CS-E5875 High-Throughput Bioinformatics
RNA-seq analysis: differential expression

Harri Lähdesmäki

Department of Computer Science
Aalto University

November 10, 2023

Contents

Linear regression[1]

- Recall the multiple linear regression model

$$y_i = \beta_0 + \sum_{k=1}^{p} x_{ik}\beta_k + \epsilon_i = \mathbf{x}_i^T \boldsymbol{\beta} + \epsilon_i,$$

where

- y_i denotes the measured response for the ith sample/data point
- $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)^T$ denotes the regression coefficients
- $\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ip})^T$ denotes the predictors for the ith sample/data point, and
- ϵ_i denotes the Gaussian observation error for the ith measurement, $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$

[1] See e.g. (Agresti, 2015) or (Murphy, 2012) or any book on (generalized) linear models

Linear regression: vector notation

- Assuming n measurements $(\mathbf{x}_i, y_i)$, $i = 1, \ldots, n$

$$\mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \quad \text{and} \quad X = \begin{pmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_n^T \end{pmatrix}$$

the linear regression model can be written as

$$\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\epsilon},$$

where

$$\boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix}$$

and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 I_n)$ and I_n is the n-by-n identity matrix

Linear regression: likelihood

- Parameters of the linear regression model are $\theta = (\boldsymbol{\beta}, \sigma^2)$
- The likelihood for the linear regression model with Gaussian noise can be written as

$$\begin{aligned} L(\theta \mid X, \mathbf{y}) &\triangleq p(\mathbf{y} \mid X, \theta) \\ &= \mathcal{N}(\mathbf{y} \mid \boldsymbol{\mu}, \Sigma) \\ &= \mathcal{N}(\mathbf{y} \mid X\boldsymbol{\beta}, \sigma^2 I_n) \\ &= \prod_{i=1}^{n} \mathcal{N}(y_i \mid \mathbf{x}_i^T \boldsymbol{\beta}, \sigma^2) \\ &= \prod_{i=1}^{n} \mathcal{N}(y_i \mid \mu_i, \sigma^2), \end{aligned}$$

where $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_n)^T$, $\mu_i = \mathbb{E}[y_i] = \mathbf{x}_i^T \boldsymbol{\beta}$, and Σ denotes the expectation and covariance of random variable y_i, and σ^2 specifies uncertainty around the expected value

Parameter estimation for linear model with Gaussian noise

- The maximum likelihood estimate (MLE): choose parameters such that they maximize the likelihood L of the observed data (i.e., optimize w.r.t. model parameters)

$$\hat{\theta} = \arg\max_{\theta} L(\theta \mid X, \mathbf{y}) = \arg\max_{\theta} p(\mathbf{y} \mid X, \theta)$$

Parameter estimation for linear model with Gaussian noise

- The maximum likelihood estimate (MLE): choose parameters such that they maximize the likelihood L of the observed data (i.e., optimize w.r.t. model parameters)

$$\hat{\theta} = \arg\max_{\theta} L(\theta \mid X, \mathbf{y}) = \arg\max_{\theta} p(\mathbf{y} \mid X, \theta)$$

- Because logarithm is a strictly increasing function, it is equivalent to maximize the (natural) logarithm of the likelihood

$$\begin{aligned}
\ell(\theta) &= \log p(\mathbf{y} \mid X, \theta) = \log \prod_{i=1}^{n} p(y_i|\mathbf{x}_i, \theta) = \sum_{i=1}^{n} \log p(y_i|\mathbf{x}_i, \theta) \\
&= \sum_{i=1}^{n} \log \left[\left(\frac{1}{2\pi\sigma^2} \right)^{\frac{1}{2}} \exp\left(-\frac{1}{2\sigma^2} (y_i - \mathbf{x}_i^T \beta)^2 \right) \right] \\
&= -\frac{n}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^T \beta)^2
\end{aligned}$$

- Instead of maximizing $\ell(\theta)$ one can minimize $-\ell(\theta)$

Parameter estimation for linear model with Gaussian noise

- Maximum (or minimum) values of a (log) likelihood function w.r.t. parameters are obtained at parameter values where the gradient of the function w.r.t. parameters, i.e. partial derivatives, are zero
- For some models, the minimum / maximum can be obtained in a closed form
- The linear regression model with additive Gaussian noise is one such model:

$$\begin{aligned}\hat{\beta} &= (X^T X)^{-1} X^T \mathbf{y} \\ \hat{\sigma}^2 &= \frac{1}{n}(\mathbf{y} - \hat{\mathbf{y}})^T(\mathbf{y} - \hat{\mathbf{y}}) \\ &= \frac{1}{n}(\mathbf{y} - X\hat{\beta})^T(\mathbf{y} - X\hat{\beta}),\end{aligned}$$

assuming X has full rank so that the inverse $(X^T X)^{-1}$ exists

An illustration of the linear regression model with Gaussian noise

- An example of regression model fitting with model $y = \beta_0 + x\beta_1 + \epsilon$

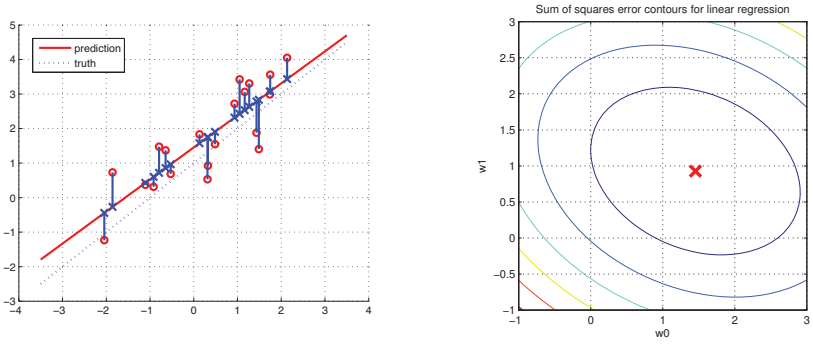


Figure: Figure from (Murphy, 2012)

Nonlinearities in the linear regression model

- To model non-linear function we can replace $\mathbf{x}$ with some non-linear function $\phi(\mathbf{x})$
  - So-called basis function expansion
  - Model is still linear in parameters, thus called as linear regression
- For example, polynomial basis functions

$$\mathbf{x} \triangleq \phi(\mathbf{x}) = (1, x, x^2, \ldots, x^d)^T$$

- The above theory works for general basis functions as well

An illustration of the linear regression model with Gaussian noise

- Examples of regression model fitting with linear and non-linear basis
  - $\phi(\mathbf{x}) = (1, x_1, x_2)^T$
  - $\phi(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_2^2)^T$

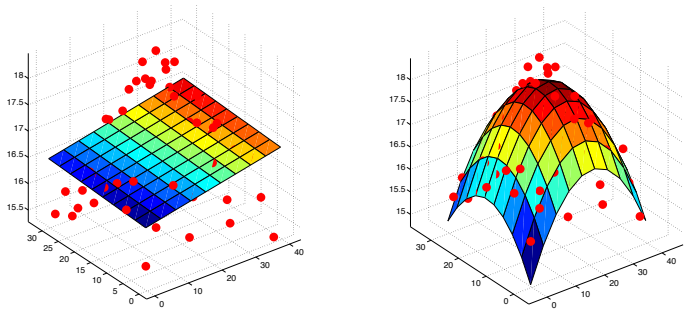


Figure: Figures from (Murphy, 2012)

Evaluation on linear regression models

- We are often interested in
  - Evaluating the model accuracy
  - Testing the significance of covariates/predictors of the model, either simultaneously or individually
- A natural measure of how well a model fits the data $\mathbf{y}$ is the so-called residual sum of squares

$$\begin{aligned} \text{RSS} &= (\mathbf{y} - \hat{\mathbf{y}})^T(\mathbf{y} - \hat{\mathbf{y}}) \\ &= (\mathbf{y} - X\hat{\boldsymbol{\beta}})^T(\mathbf{y} - X\hat{\boldsymbol{\beta}}) \\ &= \sum_{i=1}^{n}(y_i - \mathbf{x}_i^T\hat{\boldsymbol{\beta}})^2 \end{aligned}$$

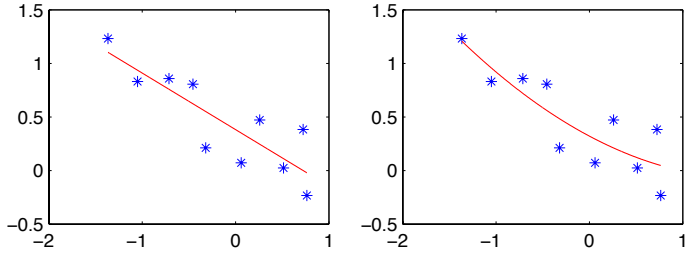
- RSS quantifies the amount of signal in $\mathbf{y}$ that a linear model cannot explain

Comparing two nested linear regression models

- Assume two nested multiple linear regression models
  - Model 1: $y_i = \beta_0 + \sum_{k=1}^{p_1} x_{ik}\beta_k + \epsilon_i$ (so-called reduced or null model with $p_1 + 1$ parameters)
  - Model 2: $y_i = \beta_0 + \sum_{k=1}^{p_1} x_{ik}\beta_k + \sum_{k=p_1+1}^{p_1+p_2} x_{ik}\beta_k + \epsilon_i$ (so-called full or alternative model with $p_1 + p_2 + 1$ parameters)

Comparing two nested linear regression models

- Assume two nested multiple linear regression models
  - Model 1: $y_i = \beta_0 + \sum_{k=1}^{p_1} x_{ik}\beta_k + \epsilon_i$ (so-called reduced or null model with $p_1 + 1$ parameters)
  - Model 2: $y_i = \beta_0 + \sum_{k=1}^{p_1} x_{ik}\beta_k + \sum_{k=p_1+1}^{p_1+p_2} x_{ik}\beta_k + \epsilon_i$ (so-called full or alternative model with $p_1 + p_2 + 1$ parameters)
- Example: compare regression models with one or two explanatory variables (first and second-order polynomials)
  - Model 1: $y_i = \beta_0 + x_{i1}\beta_1 + \epsilon_i$
  - Model 2: $y_i = \beta_0 + x_{i1}\beta_1 + x_{i1}^2\beta_2 + \epsilon_i$



Comparing two nested linear regression models: F statistic

- A test statistic that compares the RSS values between two models

$$F = \frac{(\mathrm{RSS}_1 - \mathrm{RSS}_2)/\mathrm{df}_1}{\mathrm{RSS}_2/\mathrm{df}_2},$$

where so-called degrees of freedom are

- $\mathrm{df}_1 = (1 + p_1 + p_2) - (1 + p_1) = p_2$
- $\mathrm{df}_2 = n - 1 - p_1 - p_2$

Comparing two nested linear regression models: F statistic

- A test statistic that compares the RSS values between two models

$$F = \frac{(\mathrm{RSS}_1 - \mathrm{RSS}_2)/\mathrm{df}_1}{\mathrm{RSS}_2/\mathrm{df}_2},$$

  where so-called degrees of freedom are
  - $\mathrm{df}_1 = (1 + p_1 + p_2) - (1 + p_1) = p_2$
  - $\mathrm{df}_2 = n - 1 - p_1 - p_2$

- Null hypothesis: the p_2 additional covariates included in model 2 do not provide significantly better fit
  - In other words, $H_0 : \beta_{p_1+1} = \ldots = \beta_{p_1+p_2} = 0$

- Null distribution: the F test statistic has F distribution with $(\mathrm{df}_1, \mathrm{df}_2)$ degrees of freedom

$\rightarrow$ Significance value from null hypothesis significance testing

Likelihood ratio test

- Let $L(\hat{\theta}_1 \mid X, \mathbf{y})$ and $L(\hat{\theta}_2 \mid X, \mathbf{y})$ denote the maximum likelihoods for the two nested linear models, respectively
- The likelihood ratio measures how many times less likely the data is under the reduced model (null hypothesis) than the full model (alternative hypothesis)

$$\Lambda(\mathbf{y}) = \frac{L(\hat{\theta}_1 \mid X, \mathbf{y})}{L(\hat{\theta}_2 \mid X, \mathbf{y})}$$

- Intuition:
  - Values of $\Lambda(\mathbf{y})$ close to 1 indicate no difference between the null and alternative models
  - Values close to 0 indicate that the alternative model can explain the data much better

Likelihood ratio test

- Let $L(\hat{\theta}_1 \mid X, \mathbf{y})$ and $L(\hat{\theta}_2 \mid X, \mathbf{y})$ denote the maximum likelihoods for the two nested linear models, respectively
- The likelihood ratio measures how many times less likely the data is under the reduced model (null hypothesis) than the full model (alternative hypothesis)

$$\Lambda(\mathbf{y}) = \frac{L(\hat{\theta}_1 \mid X, \mathbf{y})}{L(\hat{\theta}_2 \mid X, \mathbf{y})}$$

- Intuition:
  - Values of $\Lambda(\mathbf{y})$ close to 1 indicate no difference between the null and alternative models
  - Values close to 0 indicate that the alternative model can explain the data much better
- An asymptotic result for nested models:
  - When $n \to \infty$, the test statistic $-2 \log \Lambda(\mathbf{y})$ is chi-squared distributed with degrees of freedom equal to $\text{df} = |\theta_2| - |\theta_1|$, i.e. the difference in the number of free parameters between the two models
- This is a lot more general test than the F-test in that observation likelihoods do not need to Gaussians or the underlying model does not need to be linear

The likelihood ratio test for the linear Gaussian model

- For the two nested linear regression models with Gaussian noise, the likelihood ratio test can be written as

$$\begin{aligned}\Lambda(\mathbf{y}) &= -2\log\frac{\max_{\theta_1} L(\theta_1 \mid X, \mathbf{y})}{\max_{\theta_2} L(\theta_2 \mid X, \mathbf{y})} \\ &= -2\log\frac{L(\hat{\theta}_1 \mid X, \mathbf{y})}{L(\hat{\theta}_2 \mid X, \mathbf{y})} \\ &= \ldots = \left(1 + \frac{\mathrm{RSS}_1 - \mathrm{RSS}_2}{\mathrm{RSS}_2}\right)^{-n/2} \\ &= \left(1 + \frac{p_2}{n-1-p_1-p_2} F\right)^{-n/2}\end{aligned}$$

Contents

Generalized linear models

- In the standard linear regression models the response variable is assumed to have the Gaussian distribution
- Generalized linear models (GLM) are a generalization of linear regression models where the response variables can have an error distribution other than the normal distribution
- In commonly used GLMs the response variable is assumed to have a distribution in the exponential family, including e.g.
  - Normal, exponential, beta, gamma, Bernoulli, Poisson, etc. distributions

Generalized linear models: link function

- Recall that in the case of Gaussian likelihood, $\mathbb{E}[y_i] = \mu_i = \mathbf{x}_i^T \boldsymbol{\beta}$
- In GLMs, the mean of the random variable y_i, $\mathbb{E}[y_i] = \mu_i$, is assumed to depend on a linear model via an invertible link function g

$$g(\mu_i) = \mathbf{x}_i^T \boldsymbol{\beta}$$

Generalized linear models: link function

- Recall that in the case of Gaussian likelihood, $\mathbb{E}[y_i] = \mu_i = \mathbf{x}_i^T \boldsymbol{\beta}$
- In GLMs, the mean of the random variable y_i, $\mathbb{E}[y_i] = \mu_i$, is assumed to depend on a linear model via an invertible link function g

$$g(\mu_i) = \mathbf{x}_i^T \boldsymbol{\beta}$$

- Because g is invertible

$$\mathbb{E}[y_i] = \mu_i = g^{-1}(\mathbf{x}_i^T \boldsymbol{\beta})$$

- Free to choose $g(\cdot)$ as long as it is invertible and $g^{-1}(\cdot)$ has appropriate range
- Note: in the Gaussian linear model, the link function $g(\cdot)$ is the identify function

Generalized linear models

- GLM is obtained by using linear model together with the link function to parameterize an exponential distribution
- For example, GLM model for binary-valued data $y \in \{0, 1\}$ that has the Bernoulli distribution would be

$$p(y \mid \mathbf{x}, \boldsymbol{\beta}) = \text{Bernoulli}(y \mid g^{-1}(\mathbf{x}^T\boldsymbol{\beta})) = \begin{cases} 1, & \text{with probability } g^{-1}(\mathbf{x}^T\boldsymbol{\beta}) \\ 0, & \text{with probability } 1 - g^{-1}(\mathbf{x}^T\boldsymbol{\beta}) \end{cases}$$

where $g^{-1}(\cdot)$ maps the real line to an interval $[0, 1]$

Generalized linear models

- GLM is obtained by using linear model together with the link function to parameterize an exponential distribution
- For example, GLM model for binary-valued data $y \in \{0, 1\}$ that has the Bernoulli distribution would be

$$p(y \mid \mathbf{x}, \boldsymbol{\beta}) = \text{Bernoulli}(y \mid g^{-1}(\mathbf{x}^T\boldsymbol{\beta})) = \begin{cases} 1, & \text{with probability } g^{-1}(\mathbf{x}^T\boldsymbol{\beta}) \\ 0, & \text{with probability } 1 - g^{-1}(\mathbf{x}^T\boldsymbol{\beta}) \end{cases}$$

  where $g^{-1}(\cdot)$ maps the real line to an interval $[0, 1]$
- For example, GLM model for continuous-valued data $y \in \mathbb{R}$ that has the Gaussian distribution would be

$$p(y \mid \mathbf{x}, \boldsymbol{\beta}, \sigma^2) = \mathcal{N}(y \mid g^{-1}(\mathbf{x}^T\boldsymbol{\beta}), \sigma^2) = \mathcal{N}(y \mid \mathbf{x}^T\boldsymbol{\beta}, \sigma^2)$$

Generalized linear models

- GLM is obtained by using linear model together with the link function to parameterize an exponential distribution
- For example, GLM model for binary-valued data $y \in \{0, 1\}$ that has the Bernoulli distribution would be

$$p(y \mid \mathbf{x}, \boldsymbol{\beta}) = \text{Bernoulli}(y \mid g^{-1}(\mathbf{x}^T\boldsymbol{\beta})) = \begin{cases} 1, & \text{with probability } g^{-1}(\mathbf{x}^T\boldsymbol{\beta}) \\ 0, & \text{with probability } 1 - g^{-1}(\mathbf{x}^T\boldsymbol{\beta}) \end{cases}$$

  where $g^{-1}(\cdot)$ maps the real line to an interval $[0, 1]$
- For example, GLM model for continuous-valued data $y \in \mathbb{R}$ that has the Gaussian distribution would be

$$p(y \mid \mathbf{x}, \boldsymbol{\beta}, \sigma^2) = \mathcal{N}(y \mid g^{-1}(\mathbf{x}^T\boldsymbol{\beta}), \sigma^2) = \mathcal{N}(y \mid \mathbf{x}^T\boldsymbol{\beta}, \sigma^2)$$

- Variance of a GLM can follow the variance of the exponential family distribution or may be defined as a function $V(\cdot)$ of the predicted value
- For example, for the Gaussian linear model

$$\text{Var}(y_i) = \sigma^2 \quad \text{or} \quad V(\mu_i, \phi) = V(g^{-1}(\mathbf{x}_i^T\boldsymbol{\beta}), \phi)$$

Generalized linear models: Poisson example

- Poisson distribution is a probability distribution for discrete-valued random variable that can take values $0, 1, 2, \ldots$
- The probability mass function for a Poisson distributed random variable y is

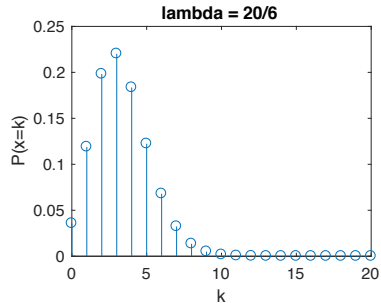
$$p(y \mid \lambda) = \text{Poisson}(y \mid \lambda) = \frac{\lambda^y \exp(-\lambda)}{y!},$$

where $\lambda > 0$ is a positive rate parameter

- The mean and variance of a Poisson distribution are

$$\mathbb{E}[y] = \lambda \quad \text{and} \quad \text{Var}(y) = \lambda$$

- An example of the Poisson distribution with $\lambda = 20/6$



Generalized linear models: Poisson example

- GLM for Poisson distributed response variables $\mathbf{Y} = (Y_1, \ldots, Y_n)^T$, i.e., non-negative count data where each $Y_i \in \{0, 1, 2, \ldots\}$
- Poisson rate parameter(s) $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_n)^T$ must be positive, so logarithmic link function is appropriate

$$\log \lambda_i = \mathbf{x}_i^T \boldsymbol{\beta} \quad \Leftrightarrow \quad \lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta})$$

  and therefore

$$\mathbb{E}[y_i] = \lambda_i = \exp(\mathbf{x}_i \boldsymbol{\beta})$$

- The variance is defined directly by the Poisson distribution, i.e., $\mathrm{Var}(Y_i) = \lambda_i = \exp(\mathbf{x}_i \boldsymbol{\beta})$

Generalized linear models: Poisson example

- GLM for Poisson distributed response variables $\mathbf{Y} = (Y_1, \ldots, Y_n)^T$, i.e., non-negative count data where each $Y_i \in \{0, 1, 2, \ldots\}$
- Poisson rate parameter(s) $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_n)^T$ must be positive, so logarithmic link function is appropriate

$$\log \lambda_i = \mathbf{x}_i^T \boldsymbol{\beta} \quad \Leftrightarrow \quad \lambda_i = \exp(\mathbf{x}_i^T \boldsymbol{\beta})$$

  and therefore

$$\mathbb{E}[y_i] = \lambda_i = \exp(\mathbf{x}_i \boldsymbol{\beta})$$

- The variance is defined directly by the Poisson distribution, i.e., $\mathrm{Var}(Y_i) = \lambda_i = \exp(\mathbf{x}_i \boldsymbol{\beta})$
- Likelihood of Poisson GLM model for the observed data $\mathbf{y} = (y_1, \ldots, y_n)^T$ is then

$$L(\boldsymbol{\beta} \mid X, \mathbf{y}) = \prod_{i=1}^{n} \mathrm{Poisson}(y_i \mid \lambda_i) = \prod_{i=1}^{n} \frac{\lambda_i^{y_i} \exp(-\lambda_i)}{y_i!} = \prod_{i=1}^{n} \frac{\exp(\mathbf{x}_i \boldsymbol{\beta})^{y_i} \exp(-\exp(\mathbf{x}_i \boldsymbol{\beta}))}{y_i!}$$

Fitting generalized linear models

- GLMs are typically estimated using maximum likelihood (or Bayesian) approach
- Maximum likelihood estimate:

$$\hat{\boldsymbol{\beta}} = \arg\max_{\boldsymbol{\beta}} L(\boldsymbol{\beta} \mid X, \mathbf{y})$$

- Note that for GLMs no closed form solutions exist, so numerical methods must be used
  - Gradient-based optimization methods

Hypothesis testing with GLMs

- For GLMs the null hypothesis is often stated by restricting the parameter vector

$$H_0 : \quad \boldsymbol{\beta} \in \Theta_0 \subset \mathbb{R}^{p+1}$$

- Consequently, the alternative hypothesis is defined via the complement of Θ_0, i.e., $\Theta_0^C = \mathbb{R}^{p+1} \setminus \Theta_0$

$$H_1 : \quad \boldsymbol{\beta}' \in \Theta_0^C$$

Hypothesis testing with GLMs

- For GLMs the null hypothesis is often stated by restricting the parameter vector

$$H_0 : \quad \boldsymbol{\beta} \in \Theta_0 \subset \mathbb{R}^{p+1}$$

- Consequently, the alternative hypothesis is defined via the complement of Θ_0, i.e., $\Theta_0^C = \mathbb{R}^{p+1} \setminus \Theta_0$

$$H_1 : \quad \boldsymbol{\beta}' \in \Theta_0^C$$

- For example, if one is interested in testing a single predictor x_i, then
  - $H_0 : \ \beta_i = 0$, or effectively $\boldsymbol{\beta} \in \mathbb{R}^p$
  - $H_1 : \ \beta_i \neq 0$, or effectively $\boldsymbol{\beta}' \in \mathbb{R}^{p+1}$

Hypothesis testing with GLMs

- For GLMs the null hypothesis is often stated by restricting the parameter vector

$$H_0: \quad \boldsymbol{\beta} \in \Theta_0 \subset \mathbb{R}^{p+1}$$

- Consequently, the alternative hypothesis is defined via the complement of Θ_0, i.e., $\Theta_0^C = \mathbb{R}^{p+1} \setminus \Theta_0$

$$H_1: \quad \boldsymbol{\beta}' \in \Theta_0^C$$

- For example, if one is interested in testing a single predictor x_i, then
  - $H_0: \ \beta_i = 0$, or effectively $\boldsymbol{\beta} \in \mathbb{R}^p$
  - $H_1: \ \beta_i \neq 0$, or effectively $\boldsymbol{\beta}' \in \mathbb{R}^{p+1}$
- Hypothesis testing can be implemented using e.g. the likelihood ratio test
- An asymptotic result for nested models: when $n \to \infty$, the test statistic $-2 \log \frac{\max_{\boldsymbol{\beta}_1} L(\boldsymbol{\beta}_1 | X, \mathbf{y})}{\max_{\boldsymbol{\beta}_2} L(\boldsymbol{\beta}_2 | X, \mathbf{y})}$ is chi-squared distributed with degrees of freedom equal to the difference in dimensionality of Θ_0 and Θ_0^C

Contents

Differential gene expression analysis

On the next slides we motivate the use of a negative binomial distribution by the following reasoning:

- Multinomial sampling across all genes...
- ...leads to binomial sampling for a single gene...
- ...leads to Poisson approximation for a single gene...
- ...leads to negative binomial model to account for larger variance

Multinomial distribution

Consider the following:

- A dice that has N different outcomes
- Each on the N outcomes is chosen randomly with probability p_i, where $\sum_{i=1}^{N} p_i = 1$
- When a dice is rolled once, one of the outcomes will be chosen randomly
- Assume an experiment where the dice is rolled n times (i.i.d.)
- Denote the number of times each outcome is observed by $\mathbf{x} = (x_1, \ldots, x_N)$
- This corresponds to multinomial sampling distribution

Multinomial distribution

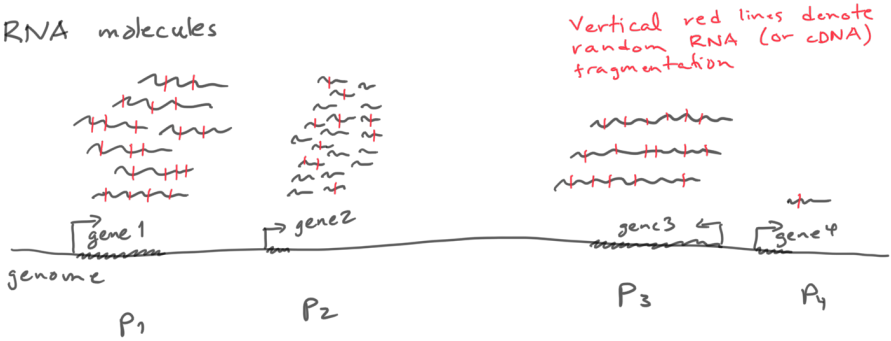
Consider the following:

- A dice that has N different outcomes
- Each on the N outcomes is chosen randomly with probability p_i, where $\sum_{i=1}^{N} p_i = 1$
- When a dice is rolled once, one of the outcomes will be chosen randomly
- Assume an experiment where the dice is rolled n times (i.i.d.)
- Denote the number of times each outcome is observed by $\mathbf{x} = (x_1, \ldots, x_N)$
- This corresponds to multinomial sampling distribution

- Denote the N probabilities by $\mathbf{p} = (p_1, \ldots, p_N)$
- The probability mass function of the random variable $X = (X_1, \ldots, X_N)$ that has the multinomial distribution with $\mathbf{p}$ and n:

$$
\begin{aligned}
&\mathrm{Multinomial}(\mathbf{x}; n, \mathbf{p}) \\
&= P(X_1 = x_1, \ldots, X_N = x_N) \\
&= \begin{cases} \frac{N!}{x_1! \ldots x_N!} p_1^{x_1} p_2^{x_2} \cdots p_N^{x_N}, & \text{if } x_1 + \ldots + x_N = n \\ 0, & \text{otherwise} \end{cases}
\end{aligned}
$$

Multinomial sampling distribution for RNA-seq



RNA molecules' relative proportions, where

$$\sum_{i=1}^{4} p_i = 1$$

Multinomial sampling distribution for RNA-seq

- N different outcomes for a dice correspond to genes: e.g. in human $N \approx 20,000$
- Probability p_i corresponds to the proportion of RNA fragments from gene i (note the effect of length of gene i)
- "One roll of a dice" corresponds to measuring a single RNA fragment for one specific gene from a very large pool of RNA fragments
- A sequencing run can produce e.g. 10M-1B sequencing reads, i.e., for example $n = 10^9$
- At the end of the RNA-seq experiment, pre-processing and alignment, $\mathbf{x} = (x_1, \ldots, x_N)$ denotes the number of reads mapped to each gene, where $x_1 + \ldots + x_N = n$ (assuming all n sequences can be aligned uniquely)

$\rightarrow$ For a single sample, we can assume that read counts for genes (or transcripts) have a multinomial (sampling) distribution

Multinomial sampling distribution for RNA-seq

- N different outcomes for a dice correspond to genes: e.g. in human $N \approx 20,000$
- Probability p_i corresponds to the proportion of RNA fragments from gene i (note the effect of length of gene i)
- "One roll of a dice" corresponds to measuring a single RNA fragment for one specific gene from a very large pool of RNA fragments
- A sequencing run can produce e.g. 10M-1B sequencing reads, i.e., for example $n = 10^9$
- At the end of the RNA-seq experiment, pre-processing and alignment, $\mathbf{x} = (x_1, \ldots, x_N)$ denotes the number of reads mapped to each gene, where $x_1 + \ldots + x_N = n$ (assuming all n sequences can be aligned uniquely)

$\rightarrow$ For a single sample, we can assume that read counts for genes (or transcripts) have a multinomial (sampling) distribution

- However, the use of multinomial is somewhat challenging because we would need to model all genes at the same time

Binomial distribution

- Consider a binary-valued random variable that takes value 1 with probability p and value 0 with probability $1 - p$
- For example, the probability that we obtain a sequencing read from gene i is $p = p_i$, and the probability that we obtain a sequencing read from any other gene is $1 - p = \sum_{j \neq i} p_j$

Binomial distribution

- Consider a binary-valued random variable that takes value 1 with probability p and value 0 with probability $1 - p$
- For example, the probability that we obtain a sequencing read from gene i is $p = p_i$, and the probability that we obtain a sequencing read from any other gene is $1 - p = \sum_{j \neq i} p_j$
- Take n independent random realizations of the binary-valued random variable
- Let X denote the number of success in n realizations
- The probability of getting exactly $X = k$ successes in n trials is given by probability mass function of the binomial distribution

$$\mathrm{B}(k; n, p) = P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k}$$

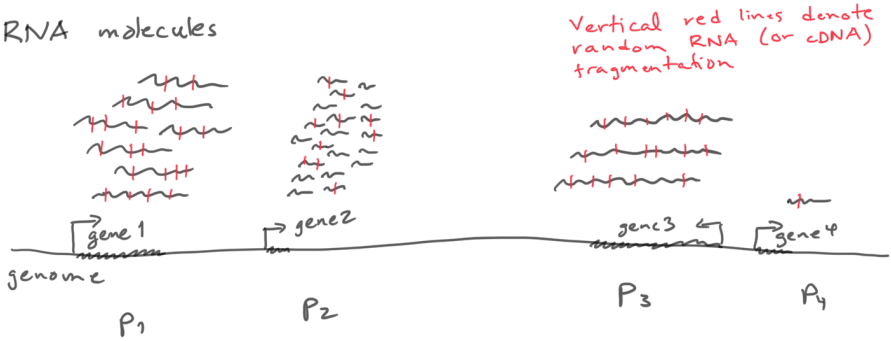
Binomial distribution

- Consider a binary-valued random variable that takes value 1 with probability p and value 0 with probability $1 - p$
- For example, the probability that we obtain a sequencing read from gene i is $p = p_i$, and the probability that we obtain a sequencing read from any other gene is $1 - p = \sum_{j \neq i} p_j$
- Take n independent random realizations of the binary-valued random variable
- Let X denote the number of success in n realizations
- The probability of getting exactly $X = k$ successes in n trials is given by probability mass function of the binomial distribution

$$\mathrm{B}(k; n, p) = P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k}$$

- Each of the components of a multinomial distribution separately (e.g. a gene) has a binomial distribution

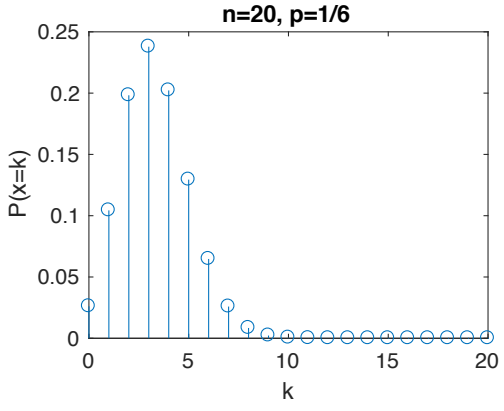
Multinomial vs. binomial distribution for RNA-seq



RNA molecules' relative proportions, where

$$\sum_{i=1}^{4} p_i = 1$$

Binomial distribution



Poisson distribution

- Consider a discrete random variable X that can have values $0, 1, 2, \ldots$ up to n, where n is very large (practically infinite)
- The discrete random variable X has a Poisson distribution with rate parameter $\lambda > 0$ if

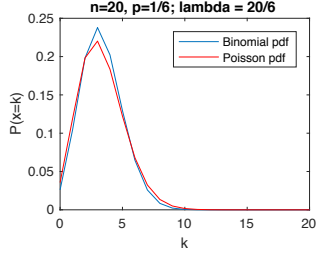
$$\text{Poisson}(k; \lambda) = P(X = k) = \frac{\lambda^k \exp(-\lambda)}{k!}$$

Poisson distribution

- Consider a discrete random variable X that can have values $0, 1, 2, \ldots$ up to n, where n is very large (practically infinite)
- The discrete random variable X has a Poisson distribution with rate parameter $\lambda > 0$ if

$$\text{Poisson}(k; \lambda) = P(X = k) = \frac{\lambda^k \exp(-\lambda)}{k!}$$

- For large number of trials n and with a small probability p (of fixed value of $n \cdot p$), binomial distribution $B(X; n, p)$ can be approximated by Poisson distribution $\text{Poisson}(X; \lambda)$ where $\lambda = n \cdot p$



Poisson approximation for Binomial distribution

- We have ($p = \frac{\lambda}{n}$)

$$
\begin{aligned}
\lim_{n\to\infty} B(X = k; n, p) &= \lim_{n\to\infty} \frac{n!}{k!(n-k)!} \left(\frac{\lambda}{n}\right)^k \left(1 - \frac{\lambda}{n}\right)^{n-k} \\
&= \left(\frac{\lambda^k}{k!}\right) \lim_{n\to\infty} \frac{n!}{(n-k)!} \left(\frac{1}{n^k}\right) \left(1 - \frac{\lambda}{n}\right)^n \left(1 - \frac{\lambda}{n}\right)^{-k} \\
&= \left(\frac{\lambda^k}{k!}\right) \lim_{n\to\infty} \frac{n(n-1)\cdots(n-k+1)}{n^k} \left(1 - \frac{\lambda}{n}\right)^n \left(1 - \frac{\lambda}{n}\right)^{-k} \\
&= \left(\frac{\lambda^k}{k!}\right) \lim_{n\to\infty} \underbrace{\left(\frac{n^k + O(n^{k-1})}{n^k}\right)}_{\to 1} \underbrace{\left(1 - \frac{\lambda}{n}\right)^n}_{e^{-\lambda}\dagger} \underbrace{\left(1 - \frac{\lambda}{n}\right)^{-k}}_{\to 1} \\
&= \frac{\lambda^k}{k!} e^{-\lambda}
\end{aligned}
$$

†Because $\lim_{x\to\infty} \left(1 + \frac{1}{x}\right)^x = e$

Poisson distribution

- For RNA-seq data
  - The number of sequencing reads (n) in an experiment is large
  - The relative abundance (p) of a single gene among all e.g. 20,000 human genes is small
- So Poisson model for sequencing read counts for a single gene in a single experiment is a reasonable approximation

Negative binomial distribution

- Read counts across biological replicates is observed to have a larger variance than what Poisson model suggests
  - So-called overdispersed noise
  - Biological variability/noise
- Negative binomial has been found to provide a good fit to sequencing count data

Negative binomial distribution

- Read counts across biological replicates is observed to have a larger variance than what Poisson model suggests
  - So-called overdispersed noise
  - Biological variability/noise
- Negative binomial has been found to provide a good fit to sequencing count data
- The negative binomial distribution is a discrete probability distribution for the following counting process:
  - Start a sequence of i.i.d. Bernoulli trials (with probability p)
  - Count the number of successes (denoted X) in your sequence until a specified (non-random) number of failures (denoted r) occurs

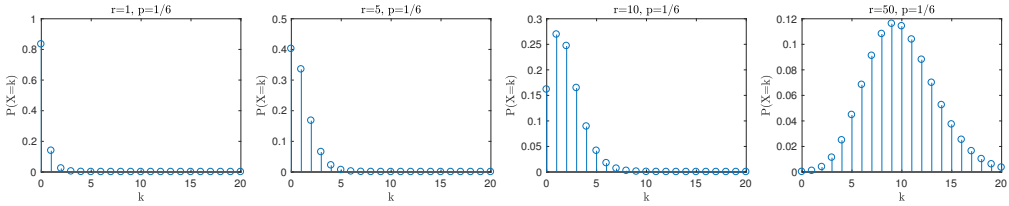
Negative binomial distribution

- Read counts across biological replicates is observed to have a larger variance than what Poisson model suggests
  - So-called overdispersed noise
  - Biological variability/noise
- Negative binomial has been found to provide a good fit to sequencing count data
- The negative binomial distribution is a discrete probability distribution for the following counting process:
  - Start a sequence of i.i.d. Bernoulli trials (with probability p)
  - Count the number of successes (denoted X) in your sequence until a specified (non-random) number of failures (denoted r) occurs
- Random variable X has the negative binomial distribution with probability mass function

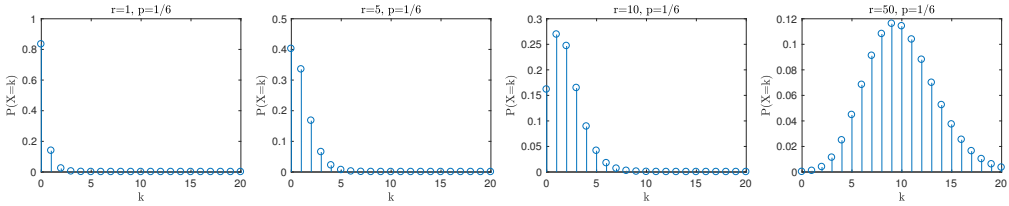
$$\mathrm{NB}(k; r, p) = P(X = k) = \binom{r + k - 1}{k} p^k (1 - p)^r$$

- The negative binomial distribution has several alternative formulations: see e.g. `https://en.wikipedia.org/wiki/Negative_binomial_distribution`
- Be careful, especially when using in different programming languages!

Negative binomial distribution



Negative binomial distribution



- Negative binomial distribution occurs in many contexts
- Negative binomial distribution can also be derived as a continuous mixture of Poisson distributions where the mixing distribution is a gamma distribution

$$\mathrm{NB}(k; r, p) = \int_0^\infty \mathrm{Poisson}(k; \lambda)\mathrm{Gamma}\left(\lambda; r, \frac{1-p}{p}\right) d\lambda$$

Gamma-Poisson compound distributions

$$
\begin{aligned}
f(k;r,p) &= \int_0^\infty f_{\text{Poisson}(\lambda)}(k) \cdot f_{\text{Gamma}\left(r,\,\frac{1-p}{p}\right)}(\lambda)\,\mathrm{d}\lambda \\
&= \int_0^\infty \frac{\lambda^k}{k!} e^{-\lambda} \cdot \lambda^{r-1} \frac{e^{-\lambda(1-p)/p}}{\left(\frac{p}{1-p}\right)^r \Gamma(r)}\,\mathrm{d}\lambda \\
&= \frac{(1-p)^r p^{-r}}{k!\,\Gamma(r)} \int_0^\infty \lambda^{r+k-1} e^{-\lambda/p}\,\mathrm{d}\lambda \\
&= \frac{(1-p)^r p^{-r}}{k!\,\Gamma(r)}\, p^{r+k}\,\Gamma(r+k) \\
&= \frac{\Gamma(r+k)}{k!\,\Gamma(r)}\, p^k (1-p)^r.
\end{aligned}
$$

Copy-pasted from wikipedia: `https://en.wikipedia.org/wiki/Negative_binomial_distribution`

Compound distributions

- Assume a random variable X with a cumulative distribution F_f (and density p_f) with parameters θ
- Assume that the parameters θ of F_f are not fixed but have a mixing distribution F_g (density p_g)
- Distribution F_f is compounded by F_g

$$p(x) = \int p_f(x|\theta) p_g(\theta) d\theta$$

- Recall the definition of the joint and marginal distributions

$$p(x, y) = p(x|y)p(y) \text{ and } p(x) = \int p(x, y)dy = \int p(x|y)p(y)dy$$

Compound distributions

Typical usage:

- Overdispersion modeling
  - Need to model a greater amount of variability than what would be expected by a given baseline model
- Bayesian inference
  - Predictive distribution of future data $p(y^*|\theta)$ given the posterior distribution of model parameters θ conditioned on observed data y, $p(y^*|y) = \int p(y^*|\theta)p(\theta|y)d\theta$

Commonly used compound distributions in bioinformatics

- Gamma-Poisson, i.e., negative binomial
- Beta-binomial
- Dirichlet-multinomial

Negative binomial distribution: reparametrizations

- The mean and variance of negative binomially distributed random variable are

$$\mathbb{E}[X] = \mu = \frac{pr}{1-p} \quad \text{and} \quad \mathbb{V}[X] = \sigma^2 = \frac{pr}{(1-p)^2}$$

Negative binomial distribution: reparametrizations

- The mean and variance of negative binomially distributed random variable are

$$\mathbb{E}[X] = \mu = \frac{pr}{1-p} \quad \text{and} \quad \mathbb{V}[X] = \sigma^2 = \frac{pr}{(1-p)^2}$$

- For our application it is useful to reparameterized NB using the mean and variance

$$\mathrm{NB}(\mu, \sigma^2) \triangleq \mathrm{NB}(r, p),$$

where

$$r = \frac{\mu^2}{\sigma^2 - \mu} \quad \text{and} \quad p = \frac{\sigma^2 - \mu}{\sigma^2}$$

Negative binomial distribution: reparametrizations

- The mean and variance of negative binomially distributed random variable are

$$\mathbb{E}[X] = \mu = \frac{pr}{1-p} \quad \text{and} \quad \mathbb{V}[X] = \sigma^2 = \frac{pr}{(1-p)^2}$$

- For our application it is useful to reparameterized NB using the mean and variance

$$\mathrm{NB}(\mu, \sigma^2) \triangleq \mathrm{NB}(r, p),$$

where

$$r = \frac{\mu^2}{\sigma^2 - \mu} \quad \text{and} \quad p = \frac{\sigma^2 - \mu}{\sigma^2}$$

- Further, we will consider a parameterization using the mean μ and dispersion ϕ

$$\mathrm{NB}(\mu, \phi) \triangleq \mathrm{NB}(\mu, \sigma^2),$$

where ϕ defines the variance as $\sigma^2 = \mu + \phi\mu^2$

Contents

Differential gene expression analysis

- We will look at edgeR (McCarthy et al., 2012), a versatile and efficient modeling method for sequencing count data
- edgeR model assumes that the number of aligned reads in sample j that are assigned to gene g can be modelled by negative binomial distribution (note: mean-dispersion reparametrization)

$$N_{gj} \sim \mathrm{NB}(s_j \lambda_{gj}, \phi_g),$$

where

  - s_j is the so-called library size: e.g. the total number of sequencing reads from sample j (or some other normalization quantity)
  - λ_{gj} is the proportion of RNA fragments that originate from gene g in sample j
    - Note that $\sum_g \lambda_{gj} = 1$
  - ϕ_g is the dispersion for gene g that defines the over-dispersion and thus the variance in the negative binomial model

Differential gene expression analysis

- For the our reparameterized definition of NB distribution the mean and variance for N_{gj} are

$$\mathbb{E}[N_{gj}] = \mu_{gj} = s_j \lambda_{gj} \quad (1)$$
$$\mathbb{V}[N_{gj}] = \mu_{gj} + \phi_g \mu_{gj}^2 = s_j \lambda_{gj} + \phi_g s_j^2 \lambda_{gj}^2 \quad (2)$$

- Recall that for the standard Poisson model $\mathbb{E}[N_{gj}] = \mu_{gj}$ and $\mathbb{V}[N_{gj}] = \mu_{gj}$

Differential gene expression analysis

- Often one is interested in comparing two populations A and B, i.e., $H_0 : \lambda_{gA} = \lambda_{gB}$
- edgeR implements a generalized linear model (GLM) with NB distribution that allows comparison of two population means as well as many other more complex experimental designs
- In GLM the mean $\mu_{gj} = s_j \lambda_{gj}$ of the NB is modeled with a log-linear model

$$\begin{aligned} \log \lambda_{gj} &= \mathbf{x}_j^T \boldsymbol{\beta}_g \\ \log \mu_{gj} &= \mathbf{x}_j^T \boldsymbol{\beta}_g + \log s_j \\ \log \mu_{gj} &= \beta_0 + \sum_{k=1}^{p} x_{jk} \beta_{gk} + \log s_j, \end{aligned}$$

  - $\mathbf{x}_j$ is a vector that contains all p covariates for sample j, and
  - $\boldsymbol{\beta}_g$ is a vector that contains the corresponding parameters for gene g
- The mean of the NB distribution is $\mu_{gj} = \exp(\mathbf{x}_j^T \boldsymbol{\beta}_g + \log s_j)$
- Recall that variance is defined as $\mu_{gj} + \phi \mu_{gj}^2$

Differential gene expression analysis

- Consider a simple example with 4 samples:
  - 2 from group A and 2 from group B
  - The 4 samples have "age" covariate values 0.5, 1, 1.5 and 2
- The GLM model and the design matrix X for the null hypothesis model (M_0) that assumes there is no difference between A and B

$$\begin{pmatrix} \log \mu_{g1} \\ \log \mu_{g2} \\ \log \mu_{g3} \\ \log \mu_{g4} \end{pmatrix} = \begin{pmatrix} 1 & 0.5 \\ 1 & 1.5 \\ 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} \beta_{g0} \\ \beta_{g1} \end{pmatrix} + \begin{pmatrix} \log s_1 \\ \log s_2 \\ \log s_3 \\ \log s_4 \end{pmatrix},$$

Differential gene expression analysis

- Consider a simple example with 4 samples:
  - 2 from group A and 2 from group B
  - The 4 samples have "age" covariate values 0.5, 1, 1.5 and 2
- The GLM model and the design matrix X for the null hypothesis model (M_0) that assumes there is no difference between A and B

$$\begin{pmatrix} \log \mu_{g1} \\ \log \mu_{g2} \\ \log \mu_{g3} \\ \log \mu_{g4} \end{pmatrix} = \begin{pmatrix} 1 & 0.5 \\ 1 & 1.5 \\ 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} \beta_{g0} \\ \beta_{g1} \end{pmatrix} + \begin{pmatrix} \log s_1 \\ \log s_2 \\ \log s_3 \\ \log s_4 \end{pmatrix},$$

- The model for the alternative hypothesis with two conditions (M_1) can be written e.g.

$$\begin{pmatrix} \log \mu_{g1} \\ \log \mu_{g2} \\ \log \mu_{g3} \\ \log \mu_{g4} \end{pmatrix} = \begin{pmatrix} 1 & 0.5 & 0 \\ 1 & 1.5 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 1 \end{pmatrix} \begin{pmatrix} \beta_{g0} \\ \beta_{g1} \\ \beta_{g2} \end{pmatrix} + \begin{pmatrix} \log s_1 \\ \log s_2 \\ \log s_3 \\ \log s_4 \end{pmatrix},$$

where samples 1 and 2 are from condition A and samples 3 and 4 are from condition B

Differential gene expression analysis

- Continuing the example from the previous page, lets denote the 4 observed read counts for gene g as $\mathbf{y}_g = (n_{g1}, \ldots, n_{g4})^T$
- In edgeR, statistical hypothesis testing for differential gene expression between conditions A and B can be implemented e.g. with the likelihood-ratio test

$$T = -2 \ln \frac{\ell(\hat{\beta}_{g0}, \hat{\beta}_{g1}, \hat{\phi}_g | \mathbf{y}_g, M_0)}{\ell(\hat{\beta}_{g0}, \hat{\beta}_{g1}, \hat{\beta}_{g2}, \hat{\phi}_g | \mathbf{y}_g, M_1)}$$

  - $\ell(\cdot)$ is the NB density function
  - $\hat{\beta}_{gi}$ denotes the maximum likelihood estimate of β_{gi} given $\mathbf{y}_g$ and M_0 (or $\mathbf{y}_g$ and M_1)
  - Similarly, $\hat{\phi}_g$ denotes the maximum likelihood estimate (or another estimate, see next slides) of dispersion ϕ
- The test statistic T is approximately chi-squared distributed with degrees of freedom equal to $\mathrm{df}_{M_1} - \mathrm{df}_{M_0}$, where df_M denotes the number of free parameters of model M

  $\rightarrow$ p-value
  - Remember multiple testing

Differential gene expression analysis

- In some applications the number of biological replicates is too small to allow accurate estimation of both β_{gi} and ϕ_g
  - edgeR tool implements a moderated test where information between genes is shared that allows more accurate dispersion estimation
- The so-called adjusted profile likelihood (APL) for dispersion ϕ_g is

$$APL_g(\phi_g) = \ell(\phi_g|\mathbf{y}_g, \hat{\beta}_g) - \frac{1}{2}\log\det\mathcal{I}_g$$

  - ϕ_g is free parameter
  - $\hat{\beta}_g$ is the ML estimate of β_g that depends on ϕ_g
  - $\mathcal{I}_g$ is the Fisher information matrix

Differential gene expression analysis

- One possible assumption is that all genes have the same dispersion value $\phi_g = \phi$
- A shared dispersion can be estimated by maximizing the sum of the adjusted profile likelihoods

$$APL_S(\phi) = \sum_{g=1}^{G} APL_g(\phi)$$

- In essence, data across all genes is shared to estimate variance/dispersion
- edgeR tool provides also options for other dispersion estimates
  - Trended: group genes into bin that have similar mean read count
  - Gene-wise

Differential gene expression analysis

- An example from edgeR User Guide (Chen et al, 2017)
- Three patient with oral squamous cell carcinomas
  - Oral squamous cell carcinomas and matched normal tissue from each patient
  - RNA-seq experiments paired experimental design
- Goal: detect genes differentially expressed between tumour and normal tissue
- Samples: 8N, 8T, 33N, 33T, 51N, 51T
- Design matrix X is

	(Intercept)	Patient33	Patient51	TissueT
8N	1	0	0	0
8T	1	0	0	1
33N	1	1	0	0
33T	1	1	0	1
51N	1	0	1	0
51T	1	0	1	1

Figure from (Chen et al, 2017)

Differential gene expression analysis

- Variance dependence on the mean (biological coefficient of variation equals the square root of the dispersion)

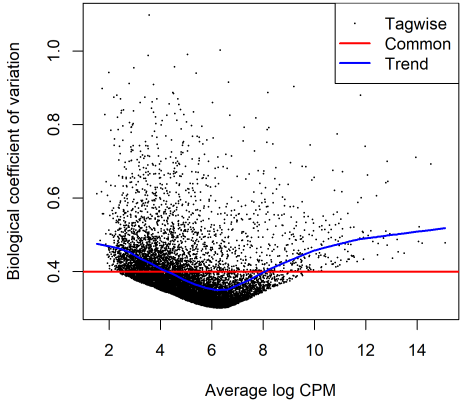


Figure from (Chen et al, 2017)

Differential gene expression analysis

- 1269 genes differentially expressed with FDR 5%
- Additionally, require at least 2-fold change (blue horizontal lines below)
- MA plot: a scatter plot where a dot corresponds to a gene g, x-axis shows mean gene expression $\frac{1}{2} \log X_{gA} X_{gB}$ and y-axis shows difference $\log \frac{X_{gA}}{X_{gB}}$

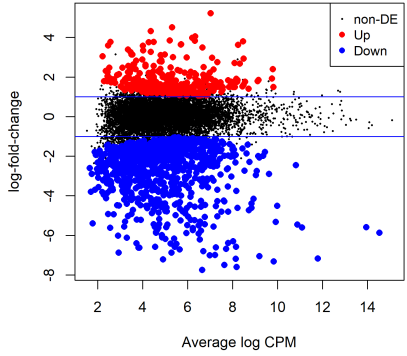


Figure from (Chen et al, 2017)

Contents

Transcript-level expression quantification

- Let us assume that each gene i is associated with J_i transcripts indexed by j, then

$$\begin{aligned}\theta_{ij} &= P(\text{sample a read from transcript } j \text{ associated with gene } i) \\ &= \frac{1}{Z}\mu_{ij}\ell_{ij},\end{aligned}$$

  where
  - μ_{ij} is the expression level of transcript j associated with gene j
  - ℓ_{ij} is the length of transcript j of gene i
  - Normalizing constant is $Z = \sum_{ij} \mu_{ij}\ell_{ij}$
- The true expression level of gene i is

$$\mu_i = \sum_{j=1}^{J_i} \mu_{ij}$$

Transcript-level expression quantification

- Lets denote the aligned RNA-seq reads as $R_1, R_2, \ldots, R_N$ (note that N now denotes the same thing as n previously)
- Let us also make an unrealistic assumption that all reads are assigned uniquely to one of the transcripts
- Then the frequency estimator gives us

$$\hat{\theta}_{ij} = \frac{k_{ij}}{N},$$

  where k_{ij} is the number of reads assigned uniquely to transcript j of gene i
- Correspondingly, we can convert the estimates into expression values by normalizing by the transcript length

$$\hat{\mu}_{ij} \propto \frac{\hat{\theta}_{ij}}{\ell_{ij}} = \frac{k_{ij}}{\ell_{ij} N} \quad \text{and} \quad \hat{\mu}_i = \sum_{j=1}^{J_i} \hat{\mu}_{ij} \propto \sum_j \frac{k_{ij}}{\ell_{ij} N}$$

Transcript-level expression quantification

- Recall the union method for estimating the gene expression level

$$k_i = \sum_j k_{ij}$$

and the frequency estimator

$$\hat{\theta}_i = \frac{k_i}{\ell_i},$$

where ℓ_i is the length of the gene i (sum of lengths of all exons)

- Union method tends to underestimate the gene expression level because

$$\begin{aligned} \hat{\theta}_i &= \frac{\sum_j k_{ij}}{\ell_i} = \frac{k_{i1}}{\ell_i} + \cdots + \frac{k_{iJ_i}}{\ell_i} \\ &\leq \frac{k_{i1}}{\ell_{i1}} + \cdots + \frac{k_{iJ_i}}{\ell_{iJ_i}}, \end{aligned}$$

where $\ell_i \geq \ell_{ij}$

Transcript-level expression quantification

- Consider a simple case of skipped exon

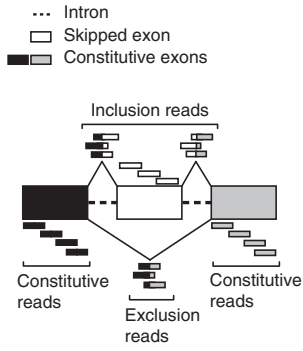


Figure from (Katz et al., 2010)

- We can use e.g. the reads in the skipped exon and the inclusion and exclusion reads together with the frequency estimator to estimate the relative expression of the two transcripts

Transcript-level expression quantification

- With paired end reads we can try to use all (non-uniquely) aligned reads assuming we can estimate insert length variability

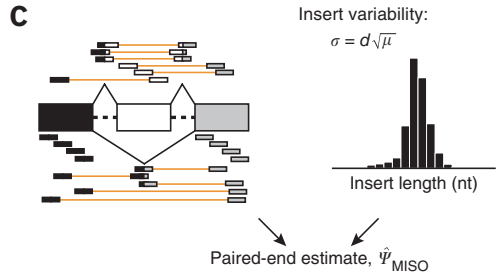


Figure from (Katz et al., 2010)

- Estimation can be done Markov chain Monte Carlo (MCMC) sampling (Katz et al., 2010)

References

- Agresti, Alan. Foundations of Linear and Generalized Linear Models, John Wiley & Sons, 2015
- Chen Y, et al., edgeR: differential expression analysis of digital gene expression data, User's Guide, 11 October 2017
- Murphy K, Machine learning: a probabilistic perspective, MIT Press, 2012
- Katz Y, et al., Analysis and design of rnA sequencing experiments for identifying isoform regulation, Nature Methods, 7(12):1009-15, 2010.