

DESeq2 - Analyzing RNA-seq data

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

Claudia Costa, matr. 153256

2025-01-19

Contents

1	Introduzione	2
2	Differential Expression Analysis	2
2.1	Introduzione	2
2.1.1	Che cosa è un gene?	2
2.1.2	Che cosa è l'espressione genica?	2
2.2	Concetti di base dell'analisi differenziale di espressione (DEA)	3
2.2.1	Distribuzione dei counts	3
2.2.2	Importanza biologica della DEA	3
3	DESeq2	4
3.1	Teoria pacchetto	4
3.1.1	Fit modello	4
3.1.2	Stima size factor - metodo median of ratios	5
3.1.3	Stima dispersione per ogni gene	5
3.1.4	Stima log fold change	6
3.1.5	Test di ipotesi per l'espressione differenziale	8
4	Esempio su dataset	8
4.1	Distribuzione dei counts	8
4.2	Creazione oggetto DESeqDataSet	10
4.2.1	Analisi di controllo di qualità dei dati	11
4.3	Analisi di espressione differenziale	13
4.3.1	Test di ipotesi	15
4.3.2	Visualizzazione ed esplorazione dei risultati	16
5	Acronimi	20

1 Introduzione

In questa tesina viene descritto il pacchetto DESeq2 (vignetta), uno strumento specifico per l'analisi differenziale dell'espressione 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 differenziale dell'espressione genica, mentre nella seconda si approfondiscono le caratteristiche del pacchetto DESeq2 e i principi teorici su cui si basa.

Infine, nella terza parte viene presentato un esempio pratico di utilizzo di DESeq2, applicandolo ad una full count matrix del dataset di RNA-seq che fa parte dello studio Kenny PJ et al, Cell Rep 2014.

2 Differential Expression Analysis

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 cellulare, la struttura e la comunicazione.

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 produrre il suo prodotto funzionale. Questo processo può essere suddiviso in due fasi principali:

1. Trascrizione: DNA viene copiato in RNA (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 quali geni sono coinvolti in una malattia e quali effetti ha un farmaco su una popolazione cellulare.

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

2.2 Concetti di base dell'analisi differenziale di espressione (DEA)

L'analisi differenziale di espressione 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-sequencing (RNA-seq), che misurano quantitativamente l'espressione genica. Questo significa che per ciascun gene viene determinato un valore numerico, chiamato conteggio/count, che rappresenta la quantità di RNA messaggero (mRNA) prodotto, fornendo così un'indicazione precisa del livello di attività di quel gene in un dato campione.

L'obiettivo principale è **l'identificazione dei geni con espressione significativamente diversa tra condizioni**, quantificando l'entità del cambiamento (\log_2 fold change) e valutandone la rilevanza statistica per distinguere variazioni reali dal caso.

In altre parole, ciò che si vuole fare con l'analisi differenziale di espressione è confrontare due condizioni biologicamente distinte, con l'obiettivo di identificare i geni che presentano schemi di espressione diversi tra queste condizioni a causa di fenomeni biologici.

2.2.1 Distribuzione dei counts

Caratteristiche dei dati di conteggio dell'RNA-Seq: in un esperimento di sequenziamento dell'RNA, dopo che le reads sono state sequenziate e quantificate, ciò che otteniamo è essenzialmente una matrice di counts in cui le righe sono i geni e le colonne sono i campioni e il valore qui è chiamato counts. Queste counts non sono altro che il numero di reads mappate per quel gene in quel campione e il counts è l'unità più elementare per misurare l'espressione genica.

Controlli grafici, come quello riportato nell'analisi, mostrano che i dati dei counts non seguono una distribuzione normale. Si potrebbe pensare di utilizzare una distribuzione di Poisson, che descrive il numero di eventi che occorrono in un certo intervallo di tempo o in una regione spaziale. Essa però assume che la media e la varianza siano uguali (parametro λ), ma nei dati di RNA-seq la varianza è maggiore della media, quindi non è adatta per descrivere questi dati.

Viene perciò utilizzata la distribuzione binomiale negativa, che cattura la variabilità aggiuntiva dovuta a variazioni biologiche, introducendo un parametro di dispersione (α).

2.2.2 Importanza biologica della DEA

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

- Biologia delle malattie: identificare geni responsabili di malattie o 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 1 per identificare se un gene è differenzialmente espresso o no. Tutti i passaggi sono eseguiti automaticamente dalla funzione `DESeq()`.

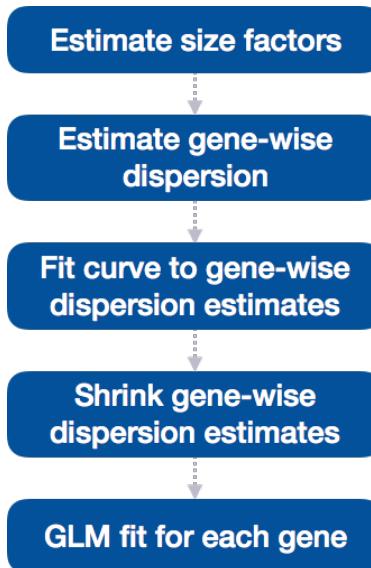


Figure 1: DESeq2 workflow

3.1.1 Fit modello

Il punto di partenza di un'analisi DESeq2 è una matrice dei counts K , con una riga per ogni gene i e una colonna per ogni campione j . L'entry della matrice K_{ij} indica il numero di reads sequenziate che sono state mappate sul gene i nel campione j .

Le read counts K_{ij} sono descritte con un GLM 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 - viene calcolato utilizzando il metodo median of ratios e viene descritto nella sezione successiva.

Per ogni gene viene fittato un modello lineare generalizzato (GLM), i quali sono un'estensione dei modelli di regressione lineare che permettono una forma più generale di espressione per la risposta media (che non segue necessariamente una distribuzione normale), utilizzando funzioni di link adeguate e considerando vari tipi di distribuzioni per la risposta. La funzione link trasforma il valore atteso della risposta per renderlo lineare rispetto ai predittori.

Viene utilizzato 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. Nel caso di un confronto tra due gruppi, come i campioni trattati e quelli di controllo, gli elementi della matrice di design indicano se un campione j è trattato o meno, e il fit GLM restituisce coefficienti che indicano la forza di espressione complessiva del gene e il log2 fold change tra trattamento e controllo.

3.1.2 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; sono considerati costanti all'interno di un campione, $s_{ij} = s_j$.

Viene calcolato impiegando tre step principali:

1. calcolo la media geometrica per ciascun gene (geometrica perché è più robusta agli outliers rispetto a quella aritmetica).
2. divido counts per la media geometrica.
3. calcolo la mediana dei rapporti (calcolati nello step 2).

Formalmente: $s_j = \text{median}_{i:K_i^R \neq 0} \frac{K_{ij}}{K_i^R}$, con $K_i^R = (\prod_{j=1}^m K_{ij})^{1/m}$

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

3.1.3 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 rappresenta quanto ci si può aspettare che un conteggio 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 2 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 dei log 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.

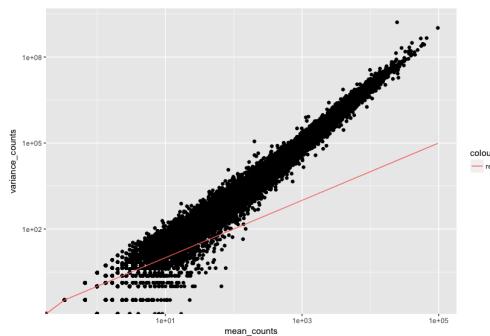


Figure 2: Plot della media vs varianza dei counts

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

Il processo inizia con la stima della dispersione di ciascun gene utilizzando il metodo della massima verosimiglianza (MLE), basato unicamente sui dati relativi al gene in questione. Tuttavia, queste stime iniziali possono essere rumorose, soprattutto per geni con bassi conteggi o alta dispersione.

Per ridurre il rumore e migliorare la precisione, viene adattata una curva smooth (curva rossa in figura 3) alle stime di dispersione di tutti i geni, tenendo conto della dipendenza dalla forza di espressione media. Questa curva rappresenta il valore atteso della dispersione per ciascun livello di espressione, fornendo un riferimento generale. Tuttavia, non cattura necessariamente le deviazioni specifiche dei singoli geni rispetto a questa tendenza.

A questo punto, viene stimata una distribuzione a priori (che rappresenta la conoscenza o le ipotesi iniziali su un parametro sconosciuto) per i valori di dispersione, basandosi sulla conoscenza derivata dal fit della curva e sulle proprietà osservate nei dati. La distribuzione a priori controlla automaticamente l'ampiezza del restringimento, regolando la quantità di “shrinkage” applicata ai valori iniziali di dispersione.

Infine, il modello utilizza questa distribuzione a priori in una seconda fase di stima, ottenendo le stime MAP (Massime a Posteriori) finali della dispersione (rappresentate dalle punte delle frecce blu). Questo restringimento spinge molte delle stime dei geni verso i valori previsti dalla curva fittata, riducendo così il rischio di falsi positivi causati da dispersioni sottostimate.

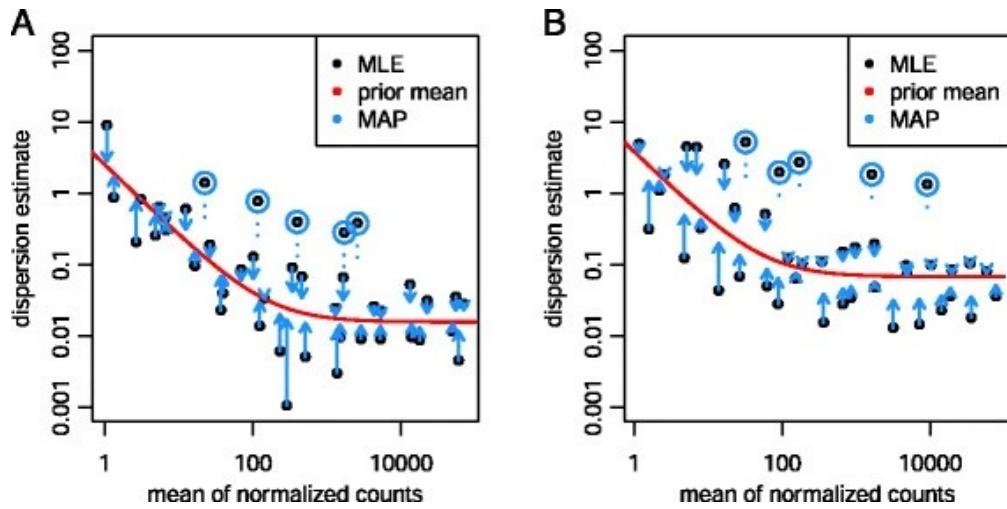


Figure 3: Stime di dispersione rispetto alla forza media di espressione

Tuttavia, non tutti i geni sono soggetti a questo processo di regolarizzazione. Alcuni geni, evidenziati con un cerchio blu, mostrano una variabilità di dispersione così elevata che DESeq2 presume che non seguano le ipotesi del modello. 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. Man mano che la dimensione del campione aumenta, il restringimento si riduce progressivamente fino a diventare trascurabile.

Questo metodo consente stime più robuste anche in contesti con informazioni limitate. Tuttavia, per geni con basse medie di espressione, le stime della varianza tendono a essere più ampie, portando a una maggiore variabilità tra le dispersioni stimate per questi geni.

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 (LFC) per geni con bassi read count.

Il log fold change è una misura che 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. Ad esempio,

un LFC di +1 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 (processo chiamato shrinkage), 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.

Come primo passo, vengono eseguiti dei fit di GLMs per ottenere le stime di massima verosimiglianza (MLE) dei LFC. Successivamente, una distribuzione normale centrata su zero viene fittata alla distribuzione osservata delle MLE su tutti i geni. Questa distribuzione normale viene quindi utilizzata come distribuzione a priori sui LFC in una seconda serie di fit GLM. Le stime finali dei LFC sono ottenute applicando il principio della massima a posteriori (MAP), come per la stima della dispersione.

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. Ad esempio, due geni con conteggi medi simili ma dispersioni diverse subiranno un restringimento diverso. Questo concetto è illustrato nella figura 4.

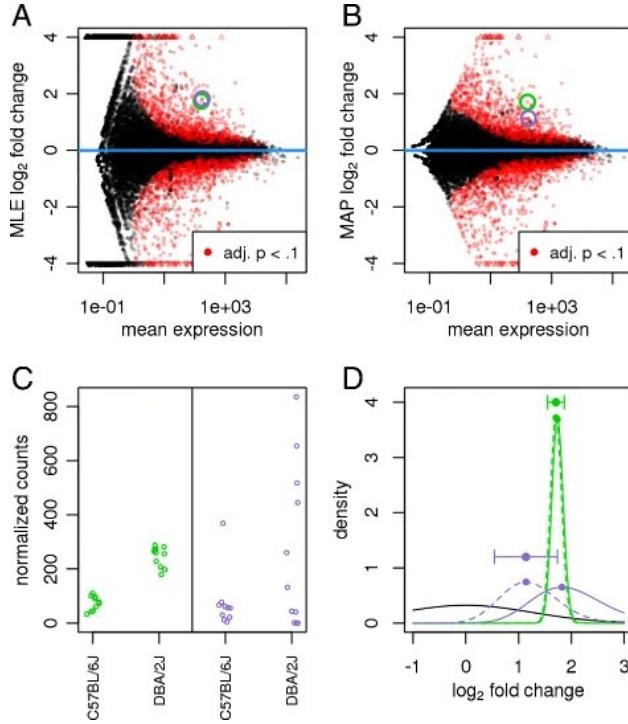


Figure 4: Effetto del restringimento sulle stime logaritmiche di fold change

Grafici della (A) stima MLE (cioè senza restringimento) e (B) MAP (cioè con restringimento) per le LFC, rispetto alla forza media di espressione. Piccoli triangoli nella parte superiore e inferiore dei grafici indicano i punti che cadrebbero al di fuori della finestra di tracciatura. Due geni con conteggi medi e variazioni logaritmiche MLE simili sono evidenziati con cerchi verdi e viola. (C) I conteggi (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 solide) e dei posteriori (linee tratteggiate) per i geni verde e viola e del priore (linea nera solida): 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).

3.1.5 Test di ipotesi per l'espressione differenziale

Dopo che i modelli sono stati adattati per ciascun gene, si può verificare se il coefficiente β_{ir} di ciascun modello differisce significativamente da zero.

Per i test di significatività, viene utilizzato un test di Wald: la stima dei LFC viene divisa per il suo errore standard, ottenendo una statistica-z, che viene confrontata con una distribuzione normale standard.

Se il p-value è piccolo, rifiutiamo l'ipotesi nulla e affermiamo che esiste un'evidenza contro l'ipotesi nulla (cioè che il gene è differenzialmente espresso).

Formalmente: $\beta_{ir}/\text{SE}(\beta_{ir})$ confrontato con una distribuzione normale standard.

4 Esempio su dataset

```
# per installare core pakage
# if (!require("BiocManager", quietly = TRUE))
#   install.packages("BiocManager")
# BiocManager::install()
```

Carico i dati e i metadati necessari, specificando che ho un header e che la prima colonna rappresenta i nomi delle righe.

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

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

Voglio 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$sampletype)

## [1] "MOV10 Knockdown"      "MOV10_overexpression" "control"
```

4.1 Distribuzione dei counts

Per visualizzare la distribuzione non normale dei counts, plotto un istogramma per il campione *Mov10_kd_2* e una versione zoomata.

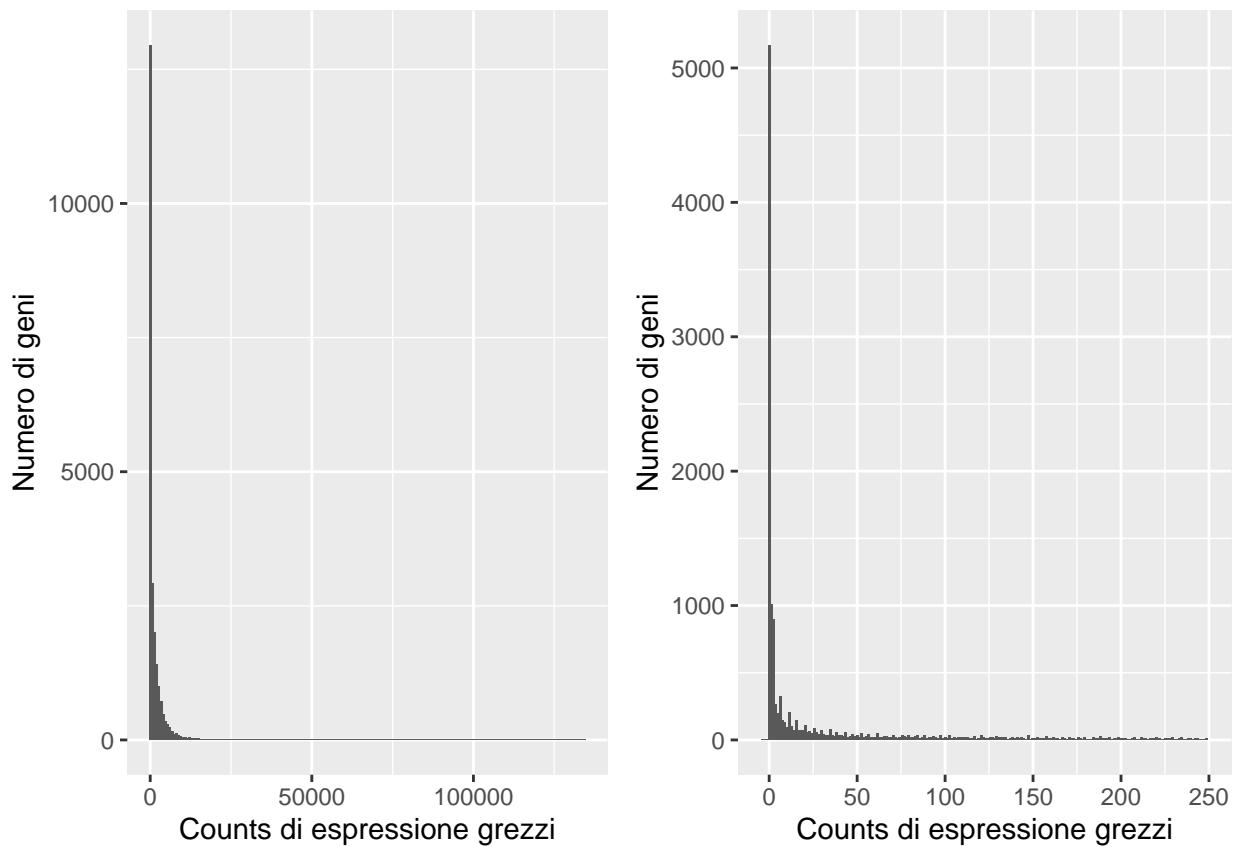
```

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

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

```



Si può vedere come la distribuzione dei counts sia asimmetrica e non normale, con molti geni con bassi conteggi.

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

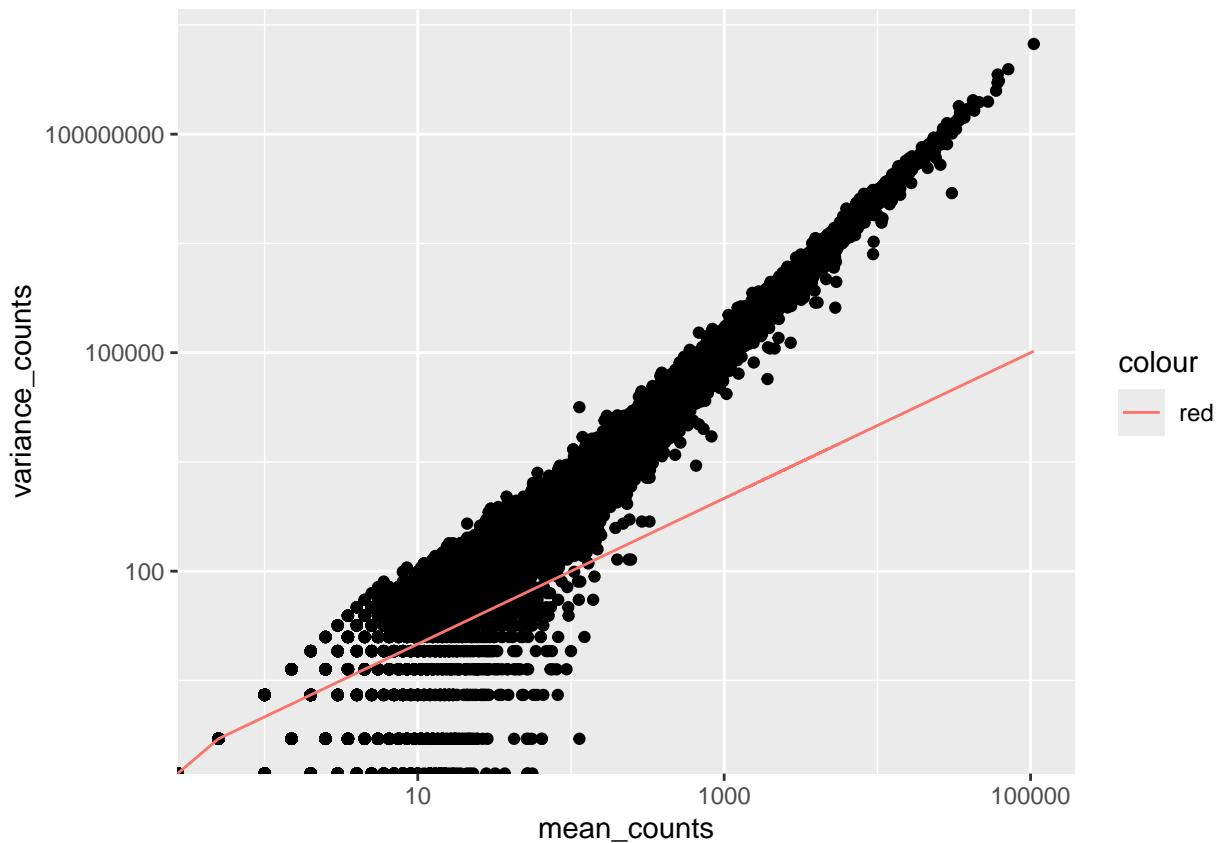
```

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()
```



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

4.2 Creazione oggetto DESeqDataSet

Prima di procedere con l'analisi, è meglio 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 `samplename`, che contiene tre livelli di 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 = ~ samplename)
```

```
## Warning in DESeqDataSet(se, design = design, ignoreRank): some variables in  
## design formula are characters, converting to factors
```

Per visualizzare i dati, è bene prima normalizzarli. Questo passaggio è in più, la funzione `DESeq()` esegue automaticamente tutti i passaggi necessari.

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

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

```
normalized_counts <- counts(dds_norm, normalized=TRUE)  
write.table(normalized_counts, file="normalized_counts.txt", sep="\t", quote=F, col.names=NA)
```

4.2.1 Analisi di controllo di qualità dei dati

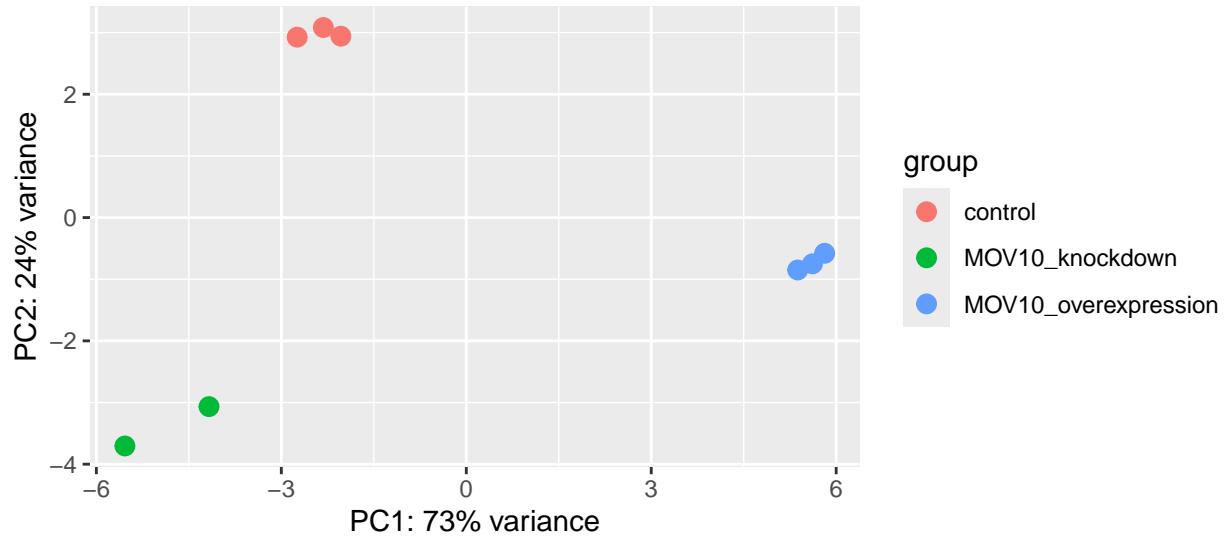
Per migliorare le distanze/clustering per i metodi di visualizzazione PCA e di clustering gerarchico, è necessario moderare la varianza rispetto alla media applicando la trasformazione *rlog* ai counts normalizzati. Questa funzione trasforma i dati di counts in scala log2 in modo da minimizzare le differenze 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="samplename")
```

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



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

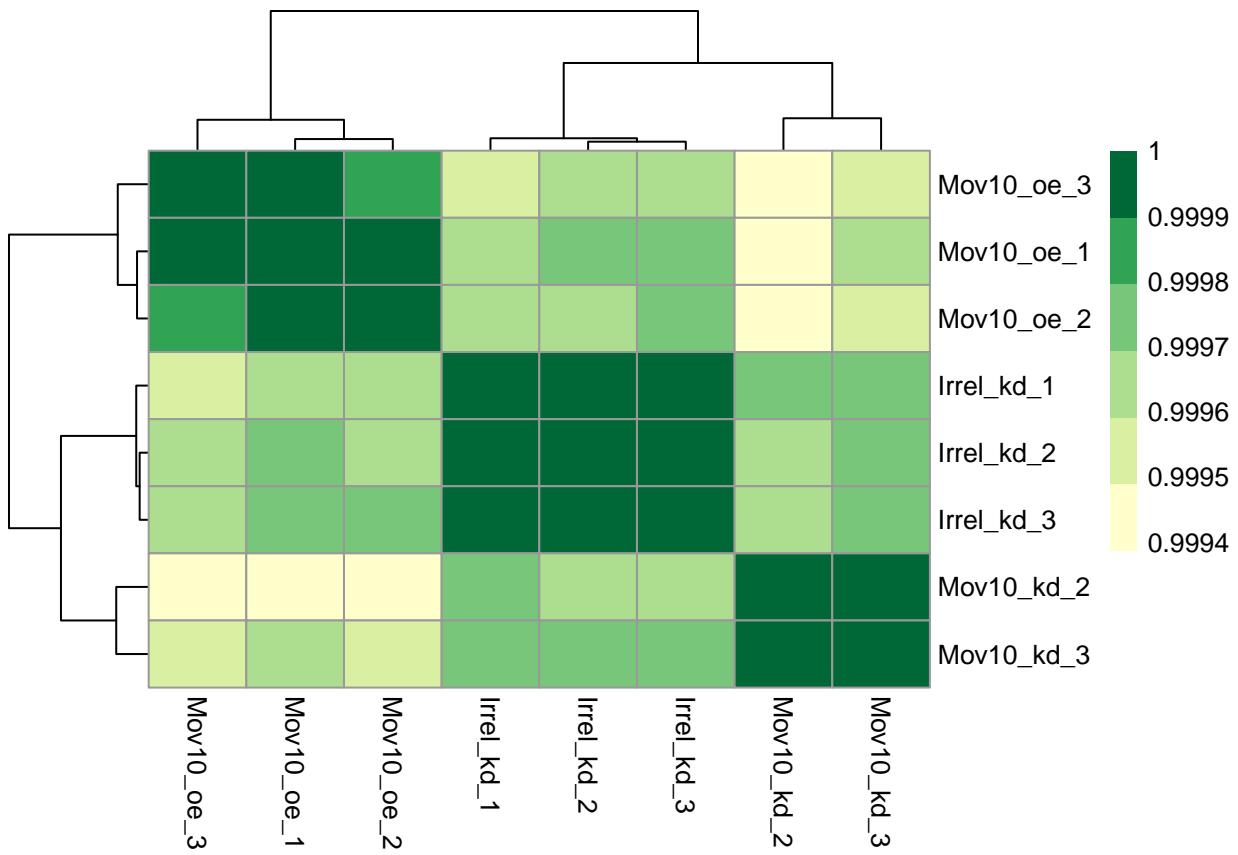
```
rld_mat <- assay(rld) # assay() funzione caricata dalle dipendenze di DESeq2
rld_cor <- cor(rld_mat)

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)
```



Osservo 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 di espressione differenziale

Uso 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
```

Controllo 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
```

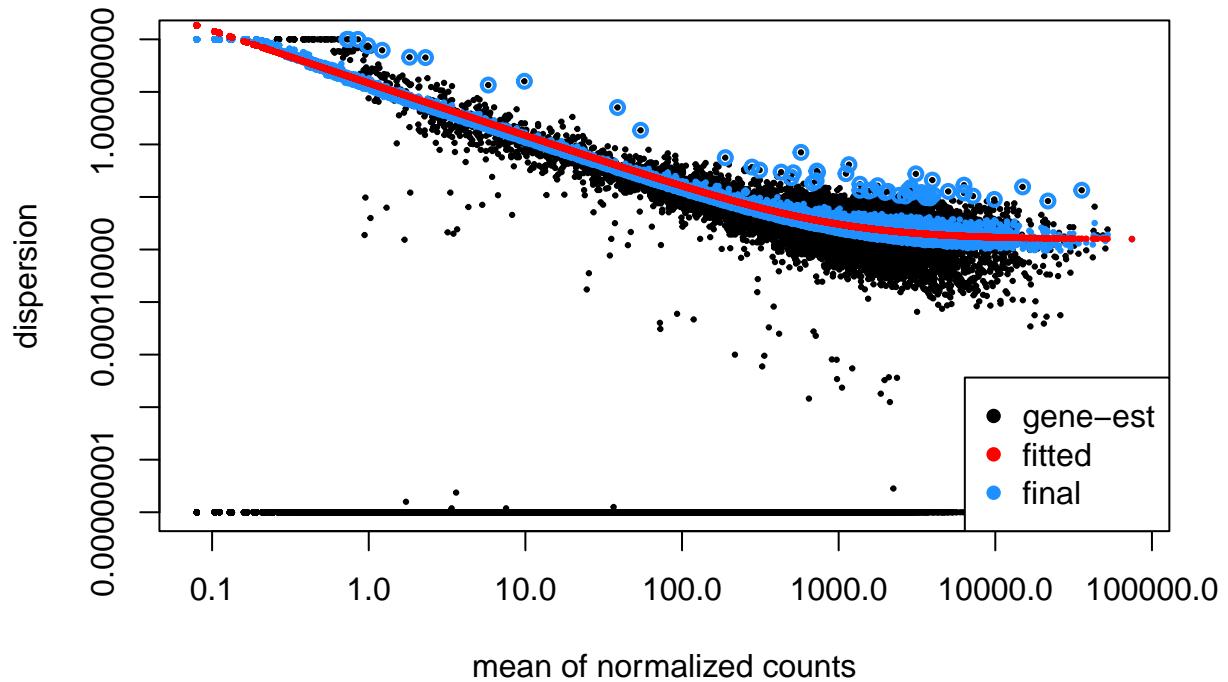
e numero totale di counts normalizzati per ogni campione:

```
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
```

Plot della dispersione stimata per ciascun gene rispetto alla media di espressione.

```
plotDispEsts(dds_an)
```



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

4.3.1 Test di ipotesi

Come descritto nella sezione 3.1.5, i coefficienti β_{ir} (ristretti/shrunken) rappresentano i LFC per ogni gruppo di campioni. Bisogna quindi testare se questi coefficienti sono significativamente diversi da zero, cioè che non vi sia espressione differenziale tra i gruppi di campioni.

Per indicare a DESeq2 i due gruppi che voglio confrontare bisogna usare i contrasti. Questi 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:

- senza fare nulla, 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

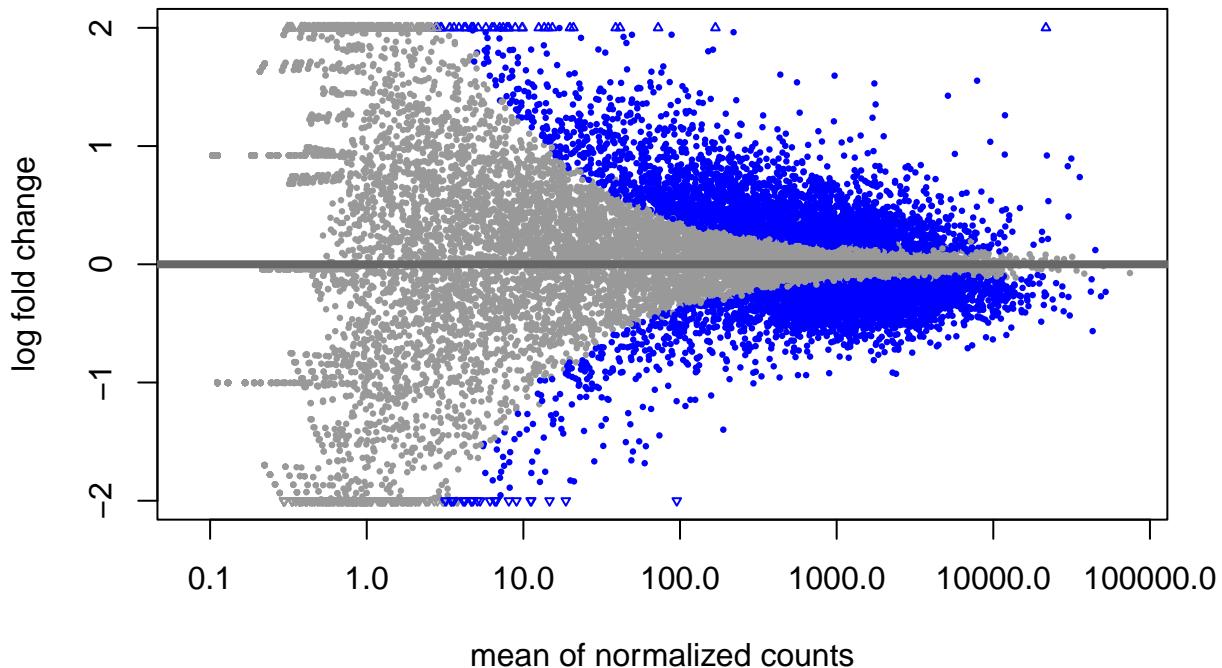
Definisco i contrasti, estraggo la tabella dei risultati e restringo i LFC.

Il nome fornito nel secondo elemento è il livello utilizzato come baseline. Ad esempio, se si osserva una variazione log2 fold di -2, significa che l'espressione genica è più bassa in Mov10_oe rispetto al controllo.

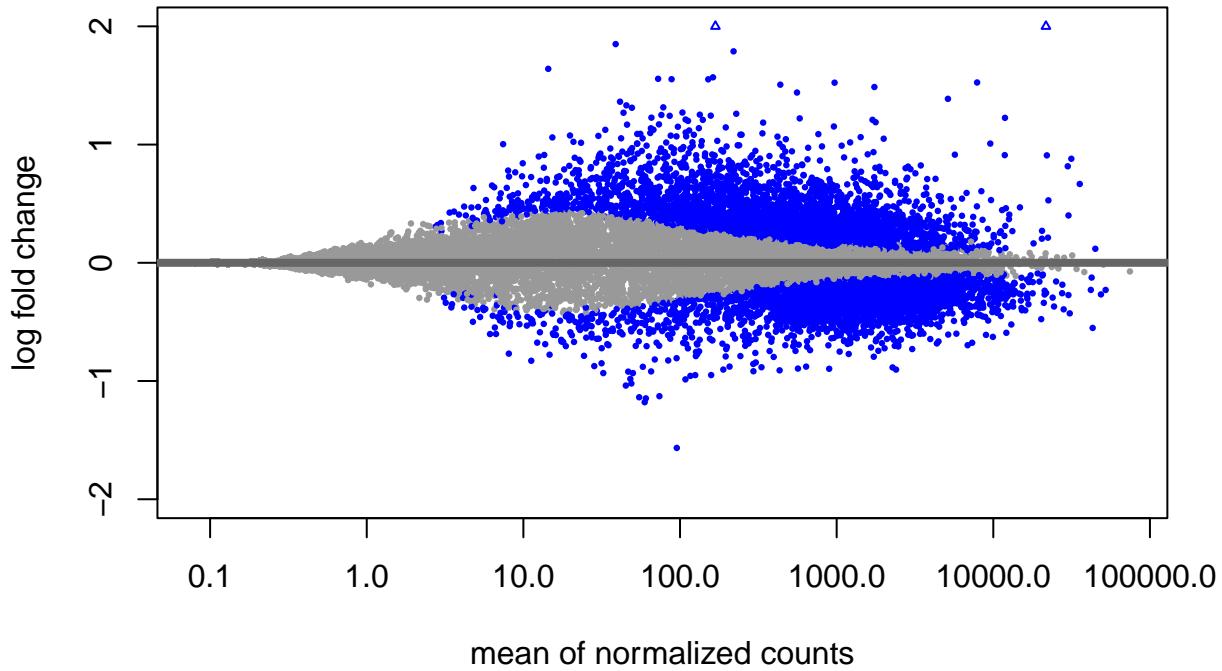
4.3.2 Visualizzazione ed esplorazione dei risultati

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))
```



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



I risultati dell'analisi sono memorizzati in una specie di dataframe, che contiene queste colonne: baseMean: media dei counts normalizzati per ogni campione; log2FoldChange: log2 fold change; lfcSE: standard error; stat: statistica 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, FDR). 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 un conteggio medio basso.

Prima di estrarre i geni significativamente espressi, aggiungo 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 diminuzione di almeno il 50% rispetto al controllo.

Per selezionare i geni significativi, utilizzo la funzione `filter()` di 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         0
## 2 H1F0      7881.         1.53  0.0555 27.5 3.00e-166 2.45e-162
## 3 HIST1H1C   1741.         1.49  0.0684 21.7 2.06e-104 1.12e-100
```

```

## 4 TXNIP      5134.      1.39  0.0676  20.5 1.62e- 93 6.63e- 90
## 5 NEAT1     21974.     0.909 0.0460  19.7 8.28e- 87 2.70e- 83
## 6 KLF10      1694.      1.21  0.0634  19.1 4.77e- 81 1.30e- 77
## 7 INSIG1    11873.     1.23  0.0678  18.1 4.55e- 73 1.06e- 69
## 8 NR1D1      970.       1.52  0.0875  17.4 1.72e- 67 3.50e- 64
## 9 WDFY1     1423.       1.06  0.0625  17.0 8.86e- 65 1.61e- 61
## 10 HSPA1A   31482.     0.880 0.0522  16.9 7.28e- 64 1.19e- 60
## # i 860 more rows

```

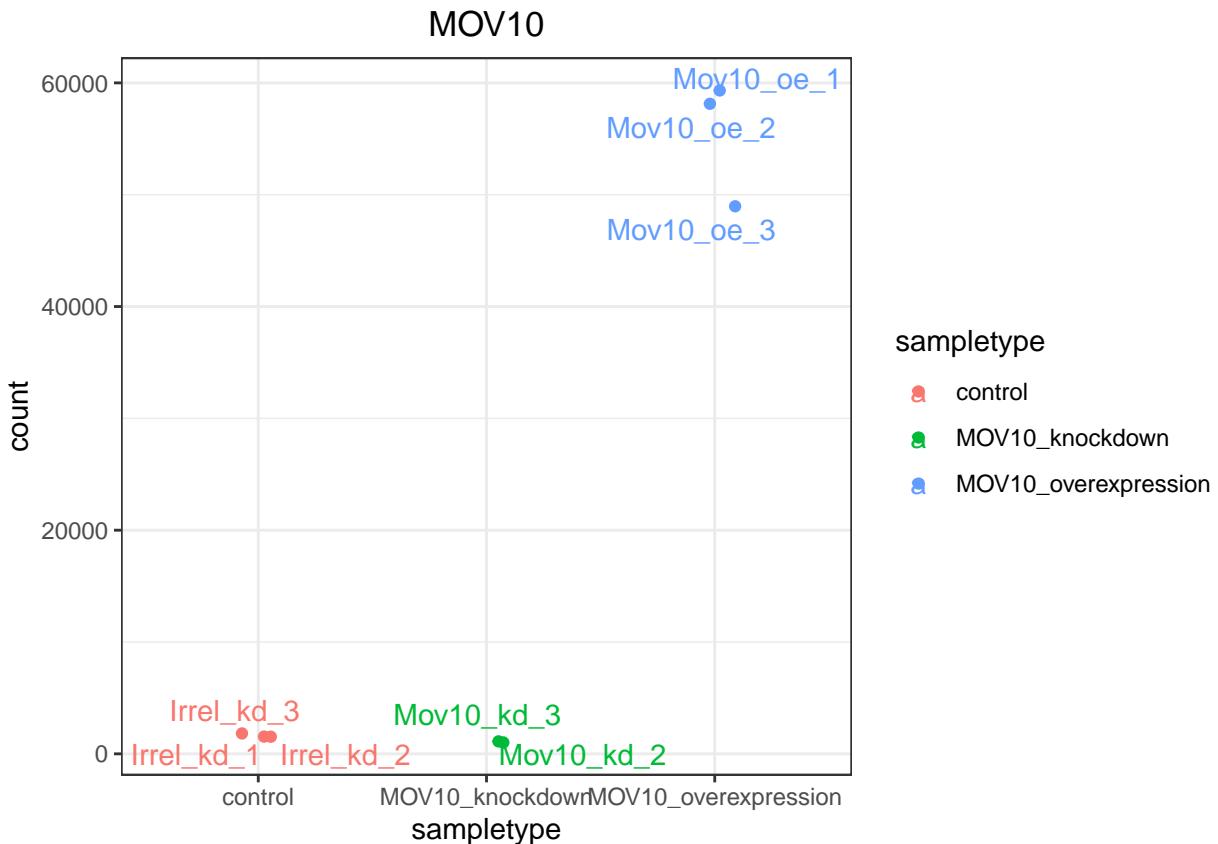
Come plot finale, esamino le read counts per un singolo gene nei vari gruppi; per fare ciò scelgo il gene che ha il p-value più piccolo.

```

name <- sigOE$gene[which.min(sigOE$padj)]
pc <- plotCounts(dds, 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))

```



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 ci sono 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

GLM generalized linear model

LFC log fold change

MAP maximum a posteriori

MLE maximum likelihood estimate

RNA-seq RNA sequencing