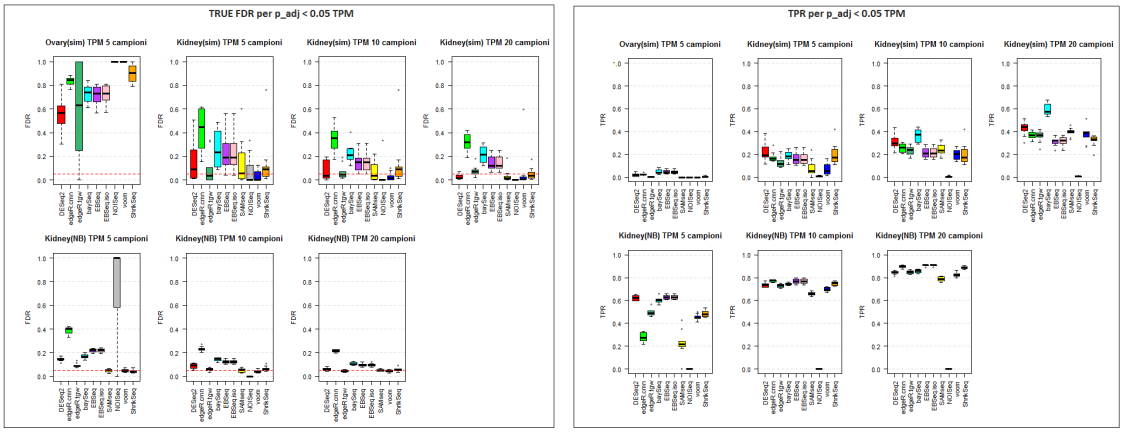


Figura 6.11: Pannello a sinistra: False discovery rates (FDR) osservato per una soglia di FDR imposta a 0.05, per i dieci metodi valutati, nei differenti studi di simulazione nel caso in cui sia applicata la normalizzazione TPM. Pannello a destra: True positive rates (TPR) osservato per una soglia di FDR imposta a 0.05, per i dieci metodi valutati, nei differenti studi di simulazione nel caso in cui sia applicata la normalizzazione TPM.

a partire dalle conte grezze. Si è interessati anche a verificare se eventuali variazioni nel controllo dell'FDR corrispondano a modifiche nella sensibilità dei metodi. Per fare ciò si comparino la Figura 6.11 con le Figure 6.6 e 6.7. In Ovary(sim) si può notare che i livelli e le variabilità degli FDR rimangono praticamente inalterati fatta eccezione per i due metodi che moderano la stima della dispersione: la variabilità di edgeR.tgw aumenta visibilmente peggiorando leggermente anche nel livello mediano, viceversa accade per DESeq2 che migliora riducendo sia livello che variabilità. Il fatto che la normalizzazione TPM non abbia un gran effetto su Ovary lo si può vedere anche dal fatto che la sensibilità dei metodi è praticamente identica a quella del caso di riferimento.
Le stesse considerazioni non valgono per Kidney(sim) su cui la TPM sembra avere l'impatto più considerevole rispetto a tutti gli altri disegni di simulazione: a parte baySeq e voom i cui risultati non sembrano variare, gli altri metodi sono soggetti ad una buona diminuzione della percentuale di falsi positivi nelle loro liste a cui però corrisponde anche un generale abbassamento della percentuale di veri positivi che sono stati trovati, cosa che si era già notata con l'analisi delle curve ROC. Permane quindi la dicotomia tra percentuale di falsi positivi rilevati e sensibilità del metodo, tranne per SAMseq e ShrinkSeq che per campioni piccoli migliorano in entrambi i campi.

Nel caso di Kidney(NB) 5 campioni, i metodi che risultano più sensibili alla normalizzazione sono DESeq2, edgeR.cmn e gli EBSeq che riescono ad operare un controllo dell'FDR migliore rispetto al caso di riferimento, ma mentre il TPR di DESeq2 scende, quelli di edgeR.cmn e di edger.tgw aumentano. SAMseq nel caso delle conte grezze non rilevava alcun gene DE, mentre con la normalizzazione TPM acquisisce una potenza maggiore e sebbene la sua sensibilità risulti inferiore a quella degli altri metodi, il controllo dell'FDR è a livelli ottimali, cosa che comunque costituisce un miglioramento rispetto alla situazione di riferimento. Per numerosità campionarie superiori, il controllo dell'FDR rimane inalterato, mentre la sensibilità subisce delle leggere fluttuazioni, in particolare DESeq2 e SAMseq tendono a diminuire il numero di veri geni DE che rilevano mentre accade il contrario per edgeR.cm.

In definitiva sembra che il pregio della normalizzazione TPM sia quello di permettere a molti metodi di migliore il controllo sull'errore di I tipo e sull'FDR a discapito di una perdita minima di sensibilità che si è riscontrata confrontando i valori di AUC e di pAUC medi. Come previsto non tutti i metodi risultano ugualmente sensibili a tale normalizzazione cosa che può modificare il ranking dei metodi stessi: i risultati di voom, ad esempio, non sembrano scostarsi molto da quelli ottenuti con lo stesso metodo a partire dalle conte grezze, mentre altri come DESeq2, edgeR.cmn e ShrinkSeq presentano leggere differenze nei risultati. Questo può essere spiegato se si considera che la potenza dei metodi basati sulla Binomiale Negativa dipende non solo dal numero di repliche, ma anche dall'entità delle conte, per cui geni con valori di conta maggiori hanno più potenza e sono più facilmente trovati DE. L'effetto della normalizzazione TPM è quello di abbassare e rendere più uniformi le conte, per cui è possibile che i metodi basati sulla NB siano maggiormente influenzati dalla trasformazione. Dati questi esiti risulta che la normalizzazione TPM non altera sostanzialmente la bontà dei metodi lasciando l'ordinamento dei geni quasi inalterato rispetto al caso di riferimento e limitando le differenze al minimo.

6.7 Caso reale

Considerando quanto fin'ora scoperto sui metodi per l'analisi della differenziale espressione, vogliamo metterli alla prova su un caso di studio reale. Si è analizzato il data set Ovary senza escludere preventivamente alcun gene. Si ricorda che Ovary è composto dai valori di espressione dei geni di 28 soggetti, 14 dei quali sono risultati resistenti alle terapie farmacologiche per la cura del tumore all'ovaio (denominati come "Resistenti") e altri 14 hanno mostrato una reazione sensibile al farmaco (denominati come "Sensibili").

Dunque in un contesto reale, la domanda in oggetto di studio sarebbe: "Quali sono quei geni che influiscono sull'impatto della terapia farmacologica per la cura del tumore all'ovaio?". Dopo aver escluso la presenza di un bias di GC e dopo aver effettuato una fase di filtraggio in cui si sono eliminati quei geni la cui somma delle conte dei campioni appartenenti allo stesso gruppo di trattamento è risultata inferiore a 10 in almeno una delle due condizioni, il data set contiene 26131 geni. A questo data set si sono applicati i dieci metodi, per trovare i geni che mostravano differenziale espressione tra le due condizioni: tutti i geni con una stima dell'FDR o del BFDR sotto 0.05 sono stati considerati DE. Non è chiaro quale sia la soglia da imporre al q-value riportato da NOISeq affinché questo risulti equivalente alle stime dell'FDR o ai p-value aggiustati degli altri metodi. Come già dimostrato in precedenza, se posto alla stesse condizioni degli altri metodi, l'lfdr non da risultati sensati per cui anche se non si è escluso NOISeq dalle analisi si tenga conto del fatto che i risultati potrebbero non essere soddisfacenti.

Per prima cosa si è paragonata la numerosità delle liste di geni DE trovati da ciascun metodo (Figura 6.12 A). Il maggior numero di geni DE è stato trovato da edgeR.cmn, seguito da baySeq: per quello che riguarda edgeR.cmn si è precedentemente mostrato tramite le simulazioni che questo metodo risulta inadeguato sia in termini di TPR sia in termini di controllo dell'FDR per cui ci si aspetta che una buona parte dei geni presenti nella lista sia un falso positivo. Al contrario, in quasi tutte le simulazioni, baySeq è risultato essere uno dei metodi con sensibilità maggiore a discapito di una percentuale di falsi positivi tra le più alte, ma che comunque era in linea con quelle degli altri metodi. Come previsto NOISeq non lavora bene e non restituisce alcun gene DE. Anche voom restituisce un unico gene DE, ma questo risultato non è sorprendente: in fase di simulazione si è dimostrato che voom è un metodo molto conservativo, con il pregio di riuscire a controllare l'FDR a un livello prefissato in una varietà di assetti, ma che presenta lo svantaggio di avere una sensibilità inferiore rispetto agli altri metodi a parità di soglia. In poche parole la sua caratteristica è quella di avere liste di geni più corte, ma composte per la maggior parte da geni realmente DE. Poiché si sono presi in analisi soggetti che sono tutti affetti della stessa patologia, in questo data set ci si aspetta che l'espressione tra le due condizioni sia abbastanza omogenea, e che quindi non ci sia molta differenziale espressione tra Sensibili e Resistenti al farmaco, per cui voom in virtù della sua capacità di controllare il numero di falsi positivi, rileverà unicamente quei geni di cui è molto evidente la differenziale espressione e che in questo caso però sono molto pochi. Pur essendo metodi di natura diversa, in fase di simulazione SAMseq e voom hanno mostrato caratteristiche simili in contesti con numerosità elevata come in questo caso, per cui non stupisce che la numerosità della lista di SAMseq

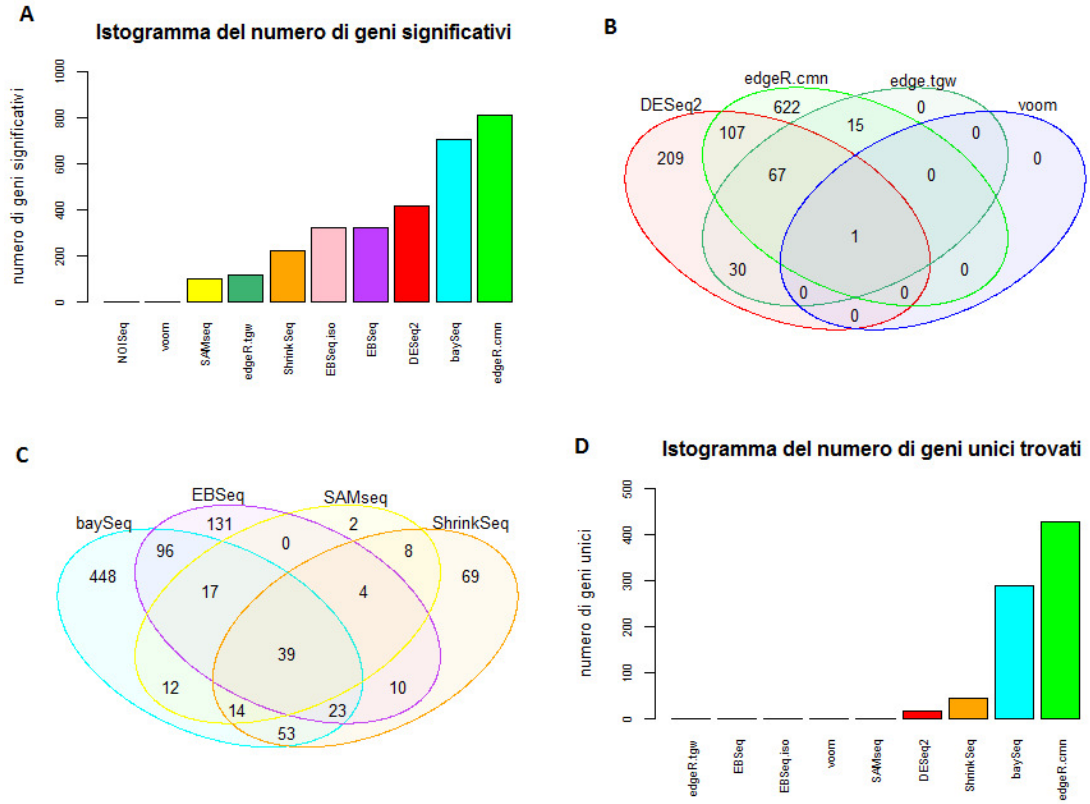


Figura 6.12: A) Istogramma del numero di geni significativi; B) Intersezione delle liste di DESeq2, edgeR.cmn, edgeR.tgw e voom; C) Intersezione delle liste di baySeq, EBSeq, SAMseq e ShrinkSeq; D) Istogramma del numero di geni unici trovati.

risulti piuttosto contenuta come quella di voom. Per quel che riguarda gli altri metodi non ci sono deviazioni dal comportamento che ci si attendeva in base alle simulazioni: i metodi più conservativi come edgeR.tgw e ShrinkSeq, che spesso avevano livelli simili di FDR e di TPR nelle simulazioni, hanno liste più contenute, EBSeq ed EBSeq.iso si trovano come sempre ad un livello intermedio, mentre DESeq2, che si è visto nelle simulazioni rilassava il controllo dell'FDR per ottenere una percentuale maggiore di DE rilevati, ha una lista più lunga.

Successivamente si è studiata la sovrapposizione delle liste dei geni indicati come DE tra i vari metodi. La Figura 6.12 B mostra la sovrapposizione tra gli insiemi di geni differenzialmente espressi trovati da DESeq2, edgeR.cmn, edgeR.tgw e voom (sono stai inclusi solo quattro metodi per

rendere il diagramma di Venn più interpretabile). Ovviamente, per quanto detto in precedenza, l'unico gene trovato da voom è condiviso anche dagli altri metodi. edgeR.tgw condivide la totalità dei geni che trova con DESeq2 e edgeR.cmn, e dunque nessun gene viene rilevato unicamente da questo metodo. Sia DESeq2 che edgeR.tgw inglobano l'informazione derivante dalla totalità dei geni per ottenere stime moderate della dispersione e utilizzano test classici per l'analisi della differenziale espressione, per cui non stupisce che di 113 geni rilevati da edgeR.tgw 98 siano in comune con DESeq2. Lo stesso si può dire dei due edgeR che si differenziano solo per la stima della dispersione. Per quanto riguarda DESeq2 e edgeR.cmn, due tra i metodi meno conservativi, rilevano solo 175 geni in comune, il che implica che la maggior parte delle loro liste è composta da geni che sono rilevati unicamente da loro (209 per DESeq2 e 622 per edgeR.cmn).
La Figura 6.12 C mostra la corrispondente comparazione per i restanti metodi ovvero baySeq, EBSeq, SAMseq e ShrinkSeq. Le liste dei due EBSeq sono praticamente equivalenti, quindi si è scelto di omettere EBSeq.iso per rendere più leggibile il diagramma. In questo caso risulta già più evidente come questi metodi siano molto diversi tra loro ed assumano un impianto teorico piuttosto differente: solo la lista di SAMseq risulta, per la maggior parte, un sottoinsieme della lista di baySeq, mentre negli altri casi la condivisione delle liste è solo parziale. Tranne che per SAMseq, gli altri metodi hanno un'alta percentuale di geni "unici", ovvero geni che non sono condivisi dagli altri metodi e questa è una prova ulteriore del fatto che questi metodi tendano a favorire aspetti differenti.
Infine si riporta un istogramma del numero di geni "unici" rilevati da ciascun metodo rispetto a tutti gli altri metodi (Figura 6.12 D). Come già visto edgeR.tgw e voom non hanno geni unici perché assorbiti da DESeq2 e edgeR.cmn; EBSeq ed EBSeq.iso possiedono praticamente la stessa lista dunque anche in questo caso il loro corrispondente valore va a zero. SAMseq condivide la lista principalmente con baySeq e ShrinkSeq lasciando solo 2 geni esclusivamente rilevati da questo metodo. DESeq2 condivide tre quarti della lista con baySeq, mentre il resto è assorbito da diversi metodi lasciando solo 17 geni "unici", mentre la lista di ShrinkSeq è gran parte contenuta in quella di DESeq2. I due metodi con le liste più lunghe anche in questo caso mostrano un numero di geni DE unicamente rilevati molto alto: il fatto che un così elevato numero di geni non venga rilevato da nessun altro metodo è un indice che quei geni sono probabilmente falsi positivi, il che confermerebbe quanto visto con le simulazioni. La Tabella 6.4 mostra le intersezioni tra le liste dei geni DE per ciascuna coppia di metodi.

Per caratterizzare l'insieme dei geni preferenzialmente chiamanti DE dai differenti metodi, si sono evidenziati i geni DE in un MA plot. L'MA plot

	DESeq2	edgeR.cmn	edgeR.tgw	baySeq	EBSeq	EBSeq.iso	SAMseq	voom	ShrinkSeq
DESeq2	**414**	175	98	300	167	167	86	1	165
edgeR.cmn	175	**812**	83	219	246	246	26	1	21
edgeR.tgw	98	83	**113**	111	88	86	34	1	45
baySeq	300	219	111	**702**	175	174	82	1	129
EBSeq	167	246	88	175	**320**	317	60	1	76
EBSeq.iso	167	246	86	174	317	**319**	58	1	75
SAMseq	86	26	34	82	60	58	**96**	1	65
voom	1	1	1	1	1	1	1	**1**	1
ShrinkSeq	165	21	45	129	76	75	65	1	**220**

Tabella 6.4: Tabella del numero di geni differenzialmente espressi che sono condivisi da ciascuna coppia di metodi. Il numero sulla diagonale, evidenziato in grassetto, indica il numero totale di geni differenzialmente espressi trovati dai rispettivi metodi.

è stato costruito come segue: per ogni gene si è fatta una media delle conte dei campioni appartenenti rispettivamente alla condizione Sensibili (S) e Resistenti (R), dopo di che si sono ricavati i valori

$$M = log_2(S/R)$$

$$A = 0.5\, log_2(SR)$$

ovvero il valore della media del logaritmo delle espressioni normalizzate e del LFC.

Osservando la Figura 6.13 si può notare quanto già sospettato, ovvero che il data set mostra una differenziale espressione piuttosto omogenea, con pochi elevati LFC solo in corrispondenza di valori di espressione medi. I risultati mostrano chiaramente che per quasi tutti i metodi, tranne che per edgeR.cmn e gli EBSeq, nessun gene viene identificato come DE tra quelli con bassa espressione media, mentre per i tre metodi rimanenti è necessario un *fold change* più elevato rispetto agli altri geni affinché vengano rilevati come DE. Per quasi tutti i metodi inoltre, i DE rilevati sono sopratutto geni con valore di espressione medi: solo DESeq2 e baySeq riescono a identificare qualche gene DE tra quelli con elevata espressione. SAMseq, voom e ShrinkSeq rilevano la differenziale espressione principalmente in una direzione e i loro DE sembrano concentrarsi in una range di *fold change* determinato, circa (-2, 0), oltre il quale nessun gene viene più considerato come DE. Per quanto riguarda DESeq2, baySeq e ShrinkSeq richiedono un *fold change* più basso per rilevare la differenziale espressione, viceversa accade per edgeR.tgw che risulta il metodo che necessita di *fold change* più elevati per dichiarare un gene DE. Infine edgeR.cmn e gli EBSeq, a parte qualche eccezione, sembrano separare chiaramente gli insieme dei DE dagli EE.

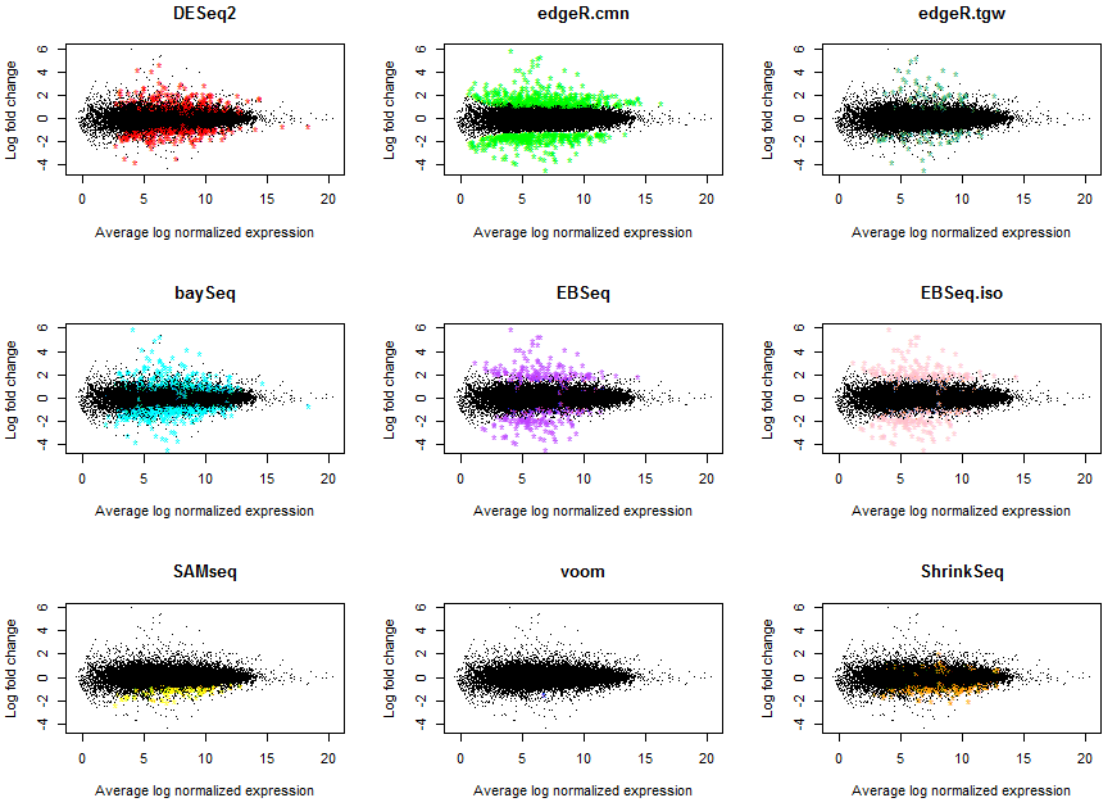


Figura 6.13: MA plot che riportano il livello di espressione (sull'asse x) e il livello di differenziale espressione tra le due condizioni (sull'asse y), con i geni rilevati DE evidenziati.

	DESeq2	edgeR.cmn	edgeR.tgw	baySeq	EBSeq	EBSeq.iso	SAMseq	NOISeq	voom	ShrinkSeq
DESeq2	**1**	0.86	0.97	0.64	0.68	0.68	0.79	0.82	0.81	0.84
edgeR.cmn	0.86	**1**	0.91	0.61	0.77	0.77	0.66	0.82	0.69	0.87
edgeR.tgw	0.97	0.91	**1**	0.67	0.71	0.71	0.78	0.85	0.80	0.87
baySeq	0.64	0.61	0.67	**1**	0.70	0.70	0.59	0.53	0.61	0.66
EBSeq	0.68	0.77	0.71	0.70	**1**	1	0.60	0.55	0.60	0.77
EBSeq.iso	0.68	0.77	0.71	0.70	1	**1**	0.60	0.55	0.60	0.77
SAMseq	0.79	0.66	0.78	0.59	0.60	0.60	**1**	0.65	0.90	0.70
NOISeq	0.82	0.82	0.85	0.53	0.55	0.55	0.65	**1**	0.66	0.73
voom	0.81	0.69	0.80	0.61	0.60	0.60	0.90	0.66	**1**	0.73
ShrinkSeq	0.84	0.87	0.87	0.66	0.77	0.77	0.70	0.73	0.73	**1**

Tabella 6.5: Tabella dei coefficienti di correlazione di Spearman per ogni coppia di metodi.

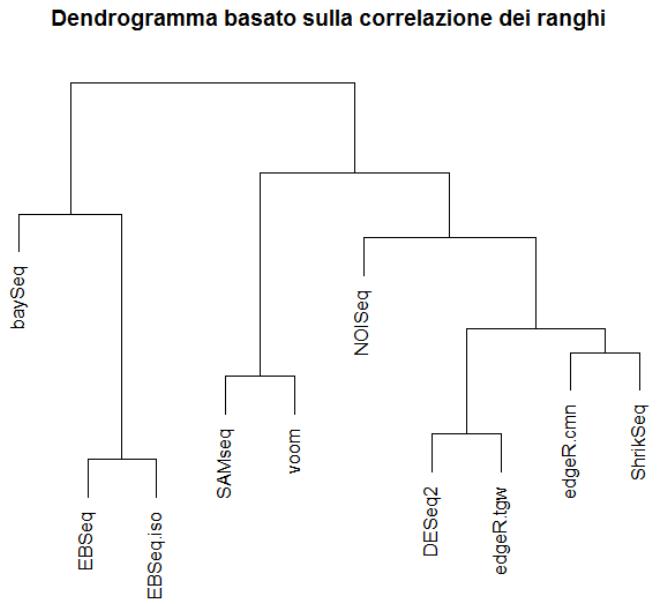


Figura 6.14: Dendrogramma informativo della similarità nell'ordinare i geni tra i metodi.

In conclusione per verificare se ci siano delle similitudini nell'ordine dei geni tra i vari metodi, si è calcolato il coefficiente di correlazione per ranghi di Spearman per ogni coppia di metodi sulla base dei quali si è applicato un algoritmo di *clustering* gerarchico con legame completo per stimare il dendrogramma in Figura 6.14. Dalla Tabella 6.5 si può notare che DESeq2, edgeR.cmn, edgeR.tgw, NOISeq e ShrinkSeq tendono a ordinare i geni in modo simile, con coefficienti di correlazione superiori 0.80. Da segnalare, in particolare la correlazione tra DESeq2 e edgeR.tgw che arriva quasi a 1, ma ciò non sorprende dato che sono entrambi metodi che moderano le stime della dispersione e usano test classici per rilevare la differenziale espressione. Anche l'ordinamento dei geni operato da SAMseq e voom è molto simile con una correlazione pari a 0.90: i due metodi, pur essendo teoricamente molto lontani, si comportano in maniera abbastanza simile per numerosità campionarie elevate come mostrato nelle simulazioni. voom sembra essere correlato anche a DESeq2 e ad edgeR.tgw. Gli ordinamenti ottenuti da baySeq e dagli EBSeq invece, non risultano particolarmente simili a nessuno degli altri metodi considerati.

Capitolo 7

Conclusioni

In questo studio, si sono valutati e comparati dieci metodi per l'analisi della differenziale espressione di dati di RNA-Seq in differenti condizioni sperimentali che variano per numerosità campionaria, tecnica di simulazione utilizzata e eterogeneità del data set.
Nessuno dei metodi valutati è risultato ottimale in tutte le circostanze, dunque la scelta del metodo in una particolare situazione dipende dalle condizioni sperimentali. La prima cosa che si è notata è la differenza nei risultati tra le simulazioni non parametriche e quelle parametriche: le prime riescono effettivamente a mimare un caso reale, per cui i dati simulati ereditano parte delle caratteristiche particolari del data set reale a partire dal quale le conte sono state simulate e risultano dunque più interessanti a fini di studio. I risultati delle simulazioni parametriche sono invece influenzati dalla distribuzione generativa dei dati che è condivisa dalla maggior parte dei metodi per l'analisi della differenziale espressione, svantaggiando in questo modo i metodi non parametrici.
Quando il numero di replicati è basso (minore o uguale a 5) i risultati dei test statistici devono essere sempre presi con cautela, poiché possono condurre a una percentuale di falsi positivi che supera ampiamente la soglia imposta dal cut off di FDR scelto. È importante tenere in considerazione questo perché nella pratica è piuttosto comune avere pochi replicati biologici per condizione. Per i metodi parametrici il problema può essere dovuto ad una certa inaccuratezza nella stima dei parametri di media e dispersione. In questi casi si consiglia di utilizzare metodi che moderano la stima della dispersione, come DESeq2 e edgeR.tgw, che sfruttano l'informazione condivisa da tutti i geni: in questo studio sono infatti risultati i migliori sia nel coniugare la più bassa percentuale di falsi positivi con una sensibilità tra le più alte, sia nell'ordinare correttamente i geni in base alla loro differenziale espressione. Se ad una scarsa numerosità campionaria è associata anche una differenziale

espressione limitata, i risultati saranno liste con pochi geni DE per la maggior parte falsi positivi per cui, dato che l'effetto che si vuole rilevare è piccolo, in questi casi è consigliabile aumentare la numerosità campionaria al fine di non ottenere risultati fuorvianti.
Tenere conto delle peculiarità dei dati, come l'eterogeneità, risulta quindi di fondamentale importanza ed è auspicabile per gli utenti familiarizzare con le caratteristiche generali tra e all'interno di ciascun gruppo di campioni utilizzando metodi di visualizzazione e di valutazione della qualità prima di scegliere lo strumento di analisi.
Quando il numero dei replicati diventa relativamente grande (10 o più campioni per gruppo di trattamento), la scelta del metodo diventa meno critica, ma comunque largamente influenzata dalle caratteristiche del data set. In generale si può vedere che se le assunzioni sulla distribuzione dei dati sono violate e la numerosità è sufficientemente elevata, i metodi non parametrici risulatano piuttosto validi, in particolare SAMseq il cui controllo dell'FDR risulta ottimale così come la sua capacità di discriminare geni DE ed EE. NOISeq presenta lo svantaggio di non avere una stima dell'FDR affidabile per cui non è possibile ottenere una lista di geni DE nel modo convenzionale. voom è tra i metodi più conservativi riuscendo ad operare generalmente bene sotto diverse condizioni sperimentali: il suo merito principale è quello di riuscire a tenere sotto controllo la percentuale di falsi positivi nelle liste a discapito però di un TRP inferiore agli altri metodi a parità di FDR. Questo metodo ha infatti bisogno di una buona differenziale espressione e di una certa numerosità campionaria per avere sufficiente potenza per rilevare qualche gene differenzialmente espresso; si consideri ad esempio il caso reale qui trattato dove nonostante una numerosità di 14 campioni per gruppo di trattamento solo un gene è stato rilevato come DE a causa dell'omogeneità del data set oggetto di studio. Le prestazioni di baySeq risultano molto variabili e altamente dipendenti dai dati: in virtù del suo impianto teorico è risultato il metodo migliore con data sets eterogenei, rilassando il controllo dell'FDR per ottenere una maggiore sensibilità. Le stesse considerazioni si possono fare per ShrinkSeq la cui performance si è rivelata la migliore per i dati simulati parametricamente, mentre è risultata più carente con gli altri data sets. Sembra quindi che tenere in considerazione la zero-inflation quando i dati sono distribuiti come una Binomiale Negativa sia una strategia vincente, oltre ad avere il pregio di mantenere un buon controllo dell'FDR in quasi tutte le situazioni prese in considerazione. Per gli EBSeq è stato difficile testare le loro reali potenzialità dato che non siamo riusciti a simulare quell'incertezza nella fase di quantificazione dell'espressione delle isoforme che il modello si prefigge di modellare. Questi metodi non si sono mai distinti particolarmente e presentano un controllo dell'FDR un po' troppo liberale senza avere un

incremento della sensibilità di molto superiore rispetto agli altri metodi. Il metodo peggiore in assoluto è risultato edgeR.cmn a causa della stima della dispersione comune che appare inadeguata in quasi tutti i contesti simulati: anche con numerosità elevate il controllo dell'FDR risulta pessimo, per cui le liste che produce saranno composte da molti falsi positivi. Per quanto riguarda i già citati DESeq2 e edgeR.tgw, anche per numerosità campionarie più elevate confermano avere le stesse buone proprietà che abbiamo descritto per i data sets con pochi replicati.
Per garantire la precisione dei risultati ottenuti, può essere informativo eseguire le analisi con più di un pacchetto software in modo che l'intersezione o l'unione delle liste ottenute ci dia conferma di quali sono i geni differenzialmente espressi in comune a tutti i metodi o in modo da aumentare le liste se i metodi rilevano diversi aspetti della differenziale espressione. Dai dendrogrammi si è visto infatti che DESeq2, edgeR.cmn, edgeR.tgw e NOISeq hanno dei modi simili di ordinare i geni in base alla loro differenziale espressione, così come SAMseq e voom anche se gli insiemi di geni rilevati significativamente differenti tra le condizioni per una certa soglia di FDR variano considerevolmente tra i metodi a causa dei diversi modi di stimare la dispersione o l'FDR.
Come ulteriore obbiettivo questo elaborato si è proposto di indagare se la normalizzazione TPM modifichi le prestazioni fornite dai metodi per l'analisi della differenziale espressione. Da quanto si è riscontrato sembra che il pregio della normalizzazione TPM sia quello di permettere a molti metodi di migliorare il controllo sull'errore di I tipo e sull'FDR a discapito di una leggera diminuzione della sensibilità. Non tutti i metodi risultano ugualmente sensibili a tale normalizzazione cosa che può modificare il ranking dei metodi stessi: i risultati di voom, ad esempio, non sembrano scostarsi molto da quelli ottenuti con lo stesso metodo a partire dalle conte grezze, mentre altri software, come DESeq2, edgeR.cmn e ShrinkSeq presentano leggere differenze nei risultati. Queste possono dipendere dal fatto che la normalizzazione TPM, uniformando e abbassando il valore delle conte, incide sulla potenza dei metodi basati sulla NB influenzando maggiormente questi ultimi. Dati questi esiti risulta che la normalizzazione TPM non altera sostanzialmente la bontà dei metodi lasciando l'ordinamento dei geni quasi inalterato rispetto al caso di riferimento.
In conclusione, in fase di applicazione reale, si consiglia di scegliere uno o più metodi per l'analisi della differenziale espressione tenendo conto sopratutto del tipo di dato e della numerosità campionaria che si ha a disposizione.

Bibliografia

[1] U. Nagalakshmi et al. The transcriptional landscape of the yeast genome defined by RNA sequencing. *Science (New York, N.Y.)*, 320(5881), Jun 6 2008.

[2] J. C. Venter et al. The sequence of the human genome. *Science (New York, N.Y.)*, 291(5507), Feb 16 2001.

[3] R. Bourgon, R. Gentleman, and W. Huber. Independent filtering increases detection power for high-throughput experiments. *Proceedings of the National Academy of Sciences of the United States of America*, 107, May 25 2010.

[4] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the royal statistical society.Series B (Methodological)*, 1995.

[5] W. Talloen, S. Hochreiter, L. Bijnens, A. Kasim, Z. Shkedy, D. Amaratunga, and H. Gohlmann. Filtering data from high-throughput experiments based on measurement reliability. *Proceedings of the National Academy of Sciences of the United States of America*, 107, Nov 16 2010.

[6] A. Oshlack, M. D. Robinson, and M. D. Young. From RNA-seq reads to differential expression results. *Genome biology*, 11, 2010.

[7] J. H. Bullard, E. Purdom, K. D. Hansen, and S. Dudoit. Evaluation of statistical methods for normalization and differential expression in mRNA-Seq experiments. *BMC bioinformatics*, 11, 2010.

[8] M. Dillies and A. Rau et al. A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis. *Briefings in bioinformatics*, 14, 2013.

[9] S. Anders and W. Huber. Differential expression analysis for sequence count data. *Genome biology*, 11, 2010.

[10] J. Li, D. M. Witten, I. M. Johnstone, and R. Tibshirani. Normalization, testing, and false discovery rate estimation for RNA-sequencing data. *Biostatistics*, 13, 2012.

[11] M. D. Robinson and A. Oshlack. A scaling normalization method for differential expression analysis of RNA-seq data. *Genome biology*, 11, 2010.

[12] B. M. Bolstad, R. A. Irizarry, M. Åstrand, and Te. P. Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19, 2003.

[13] Y. H. Yang and N. P. Thorne. Normalization for two-color cDNA microarray data. *Lecture Notes-Monograph Series*, 2003.

[14] A. Mortazavi and B. A. Williams et al. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nature methods*, 5, 2008.

[15] RNA-seq blog, RPKM, FPKM and TPM, clearly explained, 2015. `http://www.rna-seqblog.com/rpkm-fpkm-and-tpm-clearly-explained/`.

[16] M. I. Love, W. Huber, and S. Anders. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome biology*, 15, 2014.

[17] M. D. Robinson and G. K. Smyth. Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics*, 23, 2007.

[18] T. J. Hardcastle and K. A. Kelly. baySeq: empirical Bayesian methods for identifying differential expression in sequence count data. *BMC bioinformatics*, 11, 2010.

[19] N. Leng and J. A. Dawson et al. EBSeq: an empirical Bayes hierarchical model for inference in RNA-seq experiments. *Bioinformatics*, 29, 2013.

[20] M. A. Van De Wiel et al. Bayesian analysis of RNA sequencing data by estimating multiple shrinkage priors. *Biostatistics*, 14, 2013.

[21] J. Li and R. Tibshirani. Finding consistent patterns: a nonparametric approach for identifying differential expression in RNA-Seq data. *Statistical methods in medical research*, 22, 2013.

[22] S. Tarazona and P. Furió-Tarí et al. Data quality aware analysis of differential expression in RNA-seq with NOISeq R/Bioc package. *Nucleic acids research*, 43, 2015.

[23] C. W. Law, Y. Chen, W. Shi, and G. K. Smyth. Voom: precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome biology*, 15, 2014.

[24] F. Seyednasrollah, A. Laiho, and L. L. Elo. Comparison of software packages for detecting differential expression in RNA-seq studies. *Briefings in bioinformatics*, 16, 2013.

[25] G. K. Smyth. Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical applications in genetics and molecular biology*, 3, 2004.

[26] P. McCullagh and J. A. Nelder. *Generalized linear models*, volume 2. ed. Chapman e Hall/CRC, London, 1989.

[27] S. Anders, A. Reyes, and W. Huber. Detecting differential usage of exons from rna-seq data. *Genome research*, 22, 2012.

[28] D. Bottomly et al. Evaluating gene expression in C57BL/6J and DBA/2J mouse striatum using RNA-Seq and microarrays. *PloS one*, 6, 2011.

[29] D. R. Cox and N. Reid. Parameter orthogonality and approximate conditional inference. *Journal of the Royal Statistical Society.Series B (Methodological)*, 1987.

[30] Y. Pawitan. *In all likelihood: statistical modelling and inference using likelihood*. Oxford University Press, 2001.

[31] D. J. McCarthy, Y. Chen, and G. K. Smyth. Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic acids research*, 40, 2012.

[32] Y. Zhou, K. Xia, and F.A. Wright. A powerful and flexible approach to the analysis of RNA sequence count data. *Bioinformatics*, 27, 2011.

[33] H. Wu, C. Wang, and Z. Wu. A new shrinkage estimator for dispersion improves differential expression detection in RNA-seq data. *Biostatistics*, 14, 2012.

[34] Y. Di et al. The NBP negative binomial model for assessing differential gene expression from RNA-Seq. *Statistical Applications in Genetics and Molecular Biology*, 10, 2011.

[35] M. Abramowitz and I. A. Stegun. *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*, volume 55. Courier Corporation, 1964.

[36] M. Y. Park. *Generalized linear models with regularization*, 2006.

[37] E. Cule, P. Vineis, and M. De Iorio. Significance testing in ridge regression for genetic data. *BMC bioinformatics*, 12, 2011.

[38] X. Wang. Approximating Bayesian inference by weighted likelihood. *Canadian Journal of Statistics*, 34, 2006.

[39] J. Lu, J. K. Tomfohr, and T. B. Kepler. Identifying differential expression in multiple SAGE libraries: an overdispersed log-linear model approach. *BMC bioinformatics*, 6, 2005.

[40] M. D. Robinson and G. K. Smyth. Small-sample estimation of negative binomial dispersion, with applications to SAGE data. *Biostatistics*, 9, 2007.

[41] J. A. Nelder. Quasi-likelihood and pseudo-likelihood are not the same thing. *Journal of Applied Statistics*, 27, 2000.

[42] I. Lönnstedt, R. Rimini, and P. Nilsson. Empirical bayes microarray ANOVA and grouping cell lines by equal expression levels. *Statistical Applications in Genetics and Molecular Biology*, 4, 2005.

[43] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society.Series B (methodological)*, 1977.

[44] H. Rue, S. Martino, and N. Chopin. Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the royal statistical society: Series b (statistical methodology)*, 71, 2009.

[45] L. Dümbgen and K. Rufibach. logcondens: Computations related to univariate log-concave density estimation. *Journal of Statistical Software, to appear*, 2010.

[46] I. Lönnstedt and T. Speed. Replicated microarray data. *Statistica Sinica*, 2003.

[47] A. Lewin, N. Bochkina, and S. Richardson. Fully Bayesian mixture model for differential gene expression: simulations and model checks. *Statistical applications in genetics and molecular biology*, 6, 2007.

[48] M. Ventrucci, E. M. Scott, and D. Cocchi. Multiple testing on standardized mortality ratios: a Bayesian hierarchical model for FDR estimation. *Biostatistics*, 12, 2010.

[49] B. Efron, R. Tibshirani, J. D. Storey, and V. Tusher. Empirical Bayes analysis of a microarray experiment. *Journal of the American statistical association*, 96, 2001.

[50] G. W. Oehlert. A note on the delta method. *Am Statistician*, 46, 1992.

[51] W. S. Cleveland. Robust locally weighted regression and smoothing scatterplots. *Journal of the American statistical association*, 74, 1979.

[52] A. Oshlack, D. Emslie, L. M. Corcoran, and G. K. Smyth. Normalization of boutique two-color microarrays with a high proportion of differentially expressed probes. *Genome biology*, 8, 2007.

[53] J. D. Storey. *False discovery rate*. International encyclopedia of statistical science. Springer, 2011.

[54] H. Abdi. Bonferroni and Šidák corrections for multiple comparisons. *Encyclopedia of measurement and statistics*, 3, 2007.

[55] Cancer Genome Atlas Research Network. Comprehensive molecular characterization of clear cell renal cell carcinoma. *Nature*, 499, 2013.

[56] J. A. Robles et al. Efficient experimental design and analysis strategies for the detection of differential expression using RNA-Sequencing. *BMC genomics*, 13, 2012.

[57] R. Bi and P. Liu. Sample size calculation while controlling false discovery rate for differential expression analysis with RNA-sequencing experiments. *BMC bioinformatics*, 17, 2016.

[58] S. Benidt and D. Nettleton. SimSeq: a nonparametric approach to simulation of RNA-sequence datasets. *Bioinformatics*, 31, 2015.

[59] K. Strimmer. fdrtool: a versatile R package for estimating local and tail area-based false discovery rates. *Bioinformatics*, 24, 2008.

[60] K. Strimmer. A unified approach to false discovery rate estimation. *BMC bioinformatics*, 9, 2008.

[61] B. Li and C. N. Dewey. RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC bioinformatics*, 12, 2011.

[62] C. Soneson and M. Delorenzi. A comparison of methods for differential expression analysis of RNA-seq data. *BMC bioinformatics*, 14, 2013.

[63] AIRC: Tumore ovaie, 2017. `http://www.airc.it/tumori/tumore-all-ovaio.asp`.

[64] FIGO: International Federation of Gynecology and Obstetrics, 2017. `http://www.figo.org/`.

[65] N. A. Campbell et al. a cura di C. Donati, M. G. Romanelli, and N. Taddei and. *Campbell*, volume 10. ed. Pearson Italia, Milano Torino, 2015.

[66] P. H. Raven and G. B. Johnson. *Biology*, volume 4. ed. Wm. C. Brown Publ, Dubuque (IA.), 1996.

[67] A. Apolloni. Confronto di metodi statistici per la misura dell'espressione differenziale in dati di RNA sequencing. `http://tesi.cab.unipd.it/39543/`.

[68] Wikipedia, espressione genica, 2017. `https://it.wikipedia.org/wiki/Espressione_genica/`.

[69] J. G. Scott and J. O. Berger. An exploration of aspects of Bayesian multiple testing. *Journal of statistical planning and inference*, 136, 2006.

[70] E. Magro. Integrazione di livelli di espressione e metilazione genica attraverso l'analisi di pathway.