UNIVERSITEIT GENT

BIG N2N
Bioinformatics Institute Ghent
from nucleotides to networks

Differential Expression with RNA-seq: Technical Details

Lieven Clement

Ghent University, Belgium

Statistical Genomics: Master of Science in Bioinformatics

TWIST, Krijgslaan 281 (S9), Gent, Belgium e-mail: lieven.clement@ugent.be

Exponential family

$$f(y_i|\theta_i,\phi) = \exp\left\{\frac{y_i\theta_i - b(\theta_i)}{a(\phi)} + c(y_i,\phi)\right\}$$

with

- θ_i: canonical parameters
- ϕ: dispersion parameter
- $a(.)$, $b(.)$, $c(.)$: specific functions that depend on the distribution, e.g. for normal distribution $\phi = \sigma^2$, $\theta = \mu$, $a(\phi) = \phi = \sigma^2$, $b(\theta_i) = \theta_i^2/2$, $c(y_i,\phi) = -\frac{1}{2}[y^2/\phi + \log(2\pi\phi)]$

Components of Generalized Linear Model

$$\begin{cases} y_i|x_i & \sim & f(y_i|\theta_i,\phi) \\ \mathrm{E}\left[y_i|\mathbf{x}_i\right] & = & \mu_i \\ g(\mu_i) & = & \eta(\mathbf{x}_i) \\ \eta(\mathbf{x}_i) & = & \mathbf{x}_i^T\boldsymbol{\beta} \end{cases},$$

with $g(.)$ the link function, e.g.

- $g(.) = .$: identity link for Normal distribution
- $g(.) = \log(.)$: canonical link for Poisson distribution
- $g(.) = \mathrm{logit}(.) = \log\left[\frac{(.)}{(1-.)}\right]$: canonical link for Bernouilli distribution.

Parameter estimation: the likelihood

- We start from a sample, and consider it as fixed and known.
- In particular we do NOT consider the sample observations as random variables.
- Therefore we write the observed sample as $y_i, ..., y_n$
- The theory is based on the likelihood function, which can be interpreted as a measure for the probability that the given sample is observed as a realisation of a sequence of random variables $Y_1, \ldots Y_n$.
- The random variables Y_i are characterised by a distribution or density function which has typically unknown parameters, e.g. a Poisson distribution $f(Y_i) \sim \text{Poisson}(\theta_i)$.

Parameter estimation: the likelihood

- When the subjects are mutually independent the joint likelihood to observe $y_1, \ldots, y_n$ equals

$$\prod_{i=1}^{n} f(y_i, \theta_i, \phi)$$

- The densities are actually also a function of the parameters θ_i, ϕ. To stress this, we indicated that in the density formulation.
- The likelihood function is a function of all parameters

$$L(\boldsymbol{\theta}, \phi|\mathbf{y}) = \prod_{i=1}^{n} f(y_i, \theta_i, \phi)$$

- The log-likelihood function is often used, which is defined as

$$l(\boldsymbol{\theta}, \phi|\mathbf{y}) = \log L(\boldsymbol{\theta}, \phi|\mathbf{y}) = \sum_{i=1}^{n} \log f(y_i, \theta_i, \phi)$$

log-likelihood

$$l(\theta_i,\phi|y_i)=\left\{\frac{y_i\theta_i-b(\theta_i)}{a(\phi)}+c(y_i,\phi)\right\}$$

- $E[y_i]=\mu_i=b'(\theta_i)$

- $\text{var}[y_i]=b''(\theta_i)a(\phi)$

Variance $\text{var}[y_i]$ depends on mean! Often there is no dispersion parameter e.g. Bernouilli: $\text{var}[y_i]=\mu_i(1-\mu_i)$, Poisson $\text{var}[y_i]=\mu_i$.

Maximum likelihood, Score function

$$S_i(\theta_i) = \frac{\partial l(\theta_i, \phi | y_i)}{\partial \theta_i}$$

when canonical link function is used:

- $\mu_i = b'(\theta_i)$

Maximum likelihood, Score function

$$S_i(\theta_i) = \frac{\partial l(\theta_i, \phi | y_i)}{\partial \theta_i} = \frac{y_i - \mu_i}{a(\phi)}$$

when canonical link function is used:

- $\mu_i = b'(\theta_i)$

Maximum likelihood, Score function

$$S_i(\theta_i) = \frac{\partial l(\theta_i, \phi|y_i)}{\partial \theta_i} = \frac{y_i - \mu_i}{a(\phi)}$$

when canonical link function is used:

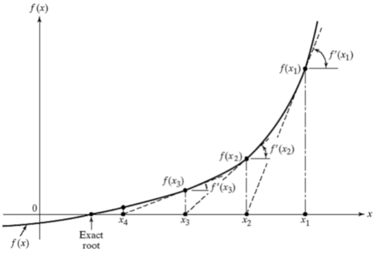
- $\mu_i = b'(\theta_i)$
- Regression (chain rule and $i = 1, \ldots, n$ i.i.d observations)

$$S_i(\boldsymbol{\beta}) = \frac{\partial l(\theta_i, \phi|; y_i)}{\partial \theta_i} \frac{\partial \theta_i}{\partial \mu_i} \frac{\partial \mu_i}{\partial \eta_i} \frac{\partial \eta_i}{\partial \boldsymbol{\beta}} = \mathbf{x}_i S_i(\theta_i) \frac{1}{b''(\theta_i)} \frac{\partial \mu_i}{\partial \eta_i}$$

$$S(\boldsymbol{\beta}) = \sum_{i=1}^{n} \mathbf{x}_i \frac{y_i - \mu_i}{a(\phi) b''(\theta_i)} \frac{\partial \mu_i}{\partial \eta_i} = \mathbf{X}^T \mathbf{A} (\mathbf{y} - \boldsymbol{\mu})$$

- **A** is a diagonal matrix: $\mathbf{A} = \text{diag}\left[\text{var}[y_i] \frac{\partial \eta_i}{\partial \mu_i}\right]^{-1}$,
  $\mathbf{y} = [y_1, \ldots, y_n]^T$, $\boldsymbol{\mu} = [\mu_1, \ldots, \mu_n]^T$
- Optimization??

Newton Raphson



$$\hat{\boldsymbol{\beta}} : S(\boldsymbol{\beta}) = 0$$

$$\boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^k - \left(\left.\frac{\partial S(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}\right|_{\boldsymbol{\beta}^k} \right)^{-1} S(\boldsymbol{\beta}^k)$$

$$\boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^k + J^{-1}(\boldsymbol{\beta}^k)\Big|_{\boldsymbol{\beta}^k} S(\boldsymbol{\beta}^k)$$

with $J(\boldsymbol{\beta}^k)$ the observed Fisher information matrix.

- Fisher scoring: replace observed Fisher information matrix $J(\boldsymbol{\beta}^k)$ by expected Fisher information matrix $I(\boldsymbol{\beta}^k) = \text{E}[J(\boldsymbol{\beta}^k)]$.
- If you use canonical link, $I(\boldsymbol{\beta}^k) = J(\boldsymbol{\beta}^k)$ → Fisher Scoring and Newton Raphson are identical.

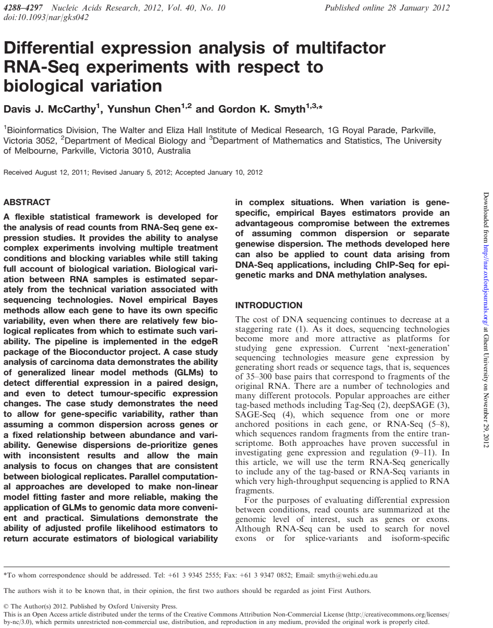
Iteratively Reweighted Least Squares (IRLS)

Newton Raphson and Fisher Scoring can be recasted in an IRLS algorithm

$$\begin{aligned}
\boldsymbol{\beta}^{k+1} &= \boldsymbol{\beta}^k + I^{-1}(\boldsymbol{\beta}^k)\Big|_{\boldsymbol{\beta}^k} S(\boldsymbol{\beta}^k) \\
&= \boldsymbol{\beta}^k + (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{A}(\mathbf{y}-\boldsymbol{\mu}) \\
&= \boldsymbol{\beta}^k + (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\tfrac{\partial\boldsymbol{\eta}}{\partial\boldsymbol{\mu}}(\mathbf{y}-\boldsymbol{\mu}) \qquad , \\
&= (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\left[\mathbf{X}\beta^k + \tfrac{\partial\boldsymbol{\eta}}{\partial\boldsymbol{\mu}}(\mathbf{y}-\boldsymbol{\mu})\right] \\
&= (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\mathbf{z}
\end{aligned}$$

with $I(\beta) = \mathbf{X}^T\mathbf{W}\mathbf{X}$, $\mathbf{W} = \mathbf{A}\text{diag}\left[\frac{\partial\eta}{\partial\mu}\right]^{-1}$ and pseudo data $\mathbf{z} = \boldsymbol{\eta} + \frac{\partial\boldsymbol{\eta}}{\partial\boldsymbol{\mu}}(\mathbf{y}-\boldsymbol{\mu})$

Defining model



4288–4297 *Nucleic Acids Research, 2012, Vol. 40, No. 10* *Published online 28 January 2012*
doi:10.1093/nar/gks042

Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation

Davis J. McCarthy[1], Yunshun Chen[1,2] and Gordon K. Smyth[1,3,*]

[1]Bioinformatics Division, The Walter and Eliza Hall Institute of Medical Research, 1G Royal Parade, Parkville, Victoria 3052, [2]Department of Medical Biology and [3]Department of Mathematics and Statistics, The University of Melbourne, Parkville, Victoria 3010, Australia

Received August 12, 2011; Revised January 5, 2012; Accepted January 10, 2012

ABSTRACT

A flexible statistical framework is developed for the analysis of read counts from RNA-Seq gene expression studies. It provides the ability to analyse complex experiments involving multiple treatment conditions and blocking variables while still taking full account of biological variation. Biological variation between RNA samples is estimated separately from the technical variation associated with sequencing technologies. Novel empirical Bayes methods allow each gene to have its own specific variability, even when there are relatively few biological replicates from which to estimate such variability. The pipeline is implemented in the edgeR package of the Bioconductor project. A case study analysis of carcinoma data demonstrates the ability of generalized linear model methods (GLMs) to detect differential expression in a paired design, and even to detect tumour-specific expression changes. The case study demonstrates the need to allow for gene-specific variability, rather than assuming a common dispersion across genes or a fixed relationship between abundance and variability. Genewise dispersions de-prioritize genes with inconsistent results and allow the main analysis to focus on changes that are consistent between biological replicates. Parallel computational approaches are developed to make non-linear model fitting faster and more reliable, making the application of GLMs to genomic data more convenient and practical. Simulations demonstrate the ability of adjusted profile likelihood estimators to return accurate estimators of biological variability in complex situations. When variation is gene-specific, empirical Bayes estimators provide an advantageous compromise between the extremes of assuming common dispersion or separate genewise dispersion. The methods developed here can also be applied to count data arising from DNA-Seq applications, including ChIP-Seq for epigenetic marks and DNA methylation analyses.

INTRODUCTION

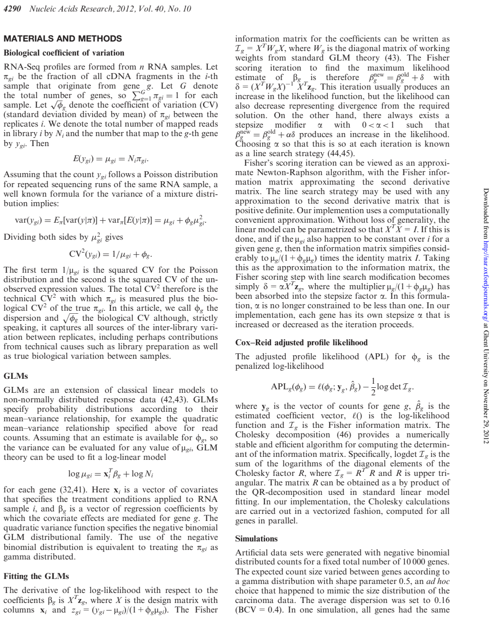
The cost of DNA sequencing continues to decrease at a staggering rate (1). As it does, sequencing technologies become more and more attractive as platforms for studying gene expression. Current 'next-generation' sequencing technologies measure gene expression by generating short reads or sequence tags, that is, sequences of 35–300 base pairs that correspond to fragments of the original RNA. There are a number of technologies and many different protocols. Popular approaches are either tag-based methods including Tag-Seq (2), deepSAGE (3), SAGE-Seq (4), which sequence from one or more anchored positions in each gene, or RNA-Seq (5–8), which sequences random fragments from the entire transcriptome. Both approaches have proven successful in investigating gene expression and regulation (9–11). In this article, we will use the term RNA-Seq generically to include any of the tag-based or RNA-Seq variants in which very high-throughput sequencing is applied to RNA fragments.

For the purposes of evaluating differential expression between conditions, read counts are summarized at the genomic level of interest, such as genes or exons. Although RNA-Seq can be used to search for novel exons or for splice-variants and isoform-specific

[*] To whom correspondence should be addressed. Tel: +61 3 9345 2555; Fax: +61 3 9347 0852; Email: smyth@wehi.edu.au

The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors.

NegBinom GLM for RNA-seq



MATERIALS AND METHODS

Biological coefficient of variation

RNA-Seq profiles are formed from n RNA samples. Let π_{gi} be the fraction of all cDNA fragments in the i-th sample that originate from gene g. Let G denote the total number of genes, so $\sum_{g=1}^{G} \pi_{gi} = 1$ for each sample. Let $\sqrt{\phi_g}$ denote the coefficient of variation (CV) (standard deviation divided by mean) of π_{gi} between the replicates i. We denote the total number of mapped reads in library i by N_i and the number that map to the g-th gene by y_{gi}. Then

$$E(y_{gi}) = \mu_{gi} = N_i \pi_{gi}.$$

Assuming that the count y_{gi} follows a Poisson distribution for repeated sequencing runs of the same RNA sample, a well known formula for the variance of a mixture distribution implies:

$$\mathrm{var}(y_{gi}) = E_\pi[\mathrm{var}(y|\pi)] + \mathrm{var}_\pi[E(y|\pi)] = \mu_{gi} + \phi_g \mu_{gi}^2.$$

Dividing both sides by μ_{gi}^2 gives

$$\mathrm{CV}^2(y_{gi}) = 1/\mu_{gi} + \phi_g.$$

The first term $1/\mu_{gi}$ is the squared CV for the Poisson distribution and the second is the squared CV of the unobserved expression values. The total CV^2 therefore is the technical CV^2 with which π_{gi} is measured plus the biological CV^2 of the true π_{gi}. In this article, we call ϕ_g the dispersion and $\sqrt{\phi_g}$ the biological CV although, strictly speaking, it captures all sources of the inter-library variation between replicates, including perhaps contributions from technical causes such as library preparation as well as true biological variation between samples.

GLMs

GLMs are an extension of classical linear models to non-normally distributed response data (42,43). GLMs specify probability distributions according to their mean–variance relationship, for example the quadratic mean–variance relationship specified above for read counts. Assuming that an estimate is available for ϕ_g, so the variance can be evaluated for any value of μ_{gi}, GLM theory can be used to fit a log-linear model

$$\log \mu_{gi} = \mathbf{x}_i^T \beta_g + \log N_i$$

for each gene (32,41). Here $\mathbf{x}_i$ is a vector of covariates that specifies the treatment conditions applied to RNA sample i, and β_g is a vector of regression coefficients by which the covariate effects are mediated for gene g. The quadratic variance function specifies the negative binomial GLM distributional family. The use of the negative binomial distribution is equivalent to treating the π_{gi} as gamma distributed.

Fitting the GLMs

The derivative of the log-likelihood with respect to the coefficients β_g is $X^T \mathbf{z}_g$, where X is the design matrix with columns $\mathbf{x}_i$ and $z_{gi} = (y_{gi} - \mu_{gi})/(1 + \phi_g \mu_{gi})$. The Fisher information matrix for the coefficients can be written as $\mathcal{I}_g = X^T W_g X$, where W_g is the diagonal matrix of working weights from standard GLM theory (43). The Fisher scoring iteration to find the maximum likelihood estimate of β_g is therefore $\beta_g^{\mathrm{new}} = \beta_g^{\mathrm{old}} + \delta$ with $\delta = (X^T W_g X)^{-1} X^T \mathbf{z}_g$. This iteration usually produces an increase in the likelihood function, but the likelihood can also decrease representing divergence from the required solution. On the other hand, there always exists a stepsize modifier α with $0 < \alpha < 1$ such that $\beta_g^{\mathrm{new}} = \beta_g^{\mathrm{old}} + \alpha\delta$ produces an increase in the likelihood. Choosing α so that this is so at each iteration is known as a line search strategy (44,45).

Fisher's scoring iteration can be viewed as an approximate Newton-Raphson algorithm, with the Fisher information matrix approximating the second derivative matrix. The line search strategy may be used with any approximation to the second derivative matrix that is positive definite. Our implemention uses a computationally convenient approximation. Without loss of generality, the linear model can be parametrized so that $X^T X = I$. If this is done, and if the μ_{gi} also happen to be constant over i for a given gene g, then the information matrix simpifies considerably to $\mu_g/(1 + \phi_g \mu_g)$ times the identity matrix I. Taking this as the approximation to the information matrix, the Fisher scoring step with line search modification becomes simply $\delta = \alpha X^T \mathbf{z}_g$, where the multiplier $\mu_g/(1 + \phi_g \mu_g)$ has been absorbed into the stepsize factor α. In this formulation, α is no longer constrained to be less than one. In our implementation, each gene has its own stepsize α that is increased or decreased as the iteration proceeds.

Cox–Reid adjusted profile likelihood

The adjusted profile likelihood (APL) for ϕ_g is the penalized log-likelihood

$$\mathrm{APL}_g(\phi_g) = \ell(\phi_g; \mathbf{y}_g, \hat{\beta}_g) - \frac{1}{2} \log \det \mathcal{I}_g.$$

where $\mathbf{y}_g$ is the vector of counts for gene g, $\hat{\beta}_g$ is the estimated coefficient vector, $\ell()$ is the log-likelihood function and $\mathcal{I}_g$ is the Fisher information matrix. The Cholesky decomposition (46) provides a numerically stable and efficient algorithm for computing the determinant of the information matrix. Specifically, $\log\det \mathcal{I}_g$ is the sum of the logarithms of the diagonal elements of the Cholesky factor R, where $\mathcal{I}_g = R^T R$ and R is upper triangular. The matrix R can be obtained as a by product of the QR-decomposition used in standard linear model fitting. In our implementation, the Cholesky calculations are carried out in a vectorized fashion, computed for all genes in parallel.

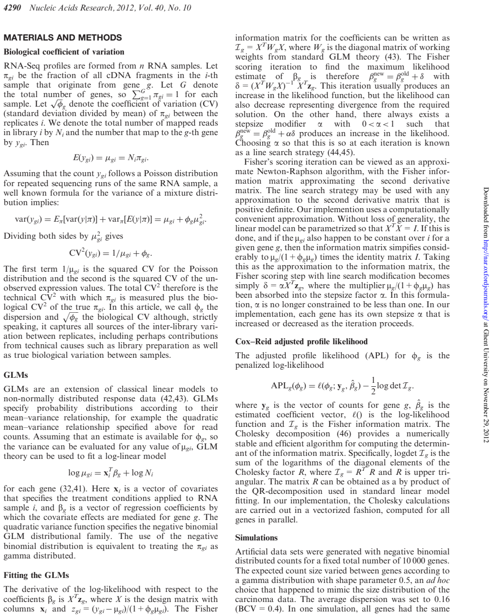
Simulations

Artificial data sets were generated with negative binomial distributed counts for a fixed total number of 10 000 genes. The expected count size varied between genes according to a gamma distribution with shape parameter 0.5, an *ad hoc* choice that happened to mimic the size distribution of the carcinoma data. The average dispersion was set to 0.16 (BCV = 0.4). In one simulation, all genes had the same

Defining model

- $b''(\theta) = \mu$
- $a(\phi) = 1 + \phi\mu$

NegBinom GLM for RNA-seq



MATERIALS AND METHODS

Biological coefficient of variation

RNA-Seq profiles are formed from n RNA samples. Let π_{gi} be the fraction of all cDNA fragments in the i-th sample that originate from gene g. Let G denote the total number of genes, so $\sum_{g=1}^{G} \pi_{gi} = 1$ for each sample. Let $\sqrt{\phi_g}$ denote the coefficient of variation (CV) (standard deviation divided by mean) of π_{gi} between the replicates i. We denote the total number of mapped reads in library i by N_i and the number that map to the g-th gene by y_{gi}. Then

$$E(y_{gi}) = \mu_{gi} = N_i \pi_{gi}.$$

Assuming that the count y_{gi} follows a Poisson distribution for repeated sequencing runs of the same RNA sample, a well known formula for the variance of a mixture distribution implies:

$$\text{var}(y_{gi}) = E_\pi[\text{var}(y|\pi)] + \text{var}_\pi[E(y|\pi)] = \mu_{gi} + \phi_g \mu_{gi}^2.$$

Dividing both sides by μ_{gi}^2 gives

$$\text{CV}^2(y_{gi}) = 1/\mu_{gi} + \phi_g.$$

The first term $1/\mu_{gi}$ is the squared CV for the Poisson distribution and the second is the squared CV of the unobserved expression values. The total CV^2 therefore is the technical CV^2 with which π_{gi} is measured plus the biological CV^2 of the true π_{gi}. In this article, we call ϕ_g the dispersion and $\sqrt{\phi_g}$ the biological CV although, strictly speaking, it captures all sources of the inter-library variation between replicates, including perhaps contributions from technical causes such as library preparation as well as true biological variation between samples.

GLMs

GLMs are an extension of classical linear models to non-normally distributed response data (42,43). GLMs specify probability distributions according to their mean–variance relationship, for example the quadratic mean–variance relationship specified above for read counts. Assuming that an estimate is available for ϕ_g, so the variance can be evaluated for any value of μ_{gi}, GLM theory can be used to fit a log-linear model

$$\log \mu_{gi} = \mathbf{x}_i^T \beta_g + \log N_i$$

for each gene (32,41). Here $\mathbf{x}_i$ is a vector of covariates that specifies the treatment conditions applied to RNA sample i, and β_g is a vector of regression coefficients by which the covariate effects are mediated for gene g. The quadratic variance function specifies the negative binomial GLM distributional family. The use of the negative binomial distribution is equivalent to treating the π_{gi} as gamma distributed.

Fitting the GLMs

The derivative of the log-likelihood with respect to the coefficients β_g is $X^T\mathbf{z}_g$, where X is the design matrix with columns $\mathbf{x}_i$ and $z_{gi} = (y_{gi} - \mu_{gi})/(1 + \phi_g \mu_{gi})$. The Fisher information matrix for the coefficients can be written as $\mathcal{I}_g = X^T W_g X$, where W_g is the diagonal matrix of working weights from standard GLM theory (43). The Fisher scoring iteration to find the maximum likelihood estimate of β_g is therefore $\beta_g^{\text{new}} = \beta_g^{\text{old}} + \delta$ with $\delta = (X^T W_g X)^{-1} X^T \mathbf{z}_g$. This iteration usually produces an increase in the likelihood function, but the likelihood can also decrease representing divergence from the required solution. On the other hand, there always exists a stepsize modifier α with $0 < \alpha < 1$ such that $\beta_g^{\text{new}} = \beta_g^{\text{old}} + \alpha\delta$ produces an increase in the likelihood. Choosing α so that this is so at each iteration is known as a line search strategy (44,45).

Fisher's scoring iteration can be viewed as an approximate Newton-Raphson algorithm, with the Fisher information matrix approximating the second derivative matrix. The line search strategy may be used with any approximation to the second derivative matrix that is positive definite. Our implemention uses a computationally convenient approximation. Without loss of generality, the linear model can be parametrized so that $X^T X = I$. If this is done, and if the μ_{gi} also happen to be constant over i for a given gene g, then the information matrix simpifies considerably to $\mu_g/(1 + \phi_g \mu_g)$ times the identity matrix I. Taking this as the approximation to the information matrix, the Fisher scoring step with line search modification becomes simply $\delta = \alpha X^T \mathbf{z}_g$, where the multiplier $\mu_g/(1 + \phi_g \mu_g)$ has been absorbed into the stepsize factor α. In this formulation, α is no longer constrained to be less than one. In our implementation, each gene has its own stepsize α that is increased or decreased as the iteration proceeds.

Cox–Reid adjusted profile likelihood

The adjusted profile likelihood (APL) for ϕ_g is the penalized log-likelihood

$$\text{APL}_g(\phi_g) = \ell(\phi_g; \mathbf{y}_g, \hat{\beta}_g) - \frac{1}{2} \log \det \mathcal{I}_g.$$

where $\mathbf{y}_g$ is the vector of counts for gene g, $\hat{\beta}_g$ is the estimated coefficient vector, $\ell()$ is the log-likelihood function and $\mathcal{I}_g$ is the Fisher information matrix. The Cholesky decomposition (46) provides a numerically stable and efficient algorithm for computing the determinant of the information matrix. Specifically, logdet $\mathcal{I}_g$ is the sum of the logarithms of the diagonal elements of the Cholesky factor R, where $\mathcal{I}_g = R^T R$ and R is upper triangular. The matrix R can be obtained as a by product of the QR-decomposition used in standard linear model fitting. In our implementation, the Cholesky calculations are carried out in a vectorized fashion, computed for all genes in parallel.

Simulations

Artificial data sets were generated with negative binomial distributed counts for a fixed total number of 10 000 genes. The expected count size varied between genes according to a gamma distribution with shape parameter 0.5, an *ad hoc* choice that happened to mimic the size distribution of the carcinoma data. The average dispersion was set to 0.16 (BCV = 0.4). In one simulation, all genes had the same

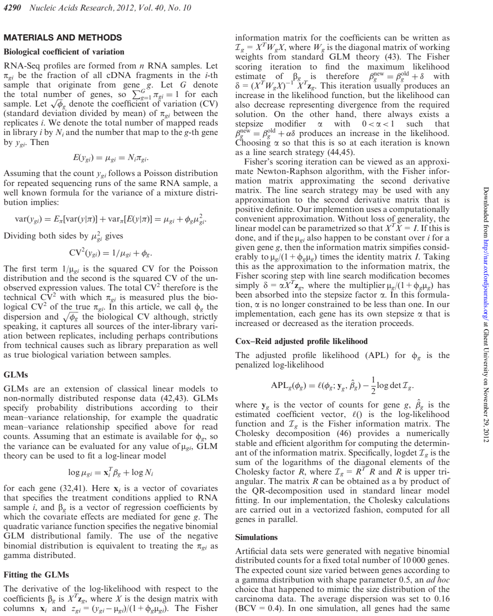
Defining model

- $b''(\theta) = \mu$
- $a(\phi) = 1 + \phi\mu$

Estimation dispersion: profiling

- $I_g = -\text{E}\left[\frac{\partial^2 L}{\partial^2 \boldsymbol{\beta}}\right]$

NegBinom GLM for RNA-seq



MATERIALS AND METHODS

Biological coefficient of variation

RNA-Seq profiles are formed from n RNA samples. Let π_{gi} be the fraction of all cDNA fragments in the i-th sample that originate from gene g. Let G denote the total number of genes, so $\sum_{g=1}^{G} \pi_{gi} = 1$ for each sample. Let $\sqrt{\phi_g}$ denote the coefficient of variation (CV) (standard deviation divided by mean) of π_{gi} between the replicates i. We denote the total number of mapped reads in library i by N_i and the number that map to the g-th gene by y_{gi}. Then

$$E(y_{gi}) = \mu_{gi} = N_i \pi_{gi}.$$

Assuming that the count y_{gi} follows a Poisson distribution for repeated sequencing runs of the same RNA sample, a well known formula for the variance of a mixture distribution implies:

$$\mathrm{var}(y_{gi}) = E_\pi[\mathrm{var}(y|\pi)] + \mathrm{var}_\pi[E(y|\pi)] = \mu_{gi} + \phi_g \mu_{gi}^2.$$

Dividing both sides by μ_{gi}^2 gives

$$\mathrm{CV}^2(y_{gi}) = 1/\mu_{gi} + \phi_g.$$

The first term $1/\mu_{gi}$ is the squared CV for the Poisson distribution and the second is the squared CV of the unobserved expression values. The total CV^2 therefore is the technical CV^2 with which π_{gi} is measured plus the biological CV^2 of the true π_{gi}. In this article, we call ϕ_g the dispersion and $\sqrt{\phi_g}$ the biological CV although, strictly speaking, it captures all sources of the inter-library variation between replicates, including perhaps contributions from technical causes such as library preparation as well as true biological variation between samples.

GLMs

GLMs are an extension of classical linear models to non-normally distributed response data (42,43). GLMs specify probability distributions according to their mean–variance relationship, for example the quadratic mean–variance relationship specified above for read counts. Assuming that an estimate is available for ϕ_g, so the variance can be evaluated for any value of μ_{gi}, GLM theory can be used to fit a log-linear model

$$\log \mu_{gi} = \mathbf{x}_i^T \beta_g + \log N_i$$

for each gene (32,41). Here $\mathbf{x}_i$ is a vector of covariates that specifies the treatment conditions applied to RNA sample i, and β_g is a vector of regression coefficients by which the covariate effects are mediated for gene g. The quadratic variance function specifies the negative binomial GLM distributional family. The use of the negative binomial distribution is equivalent to treating the π_{gi} as gamma distributed.

Fitting the GLMs

The derivative of the log-likelihood with respect to the coefficients β_g is $X^T\mathbf{z}_g$, where X is the design matrix with columns $\mathbf{x}_i$ and $z_{gi} = (y_{gi} - \mu_{gi})/(1 + \phi_g\mu_{gi})$. The Fisher information matrix for the coefficients can be written as $\mathcal{I}_g = X^T W_g X$, where W_g is the diagonal matrix of working weights from standard GLM theory (43). The Fisher scoring iteration to find the maximum likelihood estimate of β_g is therefore $\beta_g^{\mathrm{new}} = \beta_g^{\mathrm{old}} + \delta$ with $\delta = (X^T W_g X)^{-1} X^T \mathbf{z}_g$. This iteration usually produces an increase in the likelihood function, but the likelihood can also decrease representing divergence from the required solution. On the other hand, there always exists a stepsize modifier α with $0 < \alpha < 1$ such that $\beta_g^{\mathrm{new}} = \beta_g^{\mathrm{old}} + \alpha\delta$ produces an increase in the likelihood. Choosing α so that this is so at each iteration is known as a line search strategy (44,45).

Fisher's scoring iteration can be viewed as an approximate Newton-Raphson algorithm, with the Fisher information matrix approximating the second derivative matrix. The line search strategy may be used with any approximation to the second derivative matrix that is positive definite. Our implemention uses a computationally convenient approximation. Without loss of generality, the linear model can be parametrized so that $X^T X = I$. If this is done, and if the μ_{gi} also happen to be constant over i for a given gene g, then the information matrix simpifies considerably to $\mu_g/(1 + \phi_g\mu_g)$ times the identity matrix I. Taking this as the approximation to the information matrix, the Fisher scoring step with line search modification becomes simply $\delta = \alpha X^T \mathbf{z}_g$, where the multiplier $\mu_g/(1 + \phi_g\mu_g)$ has been absorbed into the stepsize factor α. In this formulation, α is no longer constrained to be less than one. In our implementation, each gene has its own stepsize α that is increased or decreased as the iteration proceeds.

Cox–Reid adjusted profile likelihood

The adjusted profile likelihood (APL) for ϕ_g is the penalized log-likelihood

$$\mathrm{APL}_g(\phi_g) = \ell(\phi_g; \mathbf{y}_g, \hat{\beta}_g) - \frac{1}{2}\log\det\mathcal{I}_g.$$

where $\mathbf{y}_g$ is the vector of counts for gene g, $\hat{\beta}_g$ is the estimated coefficient vector, $\ell()$ is the log-likelihood function and $\mathcal{I}_g$ is the Fisher information matrix. The Cholesky decomposition (46) provides a numerically stable and efficient algorithm for computing the determinant of the information matrix. Specifically, logdet $\mathcal{I}_g$ is the sum of the logarithms of the diagonal elements of the Cholesky factor R, where $\mathcal{I}_g = R^T R$ and R is upper triangular. The matrix R can be obtained as a by product of the QR-decomposition used in standard linear model fitting. In our implementation, the Cholesky calculations are carried out in a vectorized fashion, computed for all genes in parallel.

Simulations

Artificial data sets were generated with negative binomial distributed counts for a fixed total number of 10 000 genes. The expected count size varied between genes according to a gamma distribution with shape parameter 0.5, an *ad hoc* choice that happened to mimic the size distribution of the carcinoma data. The average dispersion was set to 0.16 (BCV = 0.4). In one simulation, all genes had the same

Defining model

- $b''(\theta) = \mu$
- $a(\phi) = 1 + \phi\mu$

Estimation dispersion: profiling

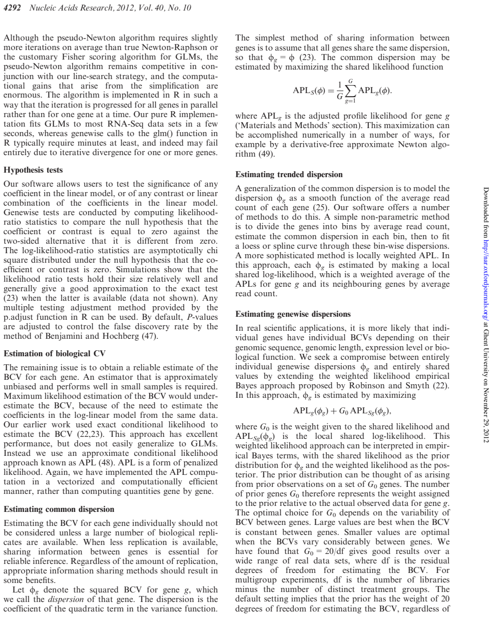
- $I_g = -\mathrm{E}\left[\frac{\partial^2 L}{\partial^2 \boldsymbol{\beta}}\right]$

Do APL with Gaussian for explaining rationale

$$-\frac{1}{2}\log\det I = \frac{p}{2}\log\sigma^2 + \frac{1}{2}\log|\mathbf{X}^T\mathbf{X}|$$

$$-2APL \sim (N-p)\log\sigma^2 + \frac{1}{\sigma^2}\|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$$

NegBinom GLM for RNA-seq



4292 *Nucleic Acids Research, 2012, Vol. 40, No. 10*

Although the pseudo-Newton algorithm requires slightly more iterations on average than true Newton-Raphson or the customary Fisher scoring algorithm for GLMs, the pseudo-Newton algorithm remains competitive in conjunction with our line-search strategy, and the computational gains that arise from the simplification are enormous. The algorithm is implemented in R in such a way that the iteration is progressed for all genes in parallel rather than for one gene at a time. Our pure R implementation fits GLMs to most RNA-Seq data sets in a few seconds, whereas genewise calls to the glm() function in R typically require minutes at least, and indeed may fail entirely due to iterative divergence for one or more genes.

Hypothesis tests

Our software allows users to test the significance of any coefficient in the linear model, or of any contrast or linear combination of the coefficients in the linear model. Genewise tests are conducted by computing likelihood-ratio statistics to compare the null hypothesis that the coefficient or contrast is equal to zero against the two-sided alternative that it is different from zero. The log-likelihood-ratio statistics are asymptotically chi square distributed under the null hypothesis that the coefficient or contrast is zero. Simulations show that the likelihood ratio tests hold their size relatively well and generally give a good approximation to the exact test (23) when the latter is available (data not shown). Any multiple testing adjustment method provided by the p.adjust function in R can be used. By default, *P*-values are adjusted to control the false discovery rate by the method of Benjamini and Hochberg (47).

Estimation of biological CV

The remaining issue is to obtain a reliable estimate of the BCV for each gene. An estimator that is approximately unbiased and performs well in small samples is required. Maximum likelihood estimation of the BCV would underestimate the BCV, because of the need to estimate the coefficients in the log-linear model from the same data. Our earlier work used exact conditional likelihood to estimate the BCV (22,23). This approach has excellent performance, but does not easily generalize to GLMs. Instead we use an approximate conditional likelihood approach known as APL (48). APL is a form of penalized likelihood. Again, we have implemented the APL computation in a vectorized and computationally efficient manner, rather than computing quantities gene by gene.

Estimating common dispersion

Estimating the BCV for each gene individually should not be considered unless a large number of biological replicates are available. When less replication is available, sharing information between genes is essential for reliable inference. Regardless of the amount of replication, appropriate information sharing methods should result in some benefits.

Let ϕ_g denote the squared BCV for gene g, which we call the *dispersion* of that gene. The dispersion is the coefficient of the quadratic term in the variance function. The simplest method of sharing information between genes is to assume that all genes share the same dispersion, so that $\phi_g = \phi$ (23). The common dispersion may be estimated by maximizing the shared likelihood function

$$\mathrm{APL}_S(\phi) = \frac{1}{G}\sum_{g=1}^{G} \mathrm{APL}_g(\phi).$$

where APL_g is the adjusted profile likelihood for gene g ('Materials and Methods' section). This maximization can be accomplished numerically in a number of ways, for example by a derivative-free approximate Newton algorithm (49).

Estimating trended dispersion

A generalization of the common dispersion is to model the dispersion ϕ_g as a smooth function of the average read count of each gene (25). Our software offers a number of methods to do this. A simple non-parametric method is to divide the genes into bins by average read count, estimate the common dispersion in each bin, then to fit a loess or spline curve through these bin-wise dispersions. A more sophisticated method is locally weighted APL. In this approach, each ϕ_g is estimated by making a local shared log-likelihood, which is a weighted average of the APLs for gene g and its neighbouring genes by average read count.

Estimating genewise dispersions

In real scientific applications, it is more likely that individual genes have individual BCVs depending on their genomic sequence, genomic length, expression level or biological function. We seek a compromise between entirely individual genewise dispersions ϕ_g and entirely shared values by extending the weighted likelihood empirical Bayes approach proposed by Robinson and Smyth (22). In this approach, ϕ_g is estimated by maximizing

$$\mathrm{APL}_g(\phi_g) + G_0\,\mathrm{APL}_{Sg}(\phi_g),$$

where G_0 is the weight given to the shared likelihood and $\mathrm{APL}_{Sg}(\phi_g)$ is the local shared log-likelihood. This weighted likelihood approach can be interpreted in empirical Bayes terms, with the shared likelihood as the prior distribution for ϕ_g and the weighted likelihood as the posterior. The prior distribution can be thought of as arising from prior observations on a set of G_0 genes. The number of prior genes G_0 therefore represents the weight assigned to the prior relative to the actual observed data for gene g. The optimal choice for G_0 depends on the variability of BCV between genes. Large values are best when the BCV is constant between genes. Smaller values are optimal when the BCVs vary considerably between genes. We have found that $G_0 = 20/\mathrm{df}$ gives good results over a wide range of real data sets, where df is the residual degrees of freedom for estimating the BCV. For multigroup experiments, df is the number of libraries minus the number of distinct treatment groups. The default setting implies that the prior has the weight of 20 degrees of freedom for estimating the BCV, regardless of

Defining model

- $b''(\theta) = \mu$
- $a(\phi) = 1 + \phi\mu$

Estimation dispersion: profiling

- $I_g = -\mathrm{E}\left[\frac{\partial^2 L}{\partial^2 \boldsymbol{\beta}}\right]$

Do APL with Gaussian for explaining rationale

$$-\frac{1}{2}\log \det I = \frac{p}{2}\log \sigma^2 + \frac{1}{2}\log |\mathbf{X}^T\mathbf{X}|$$

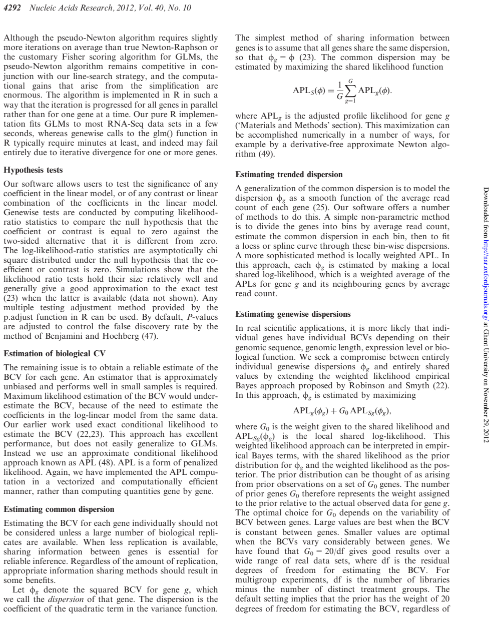
$$-2APL \sim (N-p)\log \sigma^2 + \frac{1}{\sigma^2}\|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$$

Tagwise: Weighted dispersion estimation

$$(1-\alpha)APL_g(\phi_g) + \alpha APL_S(\phi_g)$$

- Dispersion: common $ALP_S(\phi)$, trended, gene wise $APL_g(\phi_g)$, tagwise $(1-\alpha)APL_g(\phi_g) + \alpha APL_S(\phi_g)$

NegBinom GLM for RNA-seq



4292 Nucleic Acids Research, 2012, Vol. 40, No. 10

Although the pseudo-Newton algorithm requires slightly more iterations on average than true Newton-Raphson or the customary Fisher scoring algorithm for GLMs, the pseudo-Newton algorithm remains competitive in conjunction with our line-search strategy, and the computational gains that arise from the simplification are enormous. The algorithm is implemented in R in such a way that the iteration is progressed for all genes in parallel rather than for one gene at a time. Our pure R implementation fits GLMs to most RNA-Seq data sets in a few seconds, whereas genewise calls to the glm() function in R typically require minutes at least, and indeed may fail entirely due to iterative divergence for one or more genes.

Hypothesis tests

Our software allows users to test the significance of any coefficient in the linear model, or of any contrast or linear combination of the coefficients in the linear model. Genewise tests are conducted by computing likelihood-ratio statistics to compare the null hypothesis that the coefficient or contrast is equal to zero against the two-sided alternative that it is different from zero. The log-likelihood-ratio statistics are asymptotically chi square distributed under the null hypothesis that the coefficient or contrast is zero. Simulations show that the likelihood ratio tests hold their size relatively well and generally give a good approximation to the exact test (23) when the latter is available (data not shown). Any multiple testing adjustment method provided by the p.adjust function in R can be used. By default, *P*-values are adjusted to control the false discovery rate by the method of Benjamini and Hochberg (47).

Estimation of biological CV

The remaining issue is to obtain a reliable estimate of the BCV for each gene. An estimator that is approximately unbiased and performs well in small samples is required. Maximum likelihood estimation of the BCV would underestimate the BCV, because of the need to estimate the coefficients in the log-linear model from the same data. Our earlier work used exact conditional likelihood to estimate the BCV (22,23). This approach has excellent performance, but does not easily generalize to GLMs. Instead we use an approximate conditional likelihood approach known as APL (48). APL is a form of penalized likelihood. Again, we have implemented the APL computation in a vectorized and computationally efficient manner, rather than computing quantities gene by gene.

Estimating common dispersion

Estimating the BCV for each gene individually should not be considered unless a large number of biological replicates are available. When less replication is available, sharing information between genes is essential for reliable inference. Regardless of the amount of replication, appropriate information sharing methods should result in some benefits.

Let ϕ_g denote the squared BCV for gene g, which we call the *dispersion* of that gene. The dispersion is the coefficient of the quadratic term in the variance function. The simplest method of sharing information between genes is to assume that all genes share the same dispersion, so that $\phi_g = \phi$ (23). The common dispersion may be estimated by maximizing the shared likelihood function

$$\mathrm{APL}_S(\phi) = \frac{1}{G}\sum_{g=1}^{G}\mathrm{APL}_g(\phi).$$

where APL_g is the adjusted profile likelihood for gene g ('Materials and Methods' section). This maximization can be accomplished numerically in a number of ways, for example by a derivative-free approximate Newton algorithm (49).

Estimating trended dispersion

A generalization of the common dispersion is to model the dispersion ϕ_g as a smooth function of the average read count of each gene (25). Our software offers a number of methods to do this. A simple non-parametric method is to divide the genes into bins by average read count, estimate the common dispersion in each bin, then to fit a loess or spline curve through these bin-wise dispersions. A more sophisticated method is locally weighted APL. In this approach, each ϕ_g is estimated by making a local shared log-likelihood, which is a weighted average of the APLs for gene g and its neighbouring genes by average read count.

Estimating genewise dispersions

In real scientific applications, it is more likely that individual genes have individual BCVs depending on their genomic sequence, genomic length, expression level or biological function. We seek a compromise between entirely individual genewise dispersions ϕ_g and entirely shared values by extending the weighted likelihood empirical Bayes approach proposed by Robinson and Smyth (22). In this approach, ϕ_g is estimated by maximizing

$$\mathrm{APL}_g(\phi_g) + G_0\,\mathrm{APL}_{Sg}(\phi_g),$$

where G_0 is the weight given to the shared likelihood and $\mathrm{APL}_{Sg}(\phi_g)$ is the local shared log-likelihood. This weighted likelihood approach can be interpreted in empirical Bayes terms, with the shared likelihood as the prior distribution for ϕ_g and the weighted likelihood as the posterior. The prior distribution can be thought of as arising from prior observations on a set of G_0 genes. The number of prior genes G_0 therefore represents the weight assigned to the prior relative to the actual observed data for gene g. The optimal choice for G_0 depends on the variability of BCV between genes. Large values are best when the BCV is constant between genes. Smaller values are optimal when the BCVs vary considerably between genes. We have found that $G_0 = 20/\mathrm{df}$ gives good results over a wide range of real data sets, where df is the residual degrees of freedom for estimating the BCV. For multigroup experiments, df is the number of libraries minus the number of distinct treatment groups. The default setting implies that the prior has the weight of 20 degrees of freedom for estimating the BCV, regardless of

Defining model

- $b''(\theta) = \mu$
- $a(\phi) = 1 + \phi\mu$

Estimation dispersion: profiling

- $I_g = -\mathrm{E}\left[\frac{\partial^2 L}{\partial^2 \boldsymbol{\beta}}\right]$

Do APL with Gaussian for explaining rationale

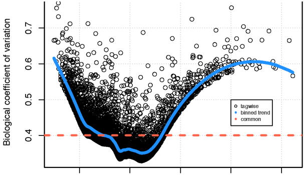
$$-\frac{1}{2}\log \det I = \frac{p}{2}\log\sigma^2 + \frac{1}{2}\log|\mathbf{X}^T\mathbf{X}|$$

$$-2APL \sim (N-p)\log\sigma^2 + \frac{1}{\sigma^2}\|\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}\|_2^2$$

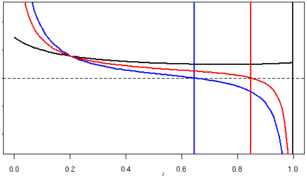
Tagwise: Weighted dispersion estimation

$$(1-\alpha)APL_g(\phi_g) + \alpha APL_S(\phi_g)$$

- Dispersion: common $ALP_S(\phi)$, trended, gene wise $APL_g(\phi_g)$, tagwise $(1-\alpha)APL_g(\phi_g) + \alpha APL_S(\phi_g)$



Score (1^{st} derivative of LL)



$$\delta = \frac{\phi}{\phi+1}$$

Defining model

- $b''(\theta) = \mu$
- $a(\phi) = 1 + \phi\mu$

Estimation dispersion: profiling

- $I_g = -\mathrm{E}\left[\frac{\partial^2 L}{\partial^2 \boldsymbol{\beta}}\right]$

Do APL with Gaussian for explaining rationale

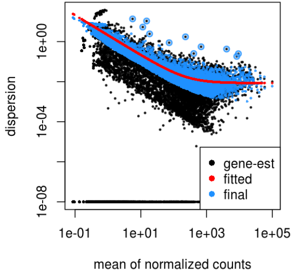
$$-\frac{1}{2}\log \det I = \frac{p}{2}\log\sigma^2 + \frac{1}{2}\log|\mathbf{X}^T\mathbf{X}|$$

$$-2APL \sim (N-p)\log\sigma^2 + \frac{1}{\sigma^2}\|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$$

Tagwise: Weighted dispersion estimation

$$(1-\alpha)APL_g(\phi_g) + \alpha APL_S(\phi_g)$$

- Dispersion: common $ALP_S(\phi)$, trended, gene wise $APL_g(\phi_g)$, tagwise $(1-\alpha)APL_g(\phi_g) + \alpha APL_S(\phi_g)$



Defining model

- $b''(\theta) = \mu$
- $a(\phi) = 1 + \phi\mu$

Estimation dispersion: profiling

- $I_g = -\mathrm{E}\left[\frac{\partial^2 L}{\partial^2 \boldsymbol{\beta}}\right]$

Do APL with Gaussian for explaining rationale

$$-\frac{1}{2}\log \det I = \frac{p}{2}\log\sigma^2 + \frac{1}{2}\log|\mathbf{X}^T\mathbf{X}|$$

$$-2APL \sim (N-p)\log\sigma^2 + \frac{1}{\sigma^2}\|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$$

Tagwise: Weighted dispersion estimation

$$(1-\alpha)APL_g(\phi_g) + \alpha APL_S(\phi_g)$$

- Dispersion: common $ALP_S(\phi)$, trended, gene wise $APL_g(\phi_g)$, tagwise $(1-\alpha)APL_g(\phi_g) + \alpha APL_S(\phi_g)$
- DESeq: maximum trended vs tagwise
- DESeq 2: Tagwise but outliers are not shrunken

Hypothesis testing: Large sample theory

- **LRT-test**

  $$\lambda = 2l_e - 2l_0$$

  for nested models (extended model (e) and null model (0)) follows an asymptotic χ^2-distribution with $df = p_e - p_0$ degrees of freedom and p_e (p_0) the number of parameters in the extended (null) model.

- **Wald test** follows immediately from the information matrix for generalized linear models

  $$I(\boldsymbol{\beta}) = \mathbf{X}^\mathbf{T}\mathbf{W}\mathbf{X}$$

  so large sample distribution of the maximum likelihood estimator $\hat{\boldsymbol{\beta}}$ is multivariate normal

  $$\hat{\boldsymbol{\beta}} \sim N\left[\boldsymbol{\beta}, \left(\mathbf{X}^\mathbf{T}\mathbf{W}\mathbf{X}\right)^{-1}\right]$$

Limma-voom

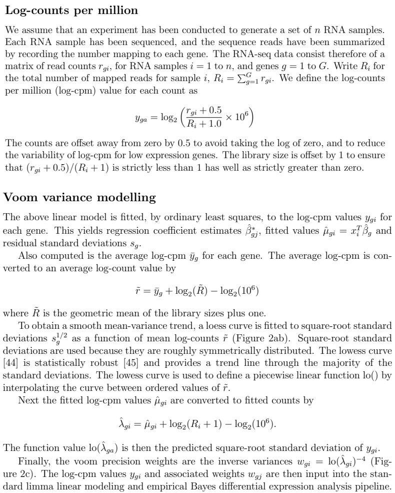
- Count models vs transformation: Poisson counts, $\sqrt(y)$ stabilises the variance, insufficient for negative binomial. Log transformation: the transformed data are still heteroscedastic.$\rightarrow$ limma-voom
- Use normalized log-cpm Limma pipeline for sequencing

Limma-voom

- Problem: counts have a mean variance relationship: heteroscedastic
- How do we deal with heteroscedasticity in traditional linear models?

Limma-voom

- Problem: counts have a mean variance relationship: heteroscedastic
- How do we deal with heteroscedasticity in traditional linear models?
- Two stage approach:
  1. Stage I
     - OLS
     - Estimate variances at each data point
     - Use variances as weights: $W = \text{diag}[1/\hat{\sigma}_i^2]$
  2. Stage II WLS $\text{argmin}_{\boldsymbol{\beta}}\{(\mathbf{y}-\mathbf{X}\beta)^T\mathbf{W}(\mathbf{y}-\mathbf{X}\beta)\}$
- Port this idea to RNA-seq pipeline



Log-counts per million

We assume that an experiment has been conducted to generate a set of n RNA samples. Each RNA sample has been sequenced, and the sequence reads have been summarized by recording the number mapping to each gene. The RNA-seq data consist therefore of a matrix of read counts r_{gi}, for RNA samples $i = 1$ to n, and genes $g = 1$ to G. Write R_i for the total number of mapped reads for sample i, $R_i = \sum_{g=1}^{G} r_{gi}$. We define the log-counts per million (log-cpm) value for each count as

$$y_{ga} = \log_2\left(\frac{r_{gi} + 0.5}{R_i + 1.0} \times 10^6\right)$$

The counts are offset away from zero by 0.5 to avoid taking the log of zero, and to reduce the variability of log-cpm for low expression genes. The library size is offset by 1 to ensure that $(r_{gi} + 0.5)/(R_i + 1)$ is strictly less than 1 has well as strictly greater than zero.

Voom variance modelling

The above linear model is fitted, by ordinary least squares, to the log-cpm values y_{gi} for each gene. This yields regression coefficient estimates $\hat{\beta}^*_{gj}$, fitted values $\hat{\mu}_{gi} = x_i^T \hat{\beta}_g$ and residual standard deviations s_g.

Also computed is the average log-cpm $\bar{y}_g$ for each gene. The average log-cpm is converted to an average log-count value by

$$\tilde{r} = \bar{y}_g + \log_2(\tilde{R}) - \log_2(10^6)$$

where $\tilde{R}$ is the geometric mean of the library sizes plus one.

To obtain a smooth mean-variance trend, a loess curve is fitted to square-root standard deviations $s_g^{1/2}$ as a function of mean log-counts $\tilde{r}$ (Figure 2ab). Square-root standard deviations are used because they are roughly symmetrically distributed. The lowess curve [44] is statistically robust [45] and provides a trend line through the majority of the standard deviations. The lowess curve is used to define a piecewise linear function lo() by interpolating the curve between ordered values of $\tilde{r}$.

Next the fitted log-cpm values $\hat{\mu}_{gi}$ are converted to fitted counts by

$$\hat{\lambda}_{gi} = \hat{\mu}_{gi} + \log_2(R_i + 1) - \log_2(10^6).$$

The function value $\text{lo}(\hat{\lambda}_{ga})$ is then the predicted square-root standard deviation of y_{gi}.

Finally, the voom precision weights are the inverse variances $w_{gi} = \text{lo}(\hat{\lambda}_{gi})^{-4}$ (Figure 2c). The log-cpm values y_{gi} and associated weights w_{gj} are then input into the standard limma linear modeling and empirical Bayes differential expression analysis pipeline.

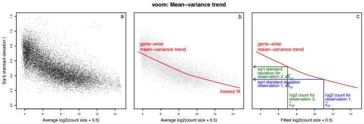


Figure 2: Voom mean-variance modelling. Panel (a), gene-wise square-root residual standard deviations are plotted against average log-count. Panel (b), a functional relationship between gene-wise means and variances is given by a robust lowess fit to the points. Panel (c), the mean-variance trend enables each observation to map to a square-root standard deviation value using its fitted value for log-count.

Law et al. (2013). Genome Biology