

DESeq2 - Analysing RNA-seq data

Tesina per il corso di Statistica Applicata e Analisi dei Dati A.A. 2024/2025

Claudia Costa, matr. 153256

26 gennaio 2025

Indice

1	Introduzione	3
2	Analisi dell'espressione differenziale	3
2.1	Introduzione	3
2.1.1	Che cosa è un gene?	3
2.1.2	Che cosa è l'espressione genica?	3
2.2	Concetti di base dell'analisi dell'espressione differenziale	3
2.2.1	Caratteristiche e distribuzione dei dati di counts	4
2.2.2	Importanza biologica della DEA	5
3	DESeq2	6
3.1	Teoria pacchetto	6
3.1.1	Stima size factor - metodo median of ratios	6
3.1.2	Stima dispersione per ogni gene	7
3.1.3	Adattamento del modello	8
3.1.4	Stima log fold change	8
3.1.5	Test di ipotesi per l'espressione differenziale	10
4	Esempio su dataset	11
4.1	Distribuzione dei counts	11
4.2	Creazione oggetto DESeqDataSet	13
4.2.1	Analisi di controllo di qualità dei dati	13
4.3	Analisi dell'espressione differenziale	15
4.3.1	Test di ipotesi	17
4.4	Visualizzazione ed esplorazione dei risultati	17
5	Acronimi	21
Riferimenti		22

Elenco delle figure

1	Processo di RNA sequencing. Esempio giocattolo in cui il genoma è costituito da soli cinque geni specificati dai colori rosso, blu, viola, verde e arancione. Si parte isolando l'RNA totale dalle cellule, che rappresenta le molecole attive in un determinato momento. Successivamente, l'RNA viene frammentato e convertito in cDNA attraverso la retrotrascrizione. Per prepararlo al sequenziamento, vengono aggiunti adattatori specifici alle estremità del cDNA. Il materiale così preparato viene quindi sottoposto ad una piattaforma di sequenziamento ad alta capacità, che legge le sequenze nucleotidiche dei frammenti. Una volta ottenute queste sequenze (chiamate reads), vengono allineate al genoma di riferimento per identificarne l'origine. Infine, i reads mappati su ciascun gene vengono conteggiati per quantificare il livello di espressione di ogni gene.	4
2	Workflow di DESeq2. I box della colonna a sinistra corrispondono alle lettere, mentre quelli di destra ai numeri romani. (A) modellazione counts grezzi per ogni gene (i) normalizzazione dei counts stimando i size factors; (ii) stima della dispersione dei geni; (iii) adattamento curva di dispersione; (iv) riduzione stime di dispersione; (v) adattamento GLM per ogni gene; (B) riduzione stime log2 fold changes; (C) test per l'espressione differenziale.	6
3	Plot della media vs varianza dei counts.	7
4	Stime di dispersione rispetto alla forza media di espressione.	8
5	Effetto del restringimento sulle stime logaritmiche di fold change. Grafici della stima MLE (A) (cioè senza restringimento) e MAP (B) (cioè con restringimento) per le LFC, rispetto alla forza media di espressione. I piccoli triangoli nella parte superiore e inferiore dei plot indicano i punti che cadrebbero al di fuori della finestra di plot. Due geni con conteggi medi e variazioni logaritmiche MLE simili sono evidenziati con cerchi verdi e viola. (C) I counts, normalizzati dai size factors s_j , per questi geni rivelano una bassa dispersione per il gene in verde e un'alta dispersione per il gene in viola. (D) Grafici di densità delle verosimiglianze (linee intere) e dei posteriori (linee tratteggiate) per i geni verde e viola e del priore (linea nera intera): a causa della maggiore dispersione del gene viola, la sua verosimiglianza è più ampia e con un picco minore (indicando meno informazioni) e il priore ha maggiore influenza sul suo posteriore rispetto al gene verde. La maggiore curvatura del posteriore verde al suo massimo si traduce in un minore errore standard riportato per la stima MAP LFC (barra di errore orizzontale).	9
6	Distribuzione dei counts di espressione grezzi.	12
7	Relazione tra media e varianza dei counts.	13
8	Heatmap delle correlazioni tra i campioni.	15
9	Stima della dispersione per ciascun gene rispetto alla media di espressione.	16
10	Media dei counts normalizzati rispetto ai log2 fold change. I geni differenzialmente espressi sono colorati in blu.	18
11	Effetto del restringimento dei LFC.	18
12	Counts per il gene con il p-value più piccolo.	21

1 Introduzione

In questa tesina viene descritto il pacchetto DESeq2 (vignetta) [5], uno strumento specifico per l'analisi dell'espressione differenziale genica basata sulla distribuzione binomiale negativa. Il pacchetto fa parte di Bioconductor, un progetto che mira a sviluppare e condividere software open source per l'analisi dati biologici.

La tesina è strutturata in tre parti principali: nella prima viene introdotta la teoria alla base dell'analisi dell'espressione differenziale genica; nella seconda si approfondiscono le caratteristiche del pacchetto DESeq2 e i principi teorici su cui si basa; nella terza parte, infine, viene presentato un esempio pratico dell'utilizzo di DESeq2, applicato ad un dataset di RNA-seq dello studio al [3, 4].

2 Analisi dell'espressione differenziale

2.1 Introduzione

2.1.1 Che cosa è un gene?

Un gene è l'unità fondamentale di informazione genetica contenuta nel DNA, che codifica per una proteina o un RNA funzionale (ad esempio RNA ribosomiale o microRNA).

Nei geni codificanti, l'informazione genetica viene trascritta in RNA messaggero (mRNA) e successivamente tradotta in proteine, che svolgono funzioni essenziali per il metabolismo, la struttura e la comunicazione cellulare. Nei geni non codificanti, l'RNA prodotto ha ruoli regolatori o strutturali.

I geni non sono espressi in modo uniforme: la loro attività, denominata espressione, varia tra tipi di cellule, condizioni ambientali, stati di sviluppo e risposte a stimoli, contribuendo alla diversità biologica e alla funzione specifica dei tessuti.

2.1.2 Che cosa è l'espressione genica?

L'**espressione genica** si riferisce al processo attraverso il quale un gene viene attivato per generare il suo prodotto funzionale. Questo processo può essere suddiviso in due fasi principali:

1. Trascrizione: DNA viene copiato in mRNA.
2. Traduzione: mRNA viene utilizzato come stampo per la sintesi proteica.

L'espressione genica può essere misurata come la quantità di mRNA prodotto (livello trascrizione) o la quantità della proteina risultante (livello traduzione).

Capire come i geni vengono espressi in condizioni diverse aiuta a rispondere a domande biologiche fondamentali, come, ad esempio, quali geni sono coinvolti in una malattia e quali effetti ha un farmaco su una popolazione cellulare.

L'analisi dell'espressione differenziale, quindi, cerca di identificare quali geni hanno un'espressione significativamente diversa tra gruppi differenti (es. cellule sane vs malate, prima e dopo un trattamento, ecc.).

2.2 Concetti di base dell'analisi dell'espressione differenziale

L'analisi dell'espressione differenziale è una tecnica bioinformatica utilizzata per confrontare l'attività di migliaia di geni tra due o più condizioni sperimentali. Si basa sui dati di RNA-seq, che misurano quantitativamente l'espressione genica. Questo significa che, per ciascun gene viene determinato un valore numerico, chiamato count, il quale rappresenta la quantità di mRNA prodotto, fornendo così un'indicazione precisa del livello di attività di quel gene in un dato campione. Un esempio del processo di RNA-seq è mostrato in figura 1.

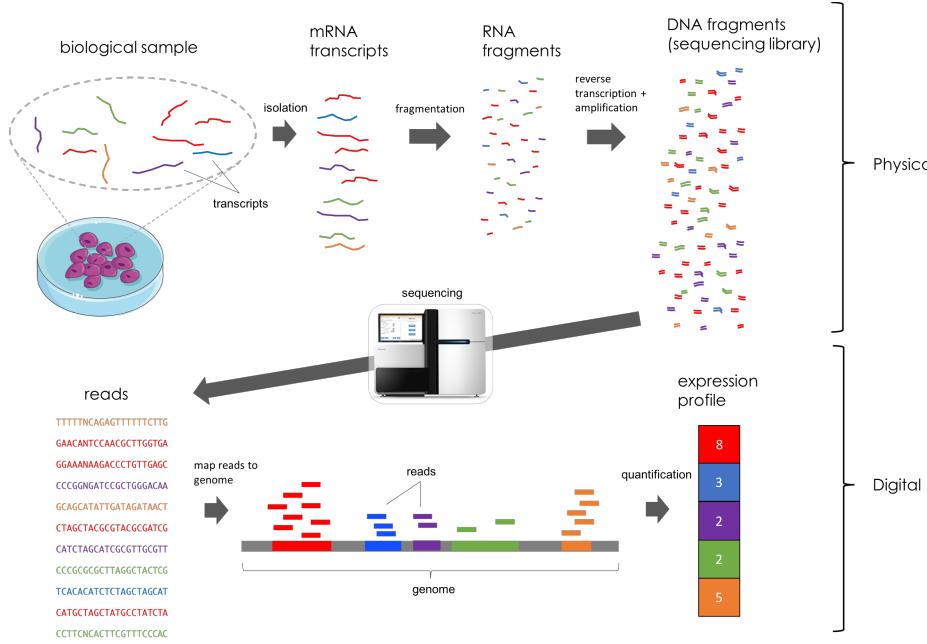


Figura 1: Processo di RNA sequencing. Esempio giocattolo in cui il genoma è costituito da soli cinque geni specificati dai colori rosso, blu, viola, verde e arancione. Si parte isolando l'RNA totale dalle cellule, che rappresenta le molecole attive in un determinato momento. Successivamente, l'RNA viene frammentato e convertito in cDNA attraverso la retrotrascrizione. Per prepararlo al sequenziamento, vengono aggiunti adattatori specifici alle estremità del cDNA. Il materiale così preparato viene quindi sottoposto ad una piattaforma di sequenziamento ad alta capacità, che legge le sequenze nucleotidiche dei frammenti. Una volta ottenute queste sequenze (chiamate reads), vengono allineate al genoma di riferimento per identificarne l'origine. Infine, i reads mappati su ciascun gene vengono conteggiati per quantificare il livello di espressione di ogni gene.

L'obiettivo principale è **l'identificazione dei geni con espressione significativamente diversa tra condizioni**, quantificando l'entità del cambiamento e valutandone la rilevanza statistica per distinguere variazioni reali dal caso.

2.2.1 Caratteristiche e distribuzione dei dati di counts

In un esperimento di sequenziamento dell'RNA, dopo che le reads sono state sequenziate e quantificate, il risultato che si ottiene è una matrice K in cui le righe sono i geni e le colonne sono i campioni. Questi counts rappresentano il **numero di reads mappate per quel gene in quel campione**.

Semplici controlli sulla distribuzione, come quello mostrato in figura 6, evidenziano che i dati dei counts non seguono una distribuzione normale. Una possibile alternativa potrebbe essere la distribuzione di Poisson, che descrive il numero di eventi che si verificano in un dato intervallo di tempo o spazio. Tuttavia, questa distribuzione presume che la media e la varianza siano uguali (parametro λ), una condizione che non si verifica nei dati di RNA-seq, dove la varianza è spesso maggiore della media. Di conseguenza, la distribuzione di Poisson non è adeguata per descrivere tali dati.

Si può immaginare un processo teorico in cui il genoma viene esplorato casualmente per generare reads, un esempio che rientra nella definizione di distribuzione di Poisson. Tuttavia, nei dati reali di RNA-seq, esistono variazioni biologiche aggiuntive che non possono essere catturate da un modello di Poisson, rendendo necessario ricorrere a modelli più complessi, come la distribuzione binomiale negativa [2].

La distribuzione binomiale negativa è progettata per situazioni in cui la varianza supera la media ed è definita da due parametri: la media μ , che rappresenta il livello di espressione medio del gene e la dispersione α , che indica quanto i conteggi si discostano dalla media e consente di modellare la variabilità extra che viene generata nei dati di RNA-seq [1, 6]. Questo concetto è illustrato nella figura 3.

2.2.2 Importanza biologica della DEA

L'analisi dell'espressione differenziale fornisce un quadro chiaro dei processi molecolari che distinguono una condizione da un'altra. Questo è fondamentale in numerosi contesti:

- ricerca medica: identificare geni responsabili di malattie o geni alterati in condizioni patologiche;
- ricerca farmaceutica: determinare quali geni rispondono a un trattamento;
- biologia dello sviluppo: studiare le variazioni geniche durante il differenziamento cellulare;
- ecologia e biologia evolutiva: analizzare l'espressione genica in risposta a stress ambientali.

3 DESeq2

3.1 Teoria pacchetto

DESeq2 esegue gli step riportati nella figura 2 per identificare se un insieme di geni è differenzialmente espresso o no. Tutti i passaggi in blu sono eseguiti automaticamente dalla funzione `DESeq()` e sono descritti nelle sezioni successive.

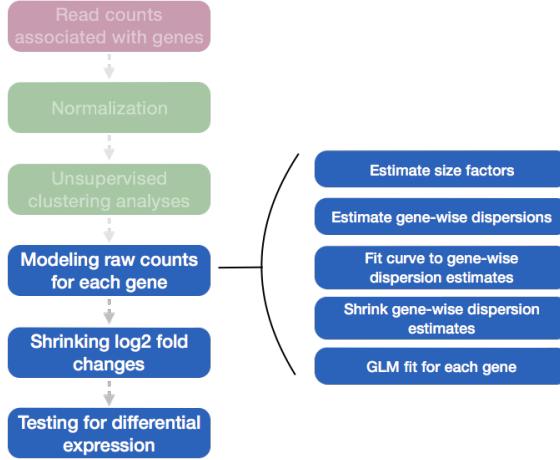


Figura 2: Workflow di DESeq2. I box della colonna a sinistra corrispondono alle lettere, mentre quelli di destra ai numeri romani. (A) modellazione counts grezzi per ogni gene (i) normalizzazione dei counts stimando i size factors; (ii) stima della dispersione dei geni; (iii) adattamento curva di dispersione; (iv) riduzione stime di dispersione; (v) adattamento GLM per ogni gene; (B) riduzione stime log2 fold changes; (C) test per l'espressione differenziale.

Il punto di partenza di un'analisi DESeq2 è una matrice K dei counts di dimensione $n \times m$, dove $i = 1, \dots, n$ indica i geni e $j = 1, \dots, m$ indica i campioni. L'entry della matrice K_{ij} indica il numero di read counts che sono stati mappati sul gene i nel campione j .

3.1.1 Stima size factor - metodo median of ratios

Bisogna innanzitutto normalizzare i dati dei counts, per permettere un corretto confronto tra i geni. Per fare ciò è necessario calcolare i size factors. Il size factor s_{ij} è un fattore di scala che tiene conto della profondità di sequenziamento e della composizione del campione; all'interno del campione i sono considerati costanti: $s_{ij} = s_j$.

Il size factor viene calcolato impiegando tre passaggi principali:

1. si calcola la media geometrica per ciascun gene (geometrica perché è più robusta agli outliers rispetto a quella aritmetica): $(\prod_{v=1}^m K_{iv})^{1/m}$;
2. si dividono i counts per la media geometrica: $\frac{K_{ij}}{(\prod_{v=1}^m K_{iv})^{1/m}}$;
3. si calcola la mediana dei rapporti (calcolati nello step 2): $\text{median}_i \frac{K_{ij}}{(\prod_{v=1}^m K_{iv})^{1/m}}$.

Formalmente: $s_j = \text{median}_i \frac{K_{ij}}{(\prod_{v=1}^m K_{iv})^{1/m}}$

Infine, per normalizzare i dati, si divide ogni counts per il size factor corrispondente.

3.1.2 Stima dispersione per ogni gene

La variabilità all'interno dei replicati è modellata dal parametro di dispersione α_i , che descrive la varianza dei counts secondo la formula $\text{Var } K_{ij} = \mu_{ij} + \alpha_i \mu_{ij}^2$. Questo parametro α rappresenta quanto ci si può aspettare che un count osservato si discosti dal valore medio atteso (μ_{ij}). Una stima accurata della dispersione è fondamentale per l'inferenza statistica nell'analisi dell'espressione differenziale, poiché campioni di piccole dimensioni possono portare a stime altamente variabili per ciascun gene.

Il grafico della media rispetto alla varianza nei dati di counts in figura 3 mostra che la varianza dell'espressione genica aumenta con l'espressione media (ogni punto nero è un gene). Si noti che la relazione tra media e varianza è lineare sulla scala logaritmica e che per medie più elevate è possibile prevedere la varianza in modo relativamente accurato in base alla media. Tuttavia, per medie basse, le stime della varianza hanno una dispersione molto più ampia; pertanto, le stime della dispersione differiranno molto di più tra geni con medie piccole.

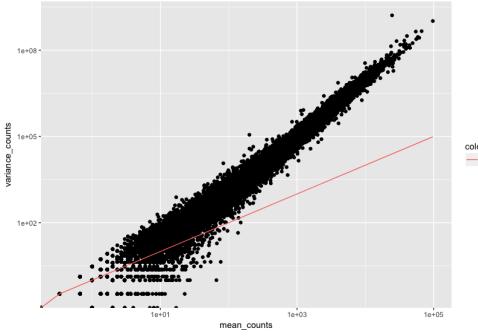


Figura 3: Plot della media vs varianza dei counts.

Per affrontare questo problema, vengono condivise informazioni tra i geni, presupponendo che quelli con livelli di espressione simili abbiano anche dispersioni simili.

Il processo di stima della dispersione inizia con una stima preliminare per ciascun gene, ottenuta utilizzando il metodo della massima verosimiglianza (MLE), basato esclusivamente sui dati relativi al gene stesso. Queste stime iniziali, però, possono essere imprecise o rumorose, specialmente per geni con counts bassi o con elevata dispersione.

Per ridurre il rumore e migliorare la precisione di queste stime, viene adattata una curva smooth (rappresentata dalla curva rossa in figura 4) che descrive il valore atteso della dispersione in funzione dell'espressione media. Questa curva fornisce un riferimento generale e riflette la dipendenza della dispersione dalla forza di espressione.

Successivamente, viene definita una distribuzione a priori per la dispersione, basata sull'adattamento della curva smooth e sulle proprietà osservate nei dati. La distribuzione a priori rappresenta le conoscenze iniziali o ipotesi sulla dispersione, fornendo un'idea del valore "plausibile" per ciascun gene in base alla media di espressione. Inoltre, controlla l'entità del restringimento (shrinkage), ovvero quanto le stime preliminari devono essere ristrette verso il valore previsto dalla curva.

Per concludere, le stime finali di dispersione (indicate dalle punte delle frecce blu) vengono aggiornate utilizzando il principio della massima a posteriori (MAP), che combina le informazioni della distribuzione a priori con i dati osservati. Questo restringimento spinge molte delle stime dei geni verso i valori previsti dalla curva adattata, riducendo l'impatto del rumore e minimizzando il rischio di falsi positivi causati da dispersioni sottostimate.

Non tutti i geni sono però soggetti a questo processo di regolarizzazione. Alcuni geni, evidenziati nella figura

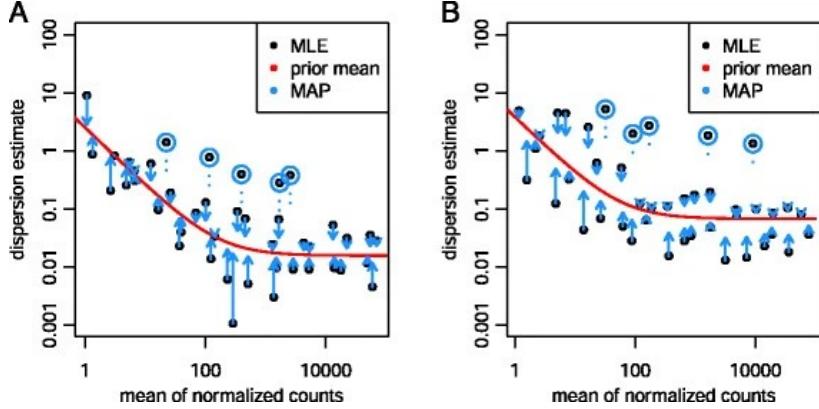


Figura 4: Stime di dispersione rispetto alla forza media di espressione.

con un cerchio blu, mostrano una variabilità di dispersione così elevata che DESeq2 presume che non seguano le ipotesi del modello (il restringimento avrebbe seguito la linea tratteggiata). Per questi geni, è presente una variabilità aggiuntiva che non può essere spiegata dalla sola variazione biologica o tecnica.

Questo approccio bilancia la variabilità specifica di ciascun gene con la tendenza generale osservata nei dati. La forza del restringimento dipende da due fattori principali: (i) quanto i valori reali di dispersione tendano ad aderire alla curva stimata e (ii) i gradi di libertà. All'aumentare della dimensione del campione, il restringimento si riduce progressivamente fino a diventare trascurabile.

3.1.3 Adattamento del modello

Le read counts K_{ij} sono descritte con un modello lineare generalizzato (GLM), la quale famiglia è un'estensione dei modelli di regressione lineare. Questi modelli sono caratterizzati da una specifica distribuzione (non gaussiana) delle risposte e da una funzione di link, che trasferisce il valore medio in una scala in cui la relazione con le variabili esplicative è lineare e additiva, per consentire una forma più generale di espressione della risposta media. In generale: $f\{E(Y_i)\} = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}$, dove $f(\cdot)$ rappresenta la funzione di link e $\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}$ il predittore lineare.

Il GLM utilizzato fa parte della famiglia binomiale negativa, con media μ_{ij} e dispersione α_i . La media $\mu_{ij} = s_j \cdot q_{ij}$ è proporzionale alla concentrazione di frammenti cDNA del gene i nel campione j : q_{ij} , scalata da un fattore di normalizzazione s_j . Questo fattore di normalizzazione (sample specific size factor) equivale a quello descritto nella sezione stima size factor.

Per ogni gene viene quindi adattato un GLM con un link logaritmico $\log_2(q_{ij}) = \sum_r x_{jr} \cdot \beta_{ir}$, con x_{jr} elementi della matrice di design e β_{ir} coefficienti che rappresentano il log fold change (LFC). La matrice di design è una matrice $n \times (p + 1)$ in cui ogni riga rappresenta una singola osservazione (dimensione n) e ogni colonna rappresenta una variabile esplicativa (dimensione p). La prima colonna aggiuntiva è il termine di intercetta.

3.1.4 Stima log fold change

Una delle difficoltà principali nell'analisi dell'espressione differenziale è la forte varianza (eteroschedasticità) delle stime del log fold change per geni con bassi read counts.

Il log fold change è una misura che descrive quanto cambia una quantità tra una misurazione originale e una successiva [7]; in questo contesto **quantifica la differenza nell'espressione di un gene tra due condizioni**. In particolare, il fold change rappresenta il rapporto tra i livelli di espressione nelle due condizioni, mentre il logaritmo in base 2 di questo rapporto rende più gestibili i valori e simmetriche le

variazioni. Si prenda ad esempio un LFC di +1: esso indica un raddoppio dell'espressione in una condizione rispetto all'altra, mentre un LFC di -1 indica una riduzione a metà.

DESeq2 supera il problema della forte varianza nelle stime del LFC restringendo queste ultime verso lo zero, in modo che il restringimento sia più marcato quando le informazioni disponibili per un gene sono limitate, ad esempio a causa di bassi conteggi, alta dispersione o pochi gradi di libertà.

Si inizia con l'adattamento di GLM per ottenere le stime di massima verosimiglianza dei LFC. Successivamente, una distribuzione normale centrata su zero viene adattata alla distribuzione delle MLE osservate su tutti i geni. Questa distribuzione normale viene utilizzata come distribuzione a priori, rappresentando le informazioni iniziali sui LFC prima di considerare i dati osservati ed è centrata su zero perché, a priori, si presume che la maggior parte dei geni non abbia variazioni significative nell'espressione genica, ovvero che il loro LFC sia vicino a zero.

Le stime finali dei LFC vengono calcolate applicando il principio della massima a posteriori, che combina la distribuzione a priori con le prove fornite dai dati osservati. Tale approccio consente di 'pesare' i dati osservati rispetto alla distribuzione a priori: se un gene mostra una variazione significativa, la stima finale si allontanerà da zero; altrimenti, sarà spinta verso il valore previsto a priori. Questa regolarizzazione è particolarmente utile per attenuare il rumore nei dati o quando il numero di campioni è limitato, rendendo le stime più robuste.

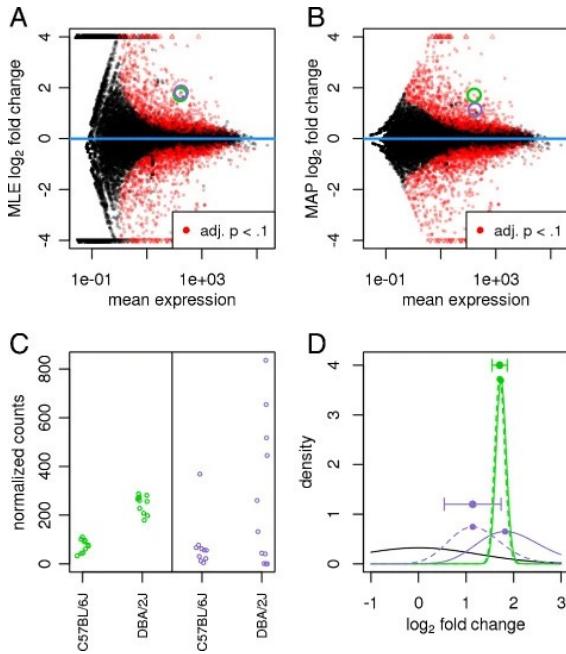


Figura 5: Effetto del restringimento sulle stime logaritmiche di fold change. Grafici della stima MLE (A) (cioè senza restringimento) e MAP (B) (cioè con restringimento) per le LFC, rispetto alla forza media di espressione. I piccoli triangoli nella parte superiore e inferiore dei plot indicano i punti che cadrebbero al di fuori della finestra di plot. Due geni con conteggi medi e variazioni logaritmiche MLE simili sono evidenziati con cerchi verdi e viola. (C) I counts, normalizzati dai size factors s_j , per questi geni rivelano una bassa dispersione per il gene in verde e un'alta dispersione per il gene in viola. (D) Grafici di densità delle verosimiglianze (linee intere) e dei posteriori (linee tratteggiate) per i geni verde e viola e del priore (linea nera intera): a causa della maggiore dispersione del gene viola, la sua verosimiglianza è più ampia e con un picco minore (indicando meno informazioni) e il priore ha maggiore influenza sul suo posteriore rispetto al gene verde. La maggiore curvatura del posteriore verde al suo massimo si traduce in un minore errore standard riportato per la stima MAP LFC (barra di errore orizzontale).

Questi LFC ridotti e i loro errori standard associati vengono utilizzati nei test di Wald per l'espressione differenziale, descritti nella sezione successiva.

La forza del restringimento non dipende semplicemente dalla media dei conteggi, ma dalla quantità complessiva di informazioni disponibili per stimare il fold change. Prendendo ad esempio, due geni con conteggi medi simili ma dispersioni diverse, questi subiranno un restringimento diverso. Tale concetto è illustrato nella figura 5.

3.1.5 Test di ipotesi per l'espressione differenziale

Dopo aver adattato i modelli per ciascun gene, si procede a verificare se il coefficiente β_{ir} differisce significativamente da zero, con i che rappresenta il gene e r il trattamento o la condizione sperimentale. L'ipotesi nulla è che il trattamento non abbia alcun effetto sull'espressione del gene: $H_0 : \beta_{ir} = 0$. L'ipotesi alternativa, invece, è che il gene sia differenzialmente espresso in risposta al trattamento: $H_1 : \beta_{ir} \neq 0$.

Per testare l'ipotesi nulla viene utilizzato il test di Wald, in cui la stima del log2 fold change β_{ir} viene divisa per il suo errore standard, producendo una statistica-z. Formalmente: $\beta_{ir}/\text{SE}(\beta_{ir})$.

Questa statistica viene confrontata con una distribuzione normale standard ($N(0, 1)$) per calcolare il p-value, che rappresenta la probabilità di ottenere un risultato uguale o più estremo di quello osservato, partendo dal presupposto che H_0 sia vera.

- Se H_0 è vera, il valore atteso della statistica-z sarà molto vicino a zero, e questa seguirà approssimativamente una distribuzione normale standard.
- Valori di z molto elevati in valore assoluto suggeriscono che l'ipotesi nulla potrebbe essere falsa. In questi casi, il p-value associato risulta piccolo e inferiore alla soglia di significatività (ad esempio 0,05). Questo porta al rifiuto di H_0 , indicando che il gene presenta una differenza significativa nell'espressione tra le condizioni confrontate.

4 Esempio su dataset

Vengono caricati i dati e i metadati necessari, specificando che si ha un header e che la prima colonna rappresenta i nomi delle righe.

```
data <- read.table("Mov10_full_counts.txt", header=T, row.names=1)
head(data)
```

```
##          Mov10_kd_2 Mov10_kd_3 Mov10_oe_1 Mov10_oe_2 Mov10_oe_3 Irrel_kd_1
## 1/2-SBSRNA4      57       41      64      55      38      45
## A1BG            71       40     100      81      41      77
## A1BG-AS1        256      177     220     189     107     213
## A1CF             0        1       1       0       0       0
## A2LD1           146      81     138     125      52      91
## A2M              10       9       2       5       2       9
##          Irrel_kd_2 Irrel_kd_3
## 1/2-SBSRNA4      31       39
## A1BG            58       40
## A1BG-AS1        172      126
## A1CF             0        0
## A2LD1           80       50
## A2M              8        4
```

```
meta <- read.table("Mov10_full_meta.txt", header=T, row.names=1)
head(meta)
```

```
##                 samplename MOVexpr
## Mov10_kd_2      MOV10 Knockdown    low
## Mov10_kd_3      MOV10 Knockdown    low
## Mov10_oe_1      MOV10 overexpression high
## Mov10_oe_2      MOV10 overexpression high
## Mov10_oe_3      MOV10 overexpression high
## Irrel_kd_1      control normal
```

Si vuole cercare geni che cambiano di espressione tra due o più gruppi, definiti nei metadati:

- **Mov10_oe** (over expression): gene sovraespresso artificialmente, per studiare gli effetti biologici della sovraregolazione del gene.
- **Mov10_kd** (knock down): condizione sperimentale in cui l'espressione del gene è deliberatamente ridotta per studiare i suoi effetti biologici e molecolari.
- **Irrelevant_kd**: condizione di controllo in cui le cellule sono state trattate in modo da non influenzare l'espressione di Mov10 o di altri geni.

```
unique(meta$samplename)
```

```
## [1] "MOV10_Knockdown"      "MOV10_overexpression" "control"
```

4.1 Distribuzione dei counts

Per visualizzare la distribuzione dei counts, si stampa un'istogramma per il campione Mov10_kd_2 e una versione zoomata.

```
plot1 <- ggplot(data) +
  geom_histogram(aes(x = Mov10_kd_2), stat = "bin", bins = 200) +
```

```

xlab("Counts di espressione grezzi") +
ylab("Numero di geni")

plot2 <- ggplot(data) +
  geom_histogram(aes(x = Mov10_kd_2), stat = "bin", bins = 200) +
  xlim(-5, 250) + # limite asse x
  xlab("Counts di espressione grezzi") +
  ylab("Numero di geni")

# combino i plot
grid.arrange(plot1, plot2, ncol = 2)

```

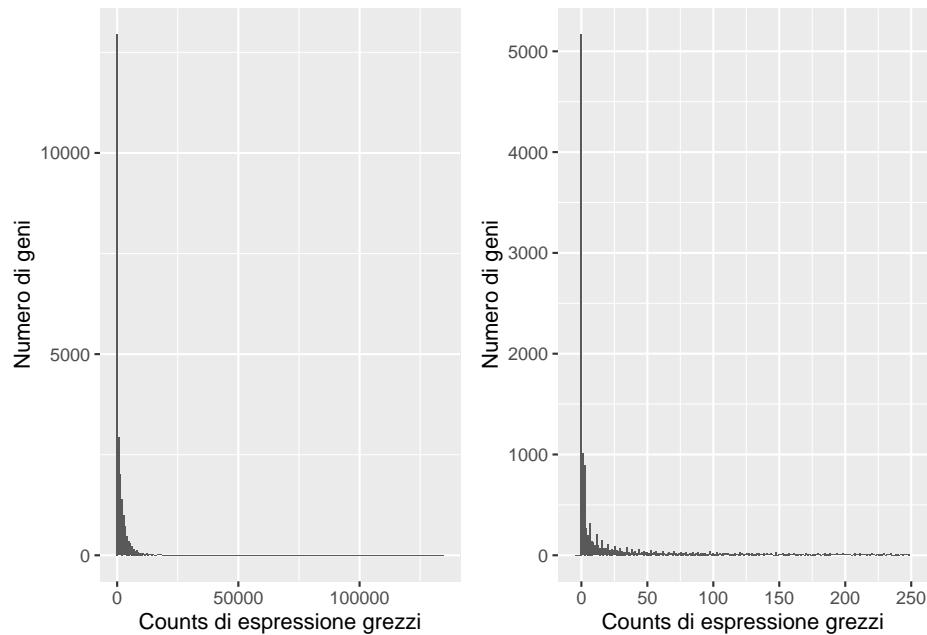


Figura 6: Distribuzione dei counts di espressione grezzi.

In figura 6 si può vedere come la distribuzione dei counts sia asimmetrica e non normale (e quindi una binomiale negativa), con molti geni con bassi conteggi.

Per controllare la relazione tra media e varianza, si calcolano media e varianza per i replicati Mov10_kd.

```

mean_counts <- apply(data[, 1:2], 1, mean)
variance_counts <- apply(data[, 1:2], 1, var)
df <- data.frame(mean_counts, variance_counts)

ggplot(df) +
  geom_point(aes(x=mean_counts, y=variance_counts)) +
  geom_line(aes(x=mean_counts, y=mean_counts, color="red")) +
  scale_y_log10() +
  scale_x_log10()

```

In figura 7 si nota che la varianza tra i replicati tende a essere maggiore della media (linea rossa), soprattutto per i geni con livelli di espressione medi elevati.

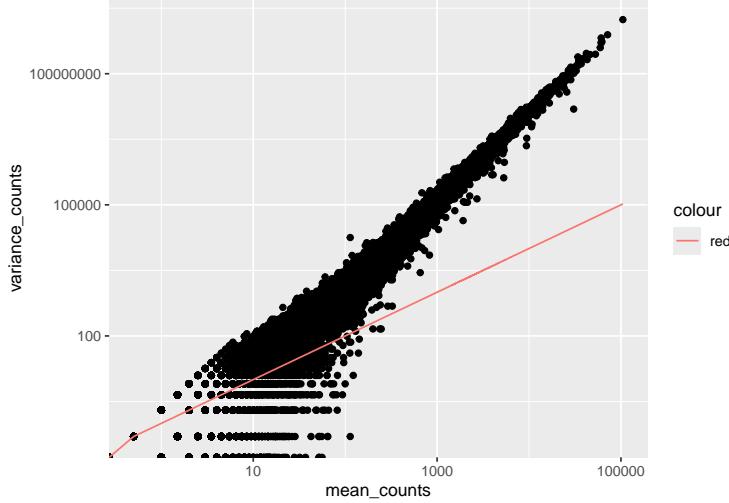


Figura 7: Relazione tra media e varianza dei counts.

4.2 Creazione oggetto DESeqDataSet

Prima di procedere con l'analisi, è buona norma controllare che i nomi dei campioni corrispondano tra i dati e i metadati.

```
all(colnames(data) %in% rownames(meta))

## [1] TRUE

all(colnames(data) == rownames(meta))

## [1] TRUE
```

Per iniziare l'analisi è necessario creare un oggetto `DESeqDataSet`, utilizzando la matrice dei counts e la tabella dei metadati. Bisogna anche specificare una formula di design, che definisce quali colonne della tabella dei metadati devono essere considerate e come devono essere utilizzate nell'analisi. Nel dataset, la colonna di interesse è `samplotype`, che contiene tre livelli di un fattore. Questi livelli indicano a `DESeq2` di valutare, per ciascun gene, le variazioni nell'espressione genica tra le diverse condizioni sperimentali rappresentate.

```
dds <- DESeqDataSetFromMatrix(countData = data, colData = meta, design = ~ samplotype)
```

Per visualizzare i dati, è necessario prima normalizzarli. Questo passaggio è solo per scrupolo, la funzione `DESeq()` esegue automaticamente tutti i passaggi necessari.

```
dds_norm <- estimateSizeFactors(dds)
```

Per recuperare la matrice dei counts normalizzati, si usa la funzione `counts()` e si aggiunge l'argomento `normalized=TRUE`.

```
normalized_counts <- counts(dds_norm, normalized=TRUE)
```

4.2.1 Analisi di controllo di qualità dei dati

Per migliorare le distanze/clustering per i metodi di visualizzazione PCA e di clustering gerarchico, è necessario moderare la varianza rispetto alla media applicando la trasformazione *rlog* ai counts normalizzati. Questa funzione trasforma i dati di counts in scala logaritmica in base 2 in modo da minimizzare le differenze

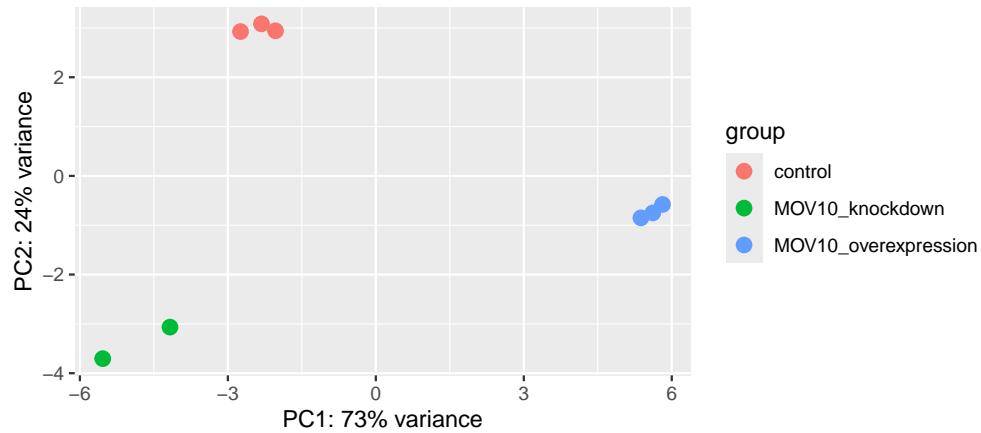
tra i campioni per le righe con counts piccoli. L'argomento `blind=TRUE` produce una trasformazione non influenzata dalle informazioni sulle condizioni del campione.

```
rld <- rlog(dds_norm, blind=TRUE)
```

4.2.1.1 PCA La libreria offre una funzione built-in per plot PCA, richiede in input l'oggetto rlog e la colonna dei metadati di interesse.

```
plotPCA(rld, intgroup="samplotype")
```

```
## using ntop=500 top features by variance
```



Questa analisi delle componenti principali è accettabile, in quanto la prima componente spiega già il 73% della variazione nei dati.

4.2.1.2 Clustering gerarchico Per visualizzare le correlazioni tra i campioni, bisogna calcolare la matrice di correlazione dei counts rlog. Prima si estrae la matrice rlog dall'oggetto e poi si calcolano i valori di correlazione a coppie per i campioni.

```
rld_mat <- assay(rld) # assay() funzione caricata dalle dipendenze di DESeq2
rld_cor <- cor(rld_mat)
```

```
head(rld_cor)
```

```
##           Mov10_kd_2 Mov10_kd_3 Mov10_oe_1 Mov10_oe_2 Mov10_oe_3 Irrel_kd_1
## Mov10_kd_2 1.0000000 0.9999492 0.9994868 0.9994565 0.9993869 0.9997202
## Mov10_kd_3 0.9999492 1.0000000 0.9996154 0.9995905 0.9995235 0.9997748
## Mov10_oe_1 0.9994868 0.9996154 1.0000000 0.9999505 0.9999196 0.9996700
## Mov10_oe_2 0.9994565 0.9995905 0.9999505 1.0000000 0.9998711 0.9996599
## Mov10_oe_3 0.9993869 0.9995235 0.9999196 0.9998711 1.0000000 0.9995804
## Irrel_kd_1 0.9997202 0.9997748 0.9996700 0.9996599 0.9995804 1.0000000
##           Irrel_kd_2 Irrel_kd_3
```

```

## Mov10_kd_2 0.9996918 0.9996816
## Mov10_kd_3 0.9997568 0.9997574
## Mov10_oe_1 0.9996984 0.9997067
## Mov10_oe_2 0.9996825 0.9997090
## Mov10_oe_3 0.9996227 0.9996026
## Irrel_kd_1 0.9999614 0.9999532

```

Valori di correlazione plottati come una heatmap.

```

heat.colors <- brewer.pal(6, "YlGn")
pheatmap(rld_cor, color = heat.colors)

```

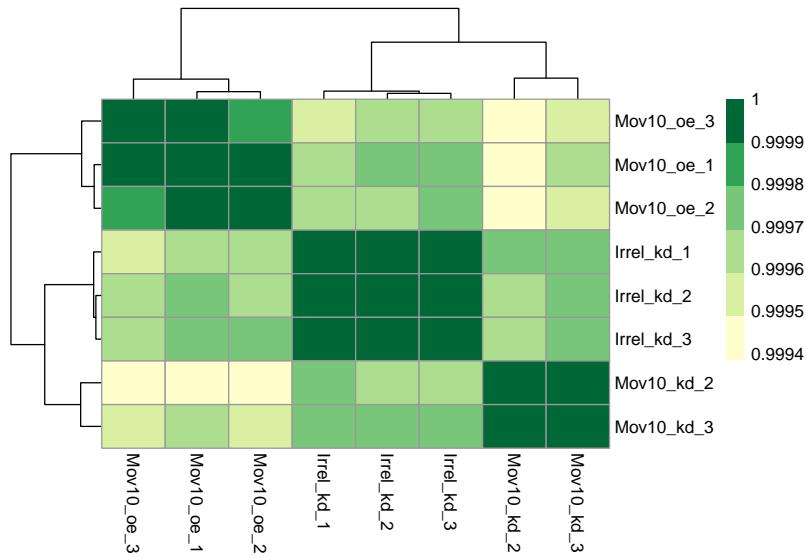


Figura 8: Heatmap delle correlazioni tra i campioni.

Si osservano correlazioni elevate che suggeriscono l'assenza di campioni anomali. Analogamente al plot PCA, i campioni si raggruppano per gruppi di campioni. L'insieme di questi grafici suggerisce che i dati sono di buona qualità.

4.3 Analisi dell'espressione differenziale

Viene utilizzato l'oggetto `DESeqDataSet` creato in precedenza. Come già ripetuto, la funzione `DESeq()` esegue tutti i passaggi ma il pacchetto offre delle singole funzioni che permettono di eseguire ogni fase del workflow in modo graduale.

```
dds_an <- DESeq(dds)
```

```

## estimating size factors
## estimating dispersions
## gene-wise dispersion estimates
## mean-dispersion relationship
## final dispersion estimates
## fitting model and testing

```

Si controllano i size factors stimati per ogni campione.

```
sizeFactors(dds_an)

## Mov10_kd_2 Mov10_kd_3 Mov10_oe_1 Mov10_oe_2 Mov10_oe_3 Irrel_kd_1 Irrel_kd_2
## 1.5646728 0.9351760 1.2016082 1.1205912 0.6534987 1.1224020 0.9625632
## Irrel_kd_3
## 0.7477715
```

Numero totale di counts grezzi per ogni campione:

```
colSums(counts(dds_an))
```

```
## Mov10_kd_2 Mov10_kd_3 Mov10_oe_1 Mov10_oe_2 Mov10_oe_3 Irrel_kd_1 Irrel_kd_2
## 32826936 19360003 23447317 21713289 12737889 22687366 19381680
## Irrel_kd_3
## 14962754
```

e numero totale di counts normalizzati per ogni campione:

```
colSums(counts(dds_an, normalized=T))
```

```
## Mov10_kd_2 Mov10_kd_3 Mov10_oe_1 Mov10_oe_2 Mov10_oe_3 Irrel_kd_1 Irrel_kd_2
## 20980064 20701989 19513279 19376636 19491836 20213226 20135487
## Irrel_kd_3
## 20009794
```

Poiché il campione è di dimensioni ridotte, per molti geni si osserva, in figura 9 , un restringimento ma nel complesso i dati sono adatti al modello di DESeq2.

```
plotDispEts(dds_an)
```

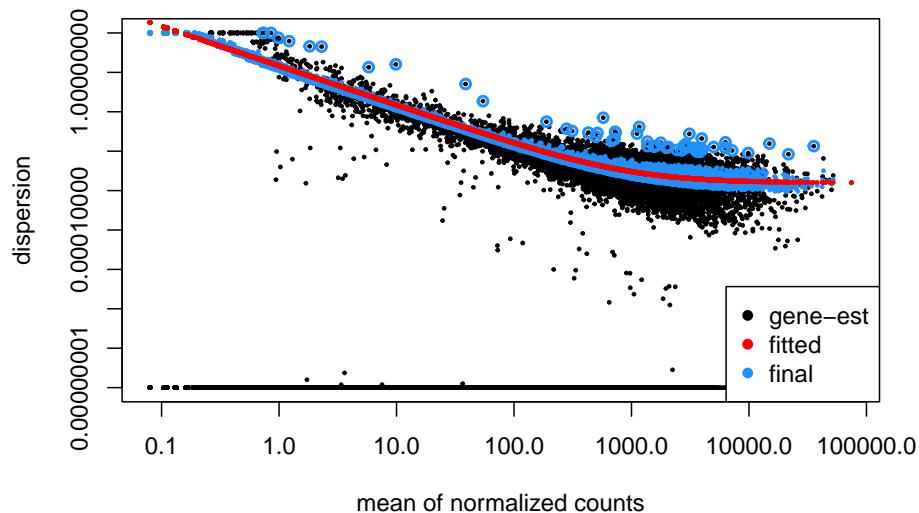


Figura 9: Stima della dispersione per ciascun gene rispetto alla media di espressione.

4.3.1 Test di ipotesi

Come descritto nella spiegazione del test d'ipotesi, i coefficienti β_{ir} ristretti rappresentano i LFC per ogni gruppo di campioni. Bisogna quindi testare se questi coefficienti sono significativamente diversi da zero, ovvero che non vi sia espressione differenziale tra i gruppi di campioni.

Per indicare a DESeq2 i due gruppi che si vogliono confrontare bisogna usare i contrasti, che vengono utilizzati per eseguire i test di espressione differenziale utilizzando il test di Wald.

I contrasti possono essere forniti a DESeq2 in due modi diversi:

- in modo automatico viene utilizzato il livello del fattore di base della condizione di interesse come base per i test statistici. Il livello di base viene scelto in base all'ordine alfabetico dei livelli.
- specificare il confronto di interesse e i livelli da confrontare nella funzione `results()`. Il livello indicato per ultimo è il livello di base per il confronto.

I possibili confronti a coppie sono i tre seguenti, con i primi due che sono più di interesse:

1. `controllo` vs `Mov10_overexpression`
2. `controllo` vs `Mov10_knockdown`
3. `Mov10_knockdown` vs `Mov10_overexpression`

Con i seguenti comandi vengono definiti i contrasti, viene estratta la tabella dei risultati e i LFC vengono ristretti.

```
contrast_oe <- c("samplename", "MOV10_overexpression", "control")

res_tableOE_unshrunken <- results(dds_an, contrast=contrast_oe)

res_tableOE <- lfcShrink(dds_an, contrast=contrast_oe, res=res_tableOE_unshrunken,
type = "normal")

## using 'normal' for LFC shrinkage, the Normal prior from Love et al (2014).
##
## Note that type='apeglm' and type='ashr' have shown to have less bias than type='normal'.
## See ?lfcShrink for more details on shrinkage type, and the DESeq2 vignette.
## Reference: https://doi.org/10.1093/bioinformatics/bty895
```

Il nome fornito nel secondo elemento è il livello utilizzato come baseline. Ad esempio, se si osserva una variazione LFC di -2 , significa che l'espressione genica è più bassa in `Mov10_oe` rispetto al `controllo`.

4.4 Visualizzazione ed esplorazione dei risultati

Il plot MA mostra la media dei counts normalizzati rispetto ai LFC per tutti i geni analizzati. I geni che sono differenzialmente espressi in modo significativo sono colorati in blu.

Questo plot consente di visualizzare graficamente l'effetto del restringimento della LFC, mostrando allo stesso tempo l'entità dei fold change e la loro distribuzione rispetto all'espressione media; in generale, ci si aspetta di osservare geni significativi lungo l'intera gamma dei livelli di espressione.

```
plotMA(res_tableOE_unshrunken, ylim=c(-2,2))
```

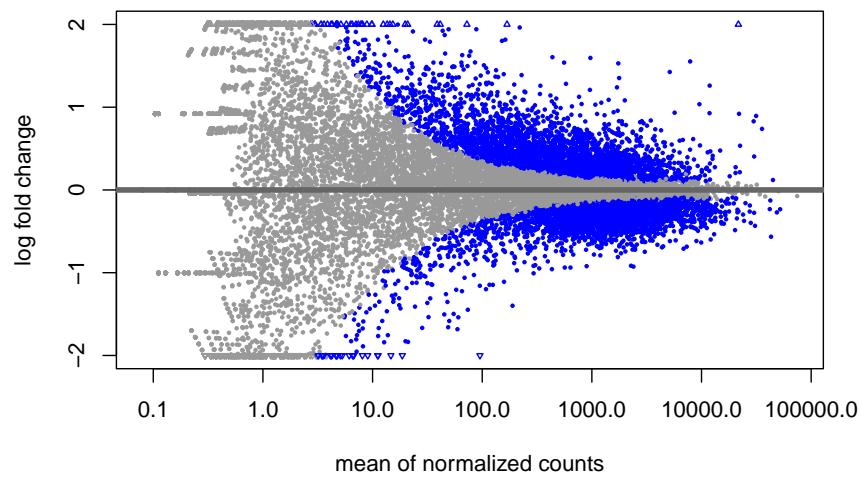


Figura 10: Media dei counts normalizzati rispetto ai log2 fold change. I geni differenzialmente espressi sono colorati in blu.

```
plotMA(res_tableOE, ylim=c(-2,2))
```

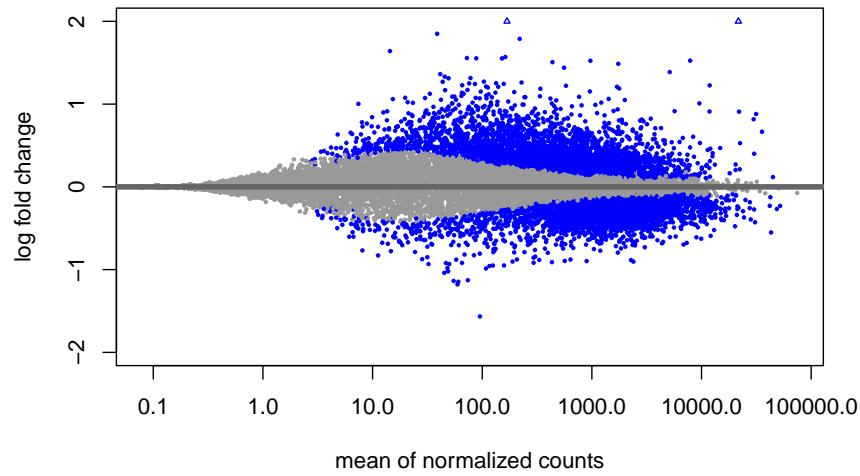


Figura 11: Effetto del restringimento dei LFC.

I risultati dell'analisi sono memorizzati in un dataframe, che contiene queste colonne:

- **baseMean**: media dei counts normalizzati per ogni campione;
- **log2FoldChange**: log2 fold change;

- `lfcSE`: standard error;
- `stat`: statistca Wald;
- `pvalue`: p-value del test di Wald;
- `padj`: p-value aggiustato BH.

```
res_tableOE
```

```
## log2 fold change (MAP): samplotype MOV10_overexpression vs control
## Wald test p-value: samplotype MOV10 overexpression vs control
## DataFrame with 23368 rows and 6 columns
##           baseMean log2FoldChange      lfcSE       stat     pvalue     padj
## <numeric>    <numeric>    <numeric>    <numeric>    <numeric>    <numeric>
## 1/2-SBSRNA4   45.652040   0.2665598  0.1890411  1.401464  0.1610752 0.2750250
## A1BG          61.093102   0.2080407  0.1747208  1.174510  0.2401909 0.3716156
## A1BG-AS1      175.665807  -0.0518245  0.1251773 -0.413922  0.6789312 0.7840468
## A1CF          0.237692    0.0125508  0.0482063  0.260351  0.7945932 NA
## A2LD1         89.617985   0.3429823  0.1608470  2.128033  0.0333343 0.0774672
## ...           ...
## ZYG11B        2973.949477 -0.0661925  0.0573605 -1.154002  0.24849951 0.3813641
## ZYX           2933.105330 -0.0614923  0.0689242 -0.892163  0.37230543 0.5157840
## ZZEF1         2132.254272 -0.1536289  0.0679300 -2.261879  0.02370489 0.0582162
## ZZZ3          2215.883805 -0.1617975  0.0611821 -2.644468  0.00818194 0.0237935
## tAKR          0.343415    -0.0199975  0.0581162 -0.344100  0.73077122 NA
```

È fondamentale aggiustare i p-value per controllare il tasso di falsi positivi (False Discovery Rate). Questo viene realizzato tramite il metodo di Benjamini-Hochberg (BH), che ordina i geni in base ai loro p-value e moltiplica ciascun p-value ordinato per il rapporto $m/rank$, dove m rappresenta il numero totale di test eseguiti e rank la posizione del gene nell'ordinamento.

La funzione `summary()` fornisce un riassunto dei risultati dell'analisi.

```
summary(res_tableOE)
```

```
##
## out of 19748 with nonzero total read count
## adjusted p-value < 0.1
## LFC > 0 (up)      : 3582, 18%
## LFC < 0 (down)    : 3847, 19%
## outliers [1]       : 0, 0%
## low counts [2]     : 3413, 17%
## (mean count < 3)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results
```

Oltre al numero di geni up- e down-regolati, la funzione riporta anche il numero di geni che sono stati testati (geni con read counts totale non nullo) e il numero di geni non inclusi nella correzione dei test multipli a causa di counts medio basso.

I geni up-regolati ($LFC > 0$) sono quelli il cui livello di espressione è aumentato significativamente nel gruppo `MOV10_overexpression` rispetto al `controllo` e sono il 18% del totale. I geni down-regolati ($LFC < 0$) sono quelli il cui livello di espressione è diminuito significativamente e sono il 19%.

Prima di estrarre i geni significativamente espressi, si aggiunge una soglia di FDR e LFC per ridurre il numero di geni significativi.

```
# setto soglie
padj_cutoff <- 0.05
lfc_cutoff <- 0.58
```

Il valore di `lfc_cutoff` è impostato su 0,58, questo si traduce in un incremento di circa 1,5 volte nell'espressione ($2^{0.58} \approx 1.5$). Considero quindi solo quei geni che mostrano un incremento o una diminuzione di almeno il 50% rispetto al controllo.

Per selezionare i geni significativi, si utilizza la funzione `filter()` della libreria dplyr per estrarre i geni usando le soglie definite sopra.

```
sigOE <- res_tableOE %>%
  data.frame() %>%
  rownames_to_column(var="gene") %>%
  as_tibble() %>%
  filter(padj < padj_cutoff & abs(log2FoldChange) > lfc_cutoff)
```

I risultati vengono ridotti, da 23368 a 870 geni significativi.

```
sigOE %>%
  arrange(padj)

## # A tibble: 870 x 7
##   gene    baseMean log2FoldChange  lfcSE   stat     pvalue      padj
##   <chr>     <dbl>          <dbl>  <dbl>  <dbl>     <dbl>      <dbl>
## 1 MOV10     21682.        4.77  0.103   46.2 0  2.45e-162
## 2 H1F0      7881.         1.53  0.0555  27.5 3.00e-166 1.12e-100
## 3 HIST1H1C   1741.         1.49  0.0684  21.7 2.06e-104 6.63e- 90
## 4 TXNIP      5134.         1.39  0.0676  20.5 1.62e- 93 2.70e- 83
## 5 NEAT1      21974.        0.909 0.0460  19.7 8.28e- 87 1.30e- 77
## 6 KLF10      1694.         1.21  0.0634  19.1 4.77e- 81 6.61e- 69
## 7 INSIG1     11873.        1.23  0.0678  18.1 4.55e- 73 1.06e- 64
## 8 NR1D1      970.          1.52  0.0875  17.4 1.72e- 67 3.50e- 61
## 9 WDFY1      1423.          1.06  0.0625  17.0 8.86e- 65 1.61e- 60
## 10 HSPA1A     31482.        0.880 0.0522  16.9 7.28e- 64 1.19e- 60
## # i 860 more rows

sigOE %>%
  count(log2FoldChange > 0) %>%
  mutate(percentuale = n/sum(n)*100)

## # A tibble: 2 x 3
##   `log2FoldChange > 0`     n percentuale
##   <lgl>                  <int>      <dbl>
## 1 FALSE                  198       22.8
## 2 TRUE                   672       77.2
```

Ora la percentuale dei geni up-regolati è del 77%, mentre quella dei geni down-regolati è del 23%. Questo suggerisce che la sovraespressione del gene ha un effetto più forte sull'espressione genica rispetto alla sua repressione.

Come plot finale, si esaminano le read counts normalizzate per un singolo gene nei vari gruppi; per fare ciò

si sceglie il gene che ha il p-value più piccolo.

```
name <- sig0E$gene[which.min(sig0E$padj)]
pc <- plotCounts(dds_norm, gene=name, intgroup="samplotype", returnData = TRUE)

ggplot(pc, aes(x = samplotype, y = count, color = samplotype)) +
  geom_point(position=position_jitter(w = 0.1,h = 0)) +
  geom_text_repel(aes(label = rownames(pc))) +
  theme_bw() +
  ggtitle(name) +
  theme(plot.title = element_text(hjust = 0.5)) # per centrare il titolo
```

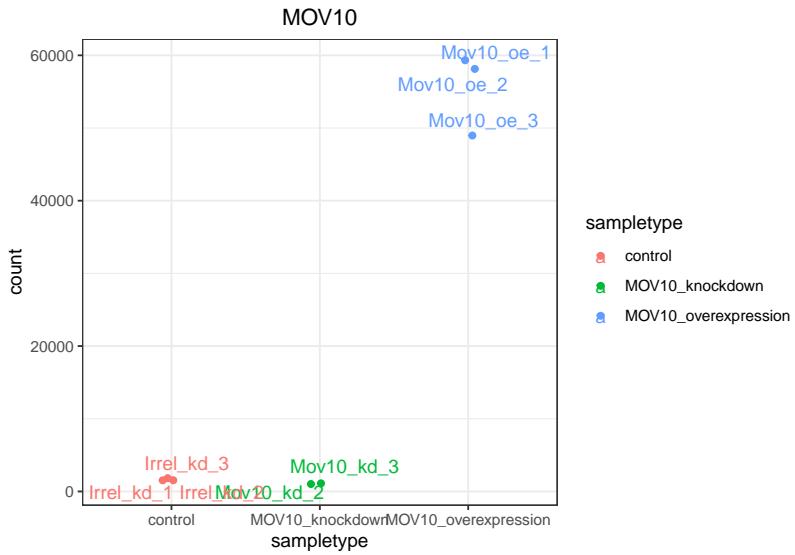


Figura 12: Counts per il gene con il p-value più piccolo.

L'espressione del gene è significativamente più alta nel gruppo **MOV10_overexpression**, questo suggerisce un'efficace sovraregolazione del gene. Nei gruppi **Control** e **MOV10_knockdown**, l'espressione del gene è trascurabile, indicando che il knockdown è riuscito e che non sono presenti livelli significativi di espressione anche nel controllo. Il plot quindi evidenzia una chiara differenza nell'espressione del gene tra i tre gruppi, supportando il successo delle condizioni sperimentali.

5 Acronimi

BH Benjamini-Hochberg

cDNA DNA complementare

FDR False Discovery Rate

GLM Generalized Linear Model

LFC Log Fold Change

MAP Maximum A Posteriori

mRNA RNA messaggero

MLE Maximum Likelihood Estimate

PCA Principal Component Analysis

RNA-seq RNA sequencing

Riferimenti

- [1] Bioconductor Support. 2017. What's the rationale for using the negative binomial distribution to model read counts? Recuperato da <https://support.bioconductor.org/p/84832/>
- [2] Biostars contributors. 2014. Why Does Rna-Seq Read Count Fit Poisson Distribution? Recuperato da <https://www.biostars.org/p/84445/>
- [3] Harvard Chan Bioinformatics Core (HBC). 2025. Differential gene expression workshop. Recuperato da https://github.com/hbctraining/DGE_workshop/tree/master
- [4] Peter J. Kenny, Hui Zhou, Michelle Kim, Geenu Skariah, Rushabh S. Khetani, Jenny Drnevich, Maria L. Arcila, Kenneth S. Kosik, e Stephanie Ceman. 2014. MOV10 and FMRP regulateAGO2 association with microRNA recognition elements. *Cell Reports* 9, 5 (dicembre 2014), 1729–1741. <https://doi.org/10.1016/j.celrep.2014.10.054>
- [5] Michael I Love, Wolfgang Huber, e Simon Anders. 2014. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology* 15, (2014), 550. <https://doi.org/10.1186/s13059-014-0550-8>
- [6] Wikipedia contributors. 2025. Negative binomial distribution. Recuperato da https://en.wikipedia.org/wiki/Negative_binomial_distribution
- [7] Wikipedia contributors. 2025. Fold change. Recuperato da https://en.wikipedia.org/wiki/Fold_change