Publication Supplement
Institute of Bioinformatics, Johannes Kepler University Linz

DEXUS: Identifying Differential Expression in RNA-Seq Studies with Unknown Conditions
— *Supplementary Information* —

Günter Klambauer, Thomas Unterthiner, and Sepp Hochreiter
Institute of Bioinformatics, Johannes Kepler University, Linz, Austria

Institute of Bioinformatics
Johannes Kepler University Linz
A-4040 Linz, Austria

Tel. +43 732 2468 4520
Fax +43 732 2468 4539
http://www.bioinf.jku.at

Contents

List of Figures

List of Tables

S1 Introduction

This report gives supplementary information to the manuscript "DEXUS: Identifying Differential Expression in RNA-Seq Studies with Unknown Conditions". The supplementary informations contain

- a result that with many conditions the detection of differential expression is only possible with a large number of samples and high coverage;
- a description and a motivation of the DEXUS model;
- a derivation of the DEXUS model selection algorithm;
- information on initialization and hyperparameter adjustment;
- results of additional experiments;
- methods for calling differentially expressed transcripts;
- further details on experiments;
- additional information on the DEXUS software, methods compared, data sets, and evaluation criteria.

Summary. Differentially expressed transcripts in RNA-Seq experiments with *known conditions* can be detected by current RNA-Seq methods. These methods test differential expression between two or more known conditions based on read counts per transcript. However in more general study designs some conditions are usually unknown, though genes may be differentially expressed between them. Current RNA-Seq methods cannot identify differentially expressed transcripts in data with unknown conditions. We suggest DEXUS, a statistical model based on finite mixture of negative binomial distributions to detect differential expression in studies with *unknown conditions*.

S2 Detection of Differential Expression With Many Conditions is Difficult

RNA-Seq data are usually represented as read counts per transcript. Read count data from technical replicates follow a Poisson distribution (Marioni *et al.* 2008). However read counts from biological replicates follow a negative binomial distribution (Anders and Huber 2010; Robinson *et al.* 2010; Hardcastle and Kelly 2010; Li and Tibshirani 2011; Wu *et al.* 2013), because the biological variation leads to overdispersion (Hansen *et al.* 2011). To confirm these findings, we analyzed RNA-Seq data sets using different normalizations. An example is the "European HapMap" data set (Montgomery *et al.* 2010), which contains RNA-Seq data of 60 individuals. After upper quartile normalization of these RNA-Seq data, we calculated the variance-to-mean ratio for each transcript. The density of this ratio is shown in Fig. S1. The vast majority of transcripts has a variance-to-mean ratio greater than one and, therefore, is in accordance with the negative binomial distribution. The value of this ratio is 1 for the Poisson distribution and smaller than 1 for the binomial distribution. The density of the variance-to-mean ratio for other RNA-Seq data sets

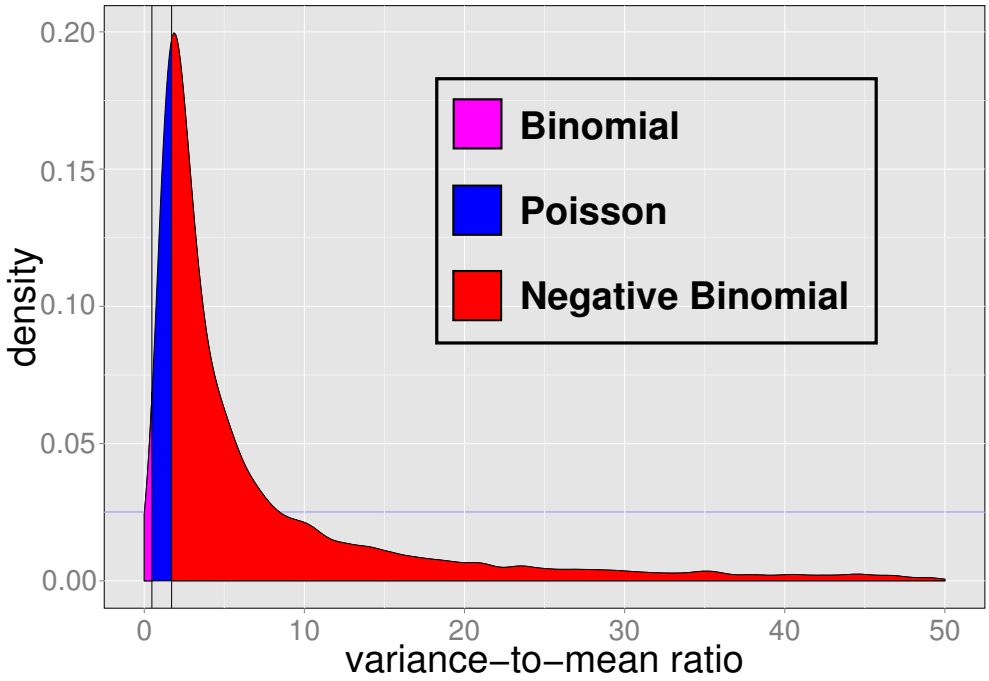

Figure S1: Density of the variance-to-mean ratio from RNA-Seq data of 60 individuals (Montgomery *et al.* 2010). Variance-to-mean ratios around one are characteristic for a Poisson distribution (blue), those smaller than one for a binomial distribution (purple), those larger than one for a negative binomial distribution (red). Thresholds are set by the test statistics of a Poisson test. The majority of transcripts has read counts that are in accordance with the negative binomial distribution.

is very similar to this data set (Bottomly *et al.* 2011; Pickrell *et al.* 2010; Blekhman *et al.* 2010; Nagalakshmi *et al.* 2008).

Due to these findings, we assume that read counts of a *set of biological replicates* follows a *negative binomial distribution* with *mean expression level* μ. The negative binomial distribution with mean μ and size r (representing the variance) is given by

$$\mathrm{NB}(x;\ \mu, r) \ = \ \frac{\Gamma(x+r)}{\Gamma(x+1)\Gamma(r)} \left(\frac{\mu}{\mu+r}\right)^{x} \left(\frac{r}{\mu+r}\right)^{r} . \tag{S1}$$

Other parametrizations and properties of the negative binomial distribution are discussed in Subsection S3.1.2.

We assume that biological replicates are generated under a particular *condition*, therefore a set of biological replicates corresponds to a particular condition. We define differential expression as different expression level between conditions (sets of replicates), where under each condition read counts are generated by a particular negative binomial distribution. A transcript is *differentially expressed* if (1) the mean expression levels μ for different conditions are different and (2) samples are observed under at least two different conditions.

If the read count data of a transcript can be explained by one condition, i.e. one negative

binomial

$$p(x) \;=\; \mathrm{NB}(x;\; \mu, r)\;, \tag{S2}$$

then this transcript is not differentially expressed. If the read counts follow a mixture of negative binomials

$$p(x) \;=\; \int p(\mu)\; \mathrm{NB}(x;\; \mu, r)\; d\mu\;, \tag{S3}$$

then the transcript is differentially expressed. Here $p(\mu)$ is the distribution of expression levels, each of which corresponds to a condition. Differential expression is identified by distinguishing a negative binomial from a mixture of negative binomials using the read count data.

An analytically tractable mixture of negative binomials is the beta negative binomial distribution (BNB):

$$\mathrm{BNB}(x;\; \iota, \kappa, r) \;=\; \int_0^\infty \underbrace{\frac{r}{(\mu + r)^2}\; \mathrm{B}(\frac{\mu}{\mu + r};\; \iota, \kappa)}_{p(\mu)}\; \mathrm{NB}(x;\; \mu, r)\; d\mu\;, \tag{S4}$$

where $\mathrm{B}(a;\; \iota, \kappa)$ is the density of the beta distribution with parameters ι and κ. Differential expression is identified by distinguishing a negative binomial from a beta negative binomial distribution (BNB) using the read count data.

We want to determine how many read counts and which coverages (given by the μs) are necessary to distinguish a negative binomial from a BNB, i.e. to identify differential expression. Whether the read count data is better represented by a more complex than by a simpler model can be decided by means of the Bayesian Information Criterion (BIC):

$$\mathrm{BIC}_{\mathcal{M}} \;=\; -\,2\;\log \mathrm{L}_{\mathcal{M}} \;+\; l_{\mathcal{M}}\;\log N\;, \tag{S5}$$

where $\mathrm{L}_{\mathcal{M}}$ is the likelihood of the model $\mathcal{M}$, $l_{\mathcal{M}}$ is the number of parameters of the model $\mathcal{M}$, and N is the number of samples. The model with smaller BIC is more appropriate to represent the data.

If the difference between the BIC of the BNB and the BIC of the negative binomial model $\mathrm{BIC}_{\mathrm{BNB}} - \mathrm{BIC}_{\mathrm{NB}}$ is negative, then BNB better represents the read counts and hints at differential expression. The number of parameters are $l_{\mathrm{BNB}} = 3$ and $l_{\mathrm{NB}} = 2$ for these models which should represent the read counts $\{x_1, \ldots, x_N\}$ for N samples. Therefore detecting differential expression requires

$$\begin{aligned}
&\mathrm{BIC}_{\mathrm{BNB}} \;-\; \mathrm{BIC}_{\mathrm{NB}} \;<\; 0 \qquad\qquad (\mathrm{S6})\\
\Leftrightarrow &-\;2\;\sum_{k=1}^{N} (\log \mathrm{BNB}(x_k)) + \;3\;\log N \;+\; 2\;\sum_{k=1}^{N} (\log \mathrm{NB}(x_k)) \;-\; 2\;\log N \;<\; 0\\
\Leftrightarrow &2\;\frac{1}{N}\;\sum_{k=1}^{N} \log\left(\frac{\mathrm{NB}(x_k)}{\mathrm{BNB}(x_k)}\right) \;+\; \frac{\log N}{N} \;<\; 0\;.
\end{aligned}$$

If averaging over data sets $\{x_1, \ldots, x_N\}$ drawn from the BNB, the following equation holds:

$$\begin{aligned} \mathrm{E}\left(\frac{1}{N}\sum_{k=1}^{N}\log\left(\frac{\mathrm{NB}(x_k)}{\mathrm{BNB}(x_k)}\right)\right) &= \frac{1}{N}\sum_{k=1}^{N}\mathrm{E}\left(\log\left(\frac{\mathrm{NB}(x_k)}{\mathrm{BNB}(x_k)}\right)\right) \\ &= \mathrm{E}\left(\log\left(\frac{\mathrm{NB}(x_k)}{\mathrm{BNB}(x_k)}\right)\right) , \end{aligned} \tag{S7}$$

where E is the expectation under the beta negative binomial distribution BNB. We have to use the expectation under the BNB, because the read counts are assumed to arise from a transcript that is differentially expressed. Note that $\mathrm{E}\left(\log\left(\frac{\mathrm{NB}(x_k)}{\mathrm{BNB}(x_k)}\right)\right) = -\mathrm{KL}(\mathrm{BNB}||\mathrm{NB})$, where KL is the Kullback-Leibler divergence. If averaging over data sets with N read counts, for differential expression following criterion is required:

$$\begin{aligned} & 2\,\mathrm{E}\left(\log\left(\frac{\mathrm{NB}(x_k)}{\mathrm{BNB}(x_k)}\right)\right) + \frac{\log N}{N} < 0 \\ \Leftrightarrow\ & \frac{N}{\log N} > \frac{1}{2\,\mathrm{KL}(\mathrm{BNB}||\mathrm{NB})} . \end{aligned} \tag{S8}$$

The Kullback-Leibler divergence is an asymmetric distance between two distributions. Thus, the more similar the distributions are to each other, the smaller is the Kullback-Leibler divergence, the more samples N are required to detect differential expression.

In the following, we compute the number of read counts that are required to detect differential expression if using the BIC criterion to discriminate between the negative binomial and the BNB. Fig. S2 shows the first example of a BNB with parameters $\iota = 204$, $\kappa = 400$, and $r = 40$ vs. a negative binomial with parameters $\mu = 78.78$ and $r = 30.81$. The Kullback-Leibler divergence of these two distributions is 0.0002. According to the BIC criterion N=27,700 samples are required to identify differential expression, that is to detect that the read counts are from the BNB and not from the negative binomial. Fig. S3 shows a second example a BNB with parameters $\iota = 40$, $\kappa = 40$, and $r = 40$ vs. a negative binomial with parameters $\mu = 40.89$ and $r = 13.07$. In this case the Kullback-Leibler divergence is 0.02 and, therefore, N=1,800 samples are required to identify differential expression. Fig. S4 shows a third example of a BNB with parameters $\iota = 40$, $\kappa = 20$, and $r = 400$ vs. a negative binomial with parameters $\mu = 204.48$ and $r = 12.7$. In this case the Kullback-Leibler divergence is 0.003 and, therefore, N=1,300 samples are necessary to identify differential expression. Fig. S5 shows a fourth example of a BNB with parameters $\iota = 10$, $\kappa = 10$, and $r = 400$ vs. a negative binomial with parameters $\mu = 435.56$ and $r = 4.43$. In this case the Kullback-Leibler divergence is 0.03. Only N=89 samples are required to identify differential expression, because the mean read counts are large. Large mean read counts means that the coverage is high.

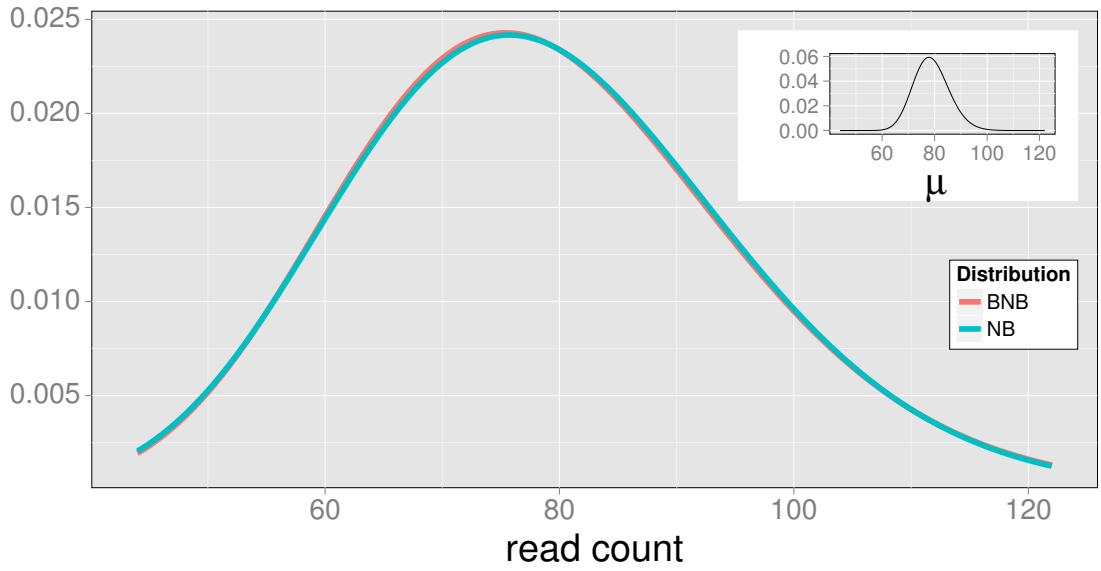

Figure S2: Probability mass function of a negative binomial distribution with $\mu = 78.78$ and $r = 30.81$ and beta negative binomial distribution with $\iota = 204$, $\kappa = 400$, and $r = 40$. The inlay figure shows the distribution $p(\mu)$ of the mean read count μ which generates the BNB distribution. The Kullback-Leibler divergence of these two distributions is 0.0001. According to the BIC criterion N=27,700 samples are required to identify differential expression.

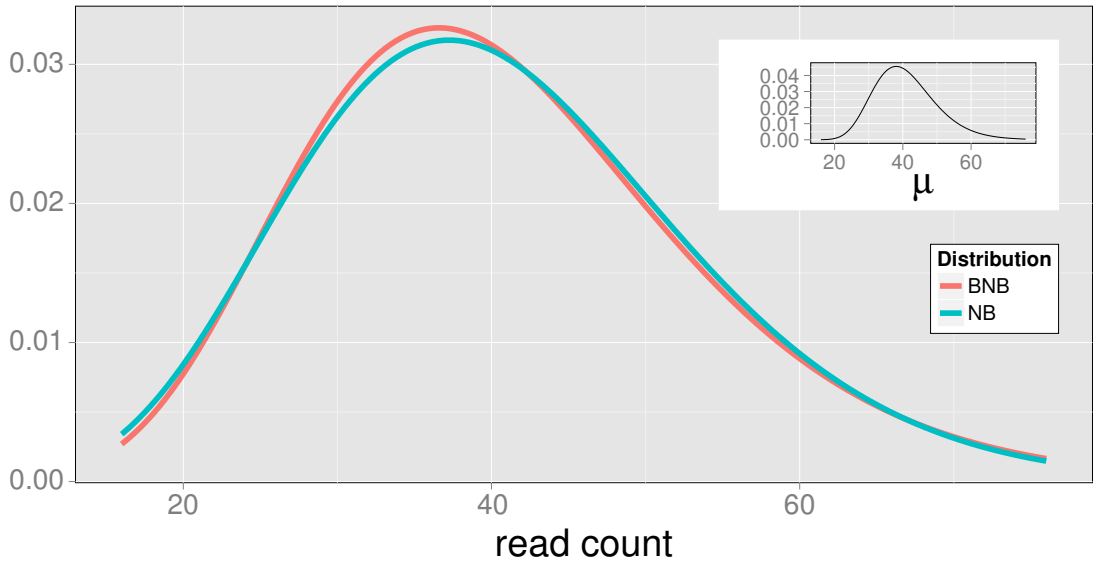

Figure S3: Probability mass function of a negative binomial distribution with $\mu = 40.89$ and $r = 13.07$ and beta negative binomial distribution with $\iota = 40$, $\kappa = 40$, and $r = 40$. The inlay figure shows the distribution $p(\mu)$ of the mean read count μ which generates the BNB distribution. The Kullback-Leibler divergence of these two distributions is 0.002. According to the BIC criterion N=1,800 samples are required to identify differential expression.

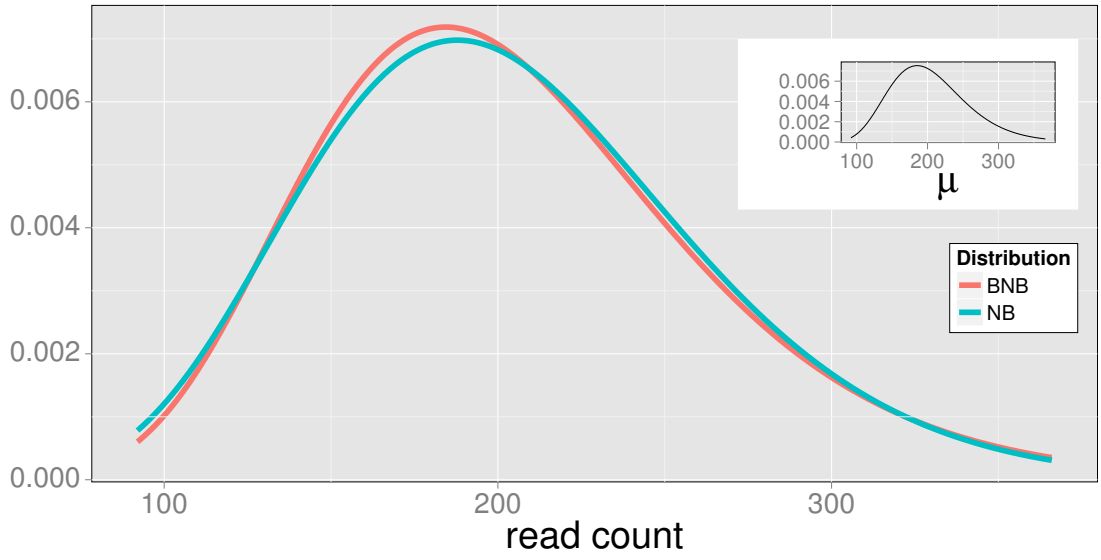

Figure S4: Probability mass function of a negative binomial distribution with $\mu = 204.48$ and $r = 12.7$ and beta negative binomial distribution with $\iota = 40$, $\kappa = 20$, and $r = 400$. The inlay figure shows the distribution $p(\mu)$ of the mean read count μ which generates the BNB distribution. The Kullback-Leibler divergence of these two distributions is 0.003. According to the BIC criterion N=1,300 samples are required to identify differential expression.

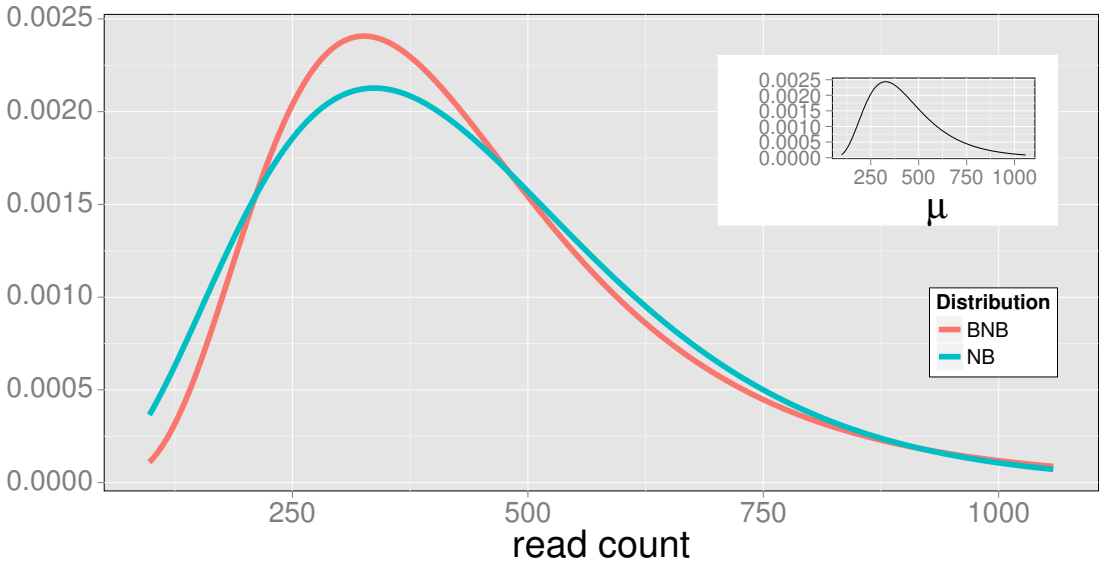

Figure S5: Probability mass function of a negative binomial distribution with $\mu = 435.56$ and $r = 4.43$ and beta negative binomial distribution with $\iota = 10$, $\kappa = 10$, and $r = 400$. The inlay figure shows the distribution $p(\mu)$ of the mean read count μ which generates the BNB distribution. The Kullback-Leibler divergence of these two distributions is 0.02. According to the BIC criterion N=87 samples are required to identify differential expression. The large mean read counts mean that the coverage is high. High coverage helps to detect differential expression.

If we want to determine whether a transcript is differentially expressed, we have to decide whether read counts are generated from a negative binomial or from a BNB. As shown above, this requires either a large number of samples or very high coverage (large μ).

For generating the BNB we used a distribution $p(\mu)$ as mixing distribution which is unimodal (see inlay figures in Fig. S2 to Fig. S5). Other unimodal distributions $p(\mu)$ lead to similar results: it is difficult to distinguish a mixture distribution from a negative binomial. If $p(\mu)$ is a weighted sum of delta distributions, then we obtain a finite mixture of negative binomials, that is a finite number of conditions:

$$p(x) \;=\; \sum_{i=1}^{n} \alpha_i \,\mathrm{NB}(x;\; \mu_i, r_i)\;. \tag{S9}$$

However, if the number of conditions is large compared to the number of samples, then it is still difficult to distinguish between a mixture model and a negative binomial using only the read counts. Thus, it is difficult to detect differential expression if the number of conditions is large compared to the number of samples.

In the following, we investigate whether a finite mixture of negative binomials can be distinguished from a single negative binomial distribution. Trivially, a mixture with only one component cannot be distinguished from a negative binomial. If one component of the mixture is dominant, then the mixture is close to a negative binomial. To avoid these trivial cases, we require that, under each condition, not more than half of the read counts are generated. The second trivial case is that

all mixture components are identical, where the mixture is a negative binomial. If the means of the compoment distributions are identical or are locally accumulated, then the mixture is close to a negative binomial. To avoid these case, we require that the means of the component distributions are placed equidistantly within a particular range of read counts. Concluding, we require $\alpha_i \leq 0.5$ and equidistantly distributed μ_i in a range to avoid trivial degenerated cases for which the mixture is close to a negative binomial. We will later use the α_i and μ_i to define an informative/non-informative call for a transcript, see Subsection S3.3. We choose the range of read counts between 0 and 150 similar to the example depicted in Fig. S2. First we construct a mixture of 10 negative binomials, where r_i are set to 50 and the mixture weights α_i follow a unimodal distribution. As shown in Fig. S6, the mixture and the negative binomial are hard to distinguish. Next we construct a mixture of two negative binomials. Now the mixture can be distinguished from the negative binomial as shown in Fig. S7. Note that all parameters are optimized to make the mixture as close as possible to the negative binomial. In contrast to the negative binomial, the mixture is a binomial distribution. Further the probability mass at the tails of the negative binomial is smaller than the mass of the mixture. This example shows that a mixture can be distinguished from a single negative binomial, if there are few conditions compared to the number of read counts.

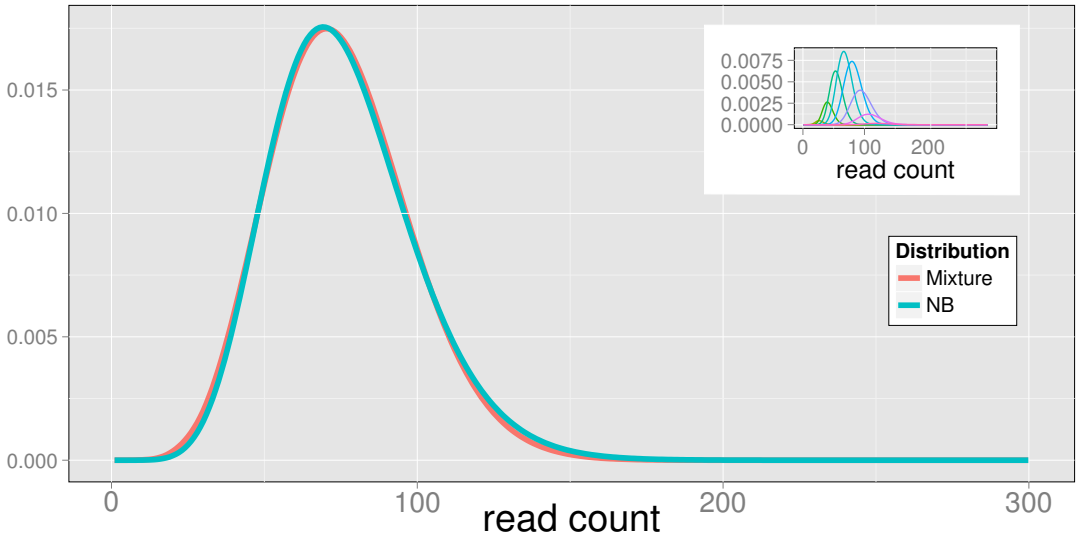

Figure S6: Probability mass function of a negative binomial distribution with $\mu = 75.22$ and $r = 8.87$ and a mixture of negative binomial distributions with mean parameters μ_i equidistantly in the range $[0, 150]$, r_i set to 50, and non-zero mixture weights α_i. The inlay figure shows the probability mass functions of the 10 negative binomial distributions of the mixture. The Kullback-Leibler divergence of these two distributions is 0.007.

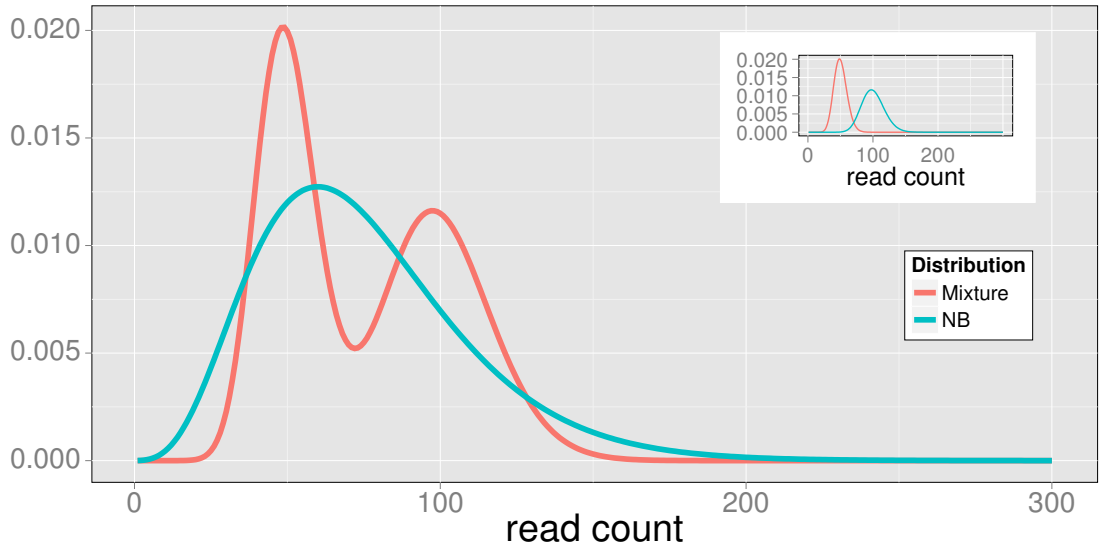

Figure S7: Probability mass function of a negative binomial distribution with $\mu = 75.07$ and $r = 15.08$ and a mixture of two negative binomial distributions with mean parameters equidistantly in $[0, 150]$ that is $(\mu_1, \mu_2) = (50, 100)$, with $(r_1, r_2) = (50, 50)$, and with mixture weights $(\alpha_1, \alpha_2) = (0.5, 0.5)$. The inlay figure shows the probability mass functions of the two negative binomial distributions of the mixture. The Kullback-Leibler divergence of these two distributions is 0.49.

Therefore we assume that the number of conditions (the number of sets of replicates) is small compared to the number of samples. We will demonstrate that under this assumption differentially expressed transcripts can be detected. As we show in the experiments, the detection of differential expression is more reliable with more samples in each of the conditions and with larger differences of the mean expression levels of the conditions.

S3 The DEXUS Method

In the first subsection, we introduce and motivate the DEXUS model, which is a finite mixture of negative binomials (as motivated in previous section). In the second subsection, we explain how DEXUS selects a model using read count data. Model selection is based on a Bayesian framework for maximizing the parameter posterior via an expectation maximization (EM) algorithm. The next subsection focuses on how to determine whether transcripts are differentially expressed. In the last subsection we investigate the sensitivity of a hyperparameter of DEXUS.

S3.1 The Mixture of Negative Binomials Model

In this subsection, we introduce and motivate the DEXUS model. Read counts per sample are modeled across samples for a gene, an exon, or a transcript. The following Subsection S3.1.1 introduces the finite mixture of negative binomial distributions model as motivated in previous

Section S2. Subsection S3.1.2 explains why we have chosen a particular parametrization of the negative binomial distribution. In the next Subsection S3.1.3 we show that the model is identifiable which is essential to infer parameters and to detect differential expression.

S3.1.1 The Model

As motivated in Section S2, read counts are distributed according to a finite mixture negative binomial distributions. Each mixture component corresponds to a condition, that is, the read counts are generated under this condition. A transcript is differentially expressed if read counts are generated under at least two different conditions with different expression levels μ.

If α_i is the probability of being in condition i, then read count x is distributed according to:

$$p(x) \;=\; \sum_{i=1}^{n} \alpha_i \, \mathrm{NB}(x;\; \mu_i, r_i) \;, \tag{S10}$$

where $\mathrm{NB}(x;\; \mu_i, r_i)$ is probability mass function of the negative binomial distribution with mean μ_i and size r_i. The $\alpha_i \geq 0$ are the non-negative mixture weights of the mixture model and sum to one: $\sum_{i=1}^{n} \alpha_i = 1$. The DEXUS model is a finite mixture of negative binomial distributions. As motivated in Section S2, we assume that the number of conditions n is smaller than the number of samples N: $n < N$.

S3.1.2 Parametrization of the Negative Binomial Distribution

We use the (μ, r)-*parametrization* of the negative binomial distribution (also called the *mean-size-parametrization*). The (p, r)-*parametrization* is the standard way to parametrize the negative binomial distribution. In the standard interpretation, p is the probability of success with $p \in [0, 1]$ and r is the number of failures with $r \in \mathbb{N}$. The probability mass function of the negative binomial in the (p, r)-*parametrization* is

$$\mathrm{NB}_{\mathrm{pr}}(x;\; p, r) \;=\; \frac{(x + r - 1)!}{x!\,(r-1)!}\,(1-p)^r\,p^x \;, \tag{S11}$$

where x is the number of successes until r failures are observed (note that the last observation must be a failure). The variance of the negative binomial is

$$\sigma^2 \;=\; \frac{pr}{(1-p)^2} \;. \tag{S12}$$

We chose a parametrization that includes μ because for RNA-Seq data with read counts per transcript, the mean read count is an important information. The mean is the expected or noise-free read count for a given condition and allows to determine whether transcripts are differentially expressed between conditions. Therefore in RNA-Seq applications, the negative binomial is re-parametrized using the mean parameter $\mu \in \mathbb{R}^+$ instead of the probability p. Furthermore in RNA-Seq applications, the overdispersion parameter $\phi \in \mathbb{R}^+$ is of interest to capture technical and biological variance which allows assessing the data quality. The overdispersion parameter measures how far the variance of the negative binomial exceeds the variance of a Poisson distribution, for which the variance is equal to the mean. These two parameters lead to the (μ, ϕ)-*parametrization*

of the negative binomial. The overdispersion parameter ϕ and the size parameter r have a very simple relationship: $\phi = 1/r$, i.e. the overdispersion is the inverse of the size parameter. The relationship between the parametrization (μ, ϕ) and (p, r) is:

$$\mu = \frac{pr}{1-p} \;\Rightarrow\; p = \frac{\mu}{\mu + r} \qquad\qquad \phi = \frac{1}{r} \;\Rightarrow\; r = \phi^{-1} \tag{S13}$$

We will choose r instead of ϕ in order to define a prior on r in a Bayesian framework. This prior gives larger overdispersions higher probabilities, which is essential to improve the parameter estimator for small sample sizes (see Subsection S3.2.1).

To represent all overdispersions ϕ and to perform model selection in a continuous space, real positive values of r are required. The definition Eq. (S11) can be generalized to $r \in \mathbb{R}^+$ by using the Γ-function instead of the factorial. Using positive real r, the probability mass function of the negative binomial for the (μ, r)-parametrization is

$$\mathrm{NB}(x;\, \mu, r) = \frac{\Gamma(x+r)}{\Gamma(x+1)\Gamma(r)} \left(\frac{\mu}{\mu+r}\right)^x \left(\frac{r}{\mu+r}\right)^r . \tag{S14}$$

The variance of the negative binomial with the (μ, ϕ)-parametrization or the (μ, r)-parametrization is

$$\sigma^2 = \frac{\frac{\mu}{\mu+r}\, r}{\left(1 - \frac{\mu}{\mu+r}\right)^2} = \mu + \frac{1}{r}\,\mu^2 = \mu + \phi\,\mu^2 . \tag{S15}$$

If DEXUS is applied to data with known conditions, we require an estimator of (μ, r) of a negative binomial distribution. In particular we require this estimator for the initialization of the EM algorithm in Subsection S3.2.4. We use the maximum likelihood estimator. Given a data set $\boldsymbol{x} = (x_1, \ldots, x_N)$ of counts of N samples, the maximum likelihood estimators for the (μ, r)-parametrization are as follows:

- The maximum likelihood estimator μ_{ML} for μ is

$$\mu_{\mathrm{ML}} = \frac{1}{N} \sum_{k=1}^{N} x_k . \tag{S16}$$

- A closed form for the maximum likelihood estimator r_{ML} for r does not exist. However following equation can be solved for r_{ML}:

$$\sum_{k=1}^{N} \psi(x_k + r_{\mathrm{ML}}) \;-\; N\,\psi(r_{\mathrm{ML}}) \;+\; N\,\log\left(\frac{r_{\mathrm{ML}}}{r_{\mathrm{ML}} \;+\; \frac{1}{N}\,\sum_{k=1}^{N} x_k}\right) \;=\; 0\,, \tag{S17}$$

  where ψ is the digamma function. The solution can be obtained numerically.

The estimator $(\mu_{\mathrm{ML}}, r_{\mathrm{ML}})$ is asymptotically unbiased and efficient. For finite sample size, however, neither the bias nor the variance of the estimator r_{ML} exists, because for data whose mean exceeds the variance r tends to infinity (Anscombe 1950).

S3.1.3 Identifiability of the Model

Finite mixtures of non-degenerate negative binomial distributions are identifiable (Yakowitz and Spragins 1968). For identifiable mixtures, Eq. (3) in (Yakowitz and Spragins 1968) and the text thereafter states that from

$$\sum_{i=1}^{n} \alpha_i \, \mathrm{NB}(x;\, \mu_i, r_i) \;=\; \sum_{i=1}^{n} \alpha'_i \, \mathrm{NB}(x;\, \mu'_i, r'_i) \,. \tag{S18}$$

follows

$$\begin{aligned} \alpha_i &= \alpha'_i \\ \mu_i &= \mu'_i \\ r_i &= r'_i \,. \end{aligned} \tag{S19}$$

We assumed that the components are sorted (avoids ambiguities through permutations of the components) and that the distributions of the components are mutually different (avoids ambiguous splitting of one component into more).

Identifiability is required for the maximum likelihood estimator to be consistent. Consistency means that a parameter estimator converges with more data points to the true parameter values. Since the parameter space will be made compact, the mixture model is continuous in its parameters, and the log mixture distribution can be bounded, the maximum likelihood estimator for the mixture of negative binomials is consistent. Note that below we will introduce an upper bound $r_{\max}$ for the size parameter r and a lower bound $\mu_{\min}$ for the mean parameter of the negative binomial distributions.

More importantly, identifiability of the mixture of negative binomials guarantees that differential expressed transcripts can be detected if sufficiently many read counts are available.

S3.2 Model Selection

In the next Subsection S3.2.1, DEXUS' expectation maximization (EM) algorithm for model selection is derived. We first define the Bayesian framework, then chose appropriate priors for the parameters, then derive a bound on the parameter posterior using the chosen priors, and then use this bound to derive the E-step and the M-step of the EM algorithm. This subsection is one of the central parts of this supplementary. The following Subsection S3.2.2 describes the case when the variance-to-mean ratio of one negative binomial is approaching one and converges toward a Poisson distribution. The next Subsection S3.2.3 summarizes DEXUS update rules for the iterative EM algorithm. Then Subsection S3.3 describes the initialization for the DEXUS model selection algorithm. The final Subsection S3.2.5 shows how the hyperparameter for the size parameter prior is adjusted depending on the number of samples.

S3.2.1 Derivation of the EM Algorithm

In a Bayes framework for model selection, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$, $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_n)$, and $\boldsymbol{r} = (r_1, \ldots, r_n)$ are considered as random variables, thus, $p(x)$ in Eq. (S10) becomes a conditional

probability $p(x \mid \boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{r})$, i.e. the likelihood that read count x has been produced by the model with parameters $\boldsymbol{\alpha}$, $\boldsymbol{\mu}$, and $\boldsymbol{r}$. The expectation maximization (EM) algorithm (Dempster *et al.* 1977) minimizes an upper bound on the negative log-posterior of the parameters. The parameter posterior of $\boldsymbol{\alpha}$, $\boldsymbol{\mu}$, and $\boldsymbol{r}$ is given by:

$$\begin{aligned} p(\boldsymbol{\mu}, \boldsymbol{r}, \boldsymbol{\alpha} \mid x) \;&=\; \frac{p(x \mid \boldsymbol{\mu}, \boldsymbol{r}, \boldsymbol{\alpha})\; p(\boldsymbol{\alpha})\; p(\boldsymbol{r}) p(\boldsymbol{\mu})}{\int p(x \mid \boldsymbol{\mu}, \boldsymbol{r}, \boldsymbol{\alpha})\; p(\boldsymbol{\alpha})\; p(\boldsymbol{r})\; p(\boldsymbol{\mu})\; d\boldsymbol{\alpha}\; d\boldsymbol{r}\; d\boldsymbol{\mu}} \\ &=\; \frac{1}{c(x)}\; p(x \mid \boldsymbol{\mu}, \boldsymbol{r}, \boldsymbol{\alpha})\; p(\boldsymbol{\alpha})\; p(\boldsymbol{r})\; p(\boldsymbol{\mu})\;, \end{aligned} \tag{S20}$$

where we assumed that the priors for $\boldsymbol{\alpha}$, $\boldsymbol{\mu}$ and $\boldsymbol{r}$ are independent of each other.

For deriving an upper bound on the log posterior as required by the EM algorithm, we deduce the following inequality *for one sample* x by introducing variables $\hat{\alpha}_i$ with $\sum_{i=1}^{n} \hat{\alpha}_i = 1$:

$$\begin{aligned} &-\; \log p(\boldsymbol{\mu}, \boldsymbol{r}, \boldsymbol{\alpha} \mid x) \\ &=\; -\; \log \sum_{i=1}^{n} \frac{\hat{\alpha}_i}{\hat{\alpha}_i}\; \alpha_i\, \mathrm{NB}(x;\; \mu_i, r_i) \;-\; \log p(\boldsymbol{\alpha}) \;-\; \log p(\boldsymbol{\mu}) \;-\; \log p(\boldsymbol{r}) \;+\; \log(c(x)) \\ &\leq\; -\; \sum_{i=1}^{n} \hat{\alpha}_i\; \log \left(\frac{\alpha_i}{\hat{\alpha}_i}\, \mathrm{NB}(x;\; \mu_i, r_i) \right) \;-\; \log p(\boldsymbol{\alpha}) \;-\; \log p(\boldsymbol{\mu}) \;-\; \log p(\boldsymbol{r}) \;+\; \log(c(x)) \quad (*) \\ &=\; -\; \sum_{i=1}^{n} \hat{\alpha}_i\; \log \left(\alpha_i\, \mathrm{NB}(x;\; \mu_i, r_i) \right) \;-\; \log p(\boldsymbol{\alpha}) \;-\; \log p(\boldsymbol{\mu}) \;-\; \log p(\boldsymbol{r}) \\ &\quad\; +\; \sum_{i=1}^{n} \hat{\alpha}_i\; \log \hat{\alpha}_i \;+\; \log(c(x))\;, \end{aligned} \tag{S21}$$

where $c(x)$ is independent of the parameters $\boldsymbol{\alpha}$, $\boldsymbol{\mu}$ and $\boldsymbol{r}$. We applied Jensen's inequality to obtain the line ending with the $(*)$-sign.

To derive an EM algorithm, we have to choose appropriate priors for the mixture weights $p(\boldsymbol{\alpha})$, the overdispersion parameters $p(\boldsymbol{r})$, and the means $p(\boldsymbol{\mu})$.

Dirichlet Prior on Mixture Weights. In the DEXUS model, the prior $p(\boldsymbol{\alpha})$ on the mixture weights $\boldsymbol{\alpha}$ should incorporate the prior knowledge that most transcripts are not differentially expressed into the model. The prior should represent the null hypothesis that the read counts are generated under only one condition. Such a prior enforces a low false discovery rate at the detection of differentially expressed transcripts because, for ambiguous data, read counts are not explained by differential expression.

A Dirichlet prior with parameters $\boldsymbol{\gamma}$ is well suited to represent this null hypothesis:

$$p(\boldsymbol{\alpha}) \;=\; \mathrm{D}(\boldsymbol{\alpha}; \boldsymbol{\gamma}) \;=\; b(\boldsymbol{\gamma}) \prod_{i=1}^{n} \alpha_i^{\gamma_i - 1}\;, \tag{S22}$$

where $\boldsymbol{\alpha}$ an the n-dimensional probability vector, i.e. $\alpha_1, \ldots, \alpha_n \geq 0$ and $\sum_{i=1}^{n} \alpha_i = 1$. Each component α_i is distributed according to a beta distribution with the following properties:

$$\mathrm{mean}(\alpha_i) \;=\; \frac{\gamma_i}{\gamma_s}\;, \tag{S23}$$

$$\mathrm{mode}(\alpha_i) \;=\; \frac{\gamma_i - 1}{\gamma_s - n}\,, \tag{S24}$$

$$\mathrm{var}(\alpha_i) \;=\; \frac{\gamma_i\,(\gamma_s - \gamma_i)}{\gamma_s^2\,(\gamma_s + 1)}\,, \tag{S25}$$

where we set

$$\gamma_s \;=\; \sum_{i=1}^{n} \gamma_i\,. \tag{S26}$$

To express our prior knowledge that most genes are not differentially expressed, we set $\gamma_1 \gg \gamma_i$ (for $i > 1$). This setting of the values ensures that the model tries to explain the data by one mixture component, that is a single negative binomial distribution. For $i \neq 1$ we set $\gamma_i = 1$ in order to enforce a mode at zero. Therefore for most drawn $\boldsymbol{\alpha}$, the component α_i is zero for $i \neq 1$. This reduces the number of hyperparameters to just one, which is γ_1.

We set the parameter $\boldsymbol{\gamma}$ of the Dirichlet prior to

$$\boldsymbol{\gamma} = (1 + G, 1, ..., 1). \tag{S27}$$

This simplifies the setting of the hyperparameters to one hyperparameter G. In all experiments we set $G = 1$.

Truncated Exponential Priors on the Size Parameter. The maximum likelihood solution r_{ML}, given in Eq. (S17), for the negative binomial tends to overestimate the true size parameter r (Piegorsch 1990). Therefore, we introduce a prior $p(r_i)$ on r_i for each condition i, which prefers small r_i-values. As prior for r_i we use an exponential distribution with parameter η:

$$p(r_i) \;=\; \mathrm{EXP}(r_i) \;=\; \eta\, e^{-\eta r_i}. \tag{S28}$$

Thus, the prior of $\boldsymbol{r}$ is $\mathrm{EXP}(\boldsymbol{r}) = \prod_{i=1}^{n} \mathrm{EXP}(r_i)$.

We truncate this exponential distribution at $r_{\max}$ in order to enforce a lower bound of $1/r_{\max}$ on the overdispersion. This bound is important to make the parameter space compact and, therefore, to ensure that the maximum likelihood estimator is consistent. Thus our prior is actually a truncated exponential distribution.

Furthermore, the bound ensures a certain minimal overdispersion for each gene which is another prior knowledge that we include for model selection. We follow Anders and Huber (2010) in their implementation of DESeq and set $r_{\max} = 13.0$. Truncating the exponential distribution changes the distribution only by a normalizing constant. In order to keep the notation uncluttered, we derive the algorithm without denoting this normalizing constant. Note that nothing changes in the derivation except that we have the constraint that $r_i < r_{\max}$ for all i.

Uniform Prior for the Mean Parameter. If, in one condition, all read counts are close to zero (the transcript is not present), the estimate of the mean of the negative binomial would not converge. The reason is that the parameter space is not compact as $\mu = 0$ is excluded. A compact parameter space is required to ensure consistency of the maximum likelihood estimator.

To make the parameter space compact, we introduce a lower bound $\mu_{\min}$ on μ_i. We implement this bound by a uniform prior on μ_i on the interval $[\mu_{\min}, \max_k x_k]$. In all experiments we used $\mu_{\min} = 0.5$.

Graphical Representation of the Model A graphical representation of the model including the parameters and hyperparameters is given in Figure S8.

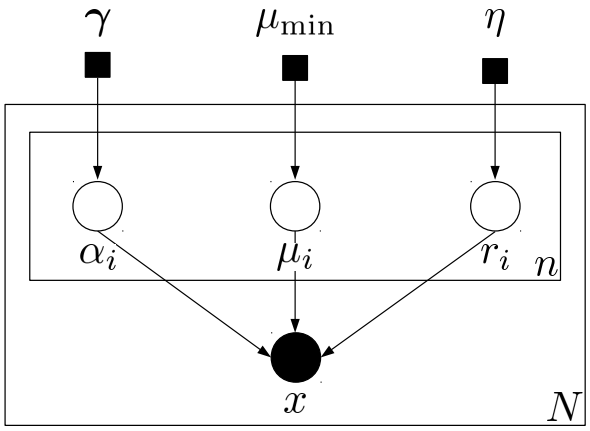

Figure S8: Graphical representation of the DEXUS model as directed acyclic graph. Squares represent hyperparameters, white circles model parameters and black circles given data.

Bound on the Posterior Using Priors and All Data. Using these priors, the upper bound in Eq. (S21) on the posterior becomes:

$$-\sum_{i=1}^{n} \hat{\alpha}_i \log\left(\alpha_i \,\mathrm{NB}(x;\, \mu_i, r_i)\right) - \log \mathrm{D}(\boldsymbol{\alpha}) - \log \mathrm{EXP}(\boldsymbol{r}) + \sum_{i=1}^{n} \hat{\alpha}_i \log \hat{\alpha}_i + \log(c(x))\,. \tag{S29}$$

The prior on $\boldsymbol{\mu}$ is constant and, therefore, is absorbed in $c(x)$. During the EM algorithm μ values smaller than $\mu_{\min}$ are projected back to $\mu_{\min}$, which corresponds to the maximum a posterior value given the uniform prior.

The likelihood for the whole data set $\{x_1, \ldots, x_N\}$ of N samples is the product of the likelihoods for data points x_k. Thus, the log likelihood is a sum over the log likelihoods for data points x_k. That means the inequality in Eq. (S21) can be applied to each single data point x_k, where $\hat{\alpha}_i$ is replaced by $\tilde{\alpha}_{ik}$. Therefore, the upper bound B on the scaled (by $\frac{1}{N}$) negative log-posterior for

a data set $\{x_1, \ldots, x_N\}$ is:

$$B \;=\; -\,\frac{1}{N}\sum_{k=1}^{N}\sum_{i=1}^{n}\tilde{\alpha}_{ik}\;\log\left(\alpha_i\,\mathrm{NB}(x_k;\;\mu_i, r_i)\right)\;-\;\frac{1}{N}\;\log \mathrm{D}(\boldsymbol{\alpha})\;-\;\frac{1}{N}\;\log \mathrm{EXP}(\boldsymbol{r}) \\ +\;\frac{1}{N}\sum_{k=1}^{N}\sum_{i=1}^{n}\tilde{\alpha}_{ik}\;\log\tilde{\alpha}_{ik}\;+\;\frac{1}{N}\sum_{k=1}^{N}\log(c(x_k))\,. \tag{S30}$$

In above formula, the log of the negative binomial probability mass function is

$$\log \mathrm{NB}(x;\;\mu_i, r_i) \;=\; \log((x + r_i - 1)!)\;-\;\log(x!)\;-\;\log((r_i-1)!) \\ +\;x\;\log\left(\frac{\mu_i}{\mu_i + r_i}\right)\;+\;r_i\;\log\left(\frac{r_i}{\mu_i + r_i}\right)\,. \tag{S31}$$

E-step: Optimization w.r.t. Posterior Estimates. For the E-step the upper bound B Eq. (S30) on the negative log posterior must be minimized with respect to $\tilde{\alpha}_{ik}$. The condition

$$\sum_{i=1}^{n}\tilde{\alpha}_{ik} \;=\; 1 \tag{S32}$$

must hold to ensure that the posterior mixture weights are an n-dimensional probability vector. The bound B ensures that the $\tilde{\alpha}_{ik}$ are positive via the $\log$.

The Lagrangian using only terms in $\tilde{\alpha}_{ik}$ is

$$L \;=\; -\,\frac{1}{N}\sum_{k=1}^{N}\sum_{i=1}^{n}\tilde{\alpha}_{ik}\;\log\left(\alpha_i\,\mathrm{NB}(x_k;\;\mu_i, r_i)\right) \\ +\;\frac{1}{N}\sum_{k=1}^{N}\sum_{i=1}^{n}\tilde{\alpha}_{ik}\;\log\tilde{\alpha}_{ik}\;-\;\lambda_k\;\left(\sum_{i=1}^{n}\tilde{\alpha}_{ik}\;-\;1\right)\,, \tag{S33}$$

where λ_k is the Lagrange multiplier for the k-th constraint given by Eq. (S32).

For the optimal value, the derivative of the Lagrangian L with respect to to $\tilde{\alpha}_{ik}$ must be zero:

$$\frac{\partial L}{\partial \tilde{\alpha}_{ik}} \;=\; -\frac{1}{N}\;\log\left(\alpha_i\,\mathrm{NB}(x_k;\;\mu_i, r_i)\right)\;+\;\frac{1}{N}\left(\log\tilde{\alpha}_{ik}\;+\;1\right)\;-\;\lambda_k\;=\;0\,. \tag{S34}$$

This equation can be solved for $\tilde{\alpha}_{ik}$:

$$\tilde{\alpha}_{ik} \;=\; \alpha_i\,\mathrm{NB}(x_k;\;\mu_i, r_i)\;e^{N\lambda_k - 1}\,. \tag{S35}$$

Exponentiation during solving the equation ensures positive $\tilde{\alpha}_{ik}$. Summing over i from 1 to n gives

$$e^{N\lambda_k - 1} \;=\; \frac{1}{\sum_{i=1}^{n}\alpha_i\,\mathrm{NB}(x_k;\;\mu_i, r_i)}\,. \tag{S36}$$

Inserting this equation into Eq. (S35) gives:

$$\tilde{\alpha}_{ik} \;=\; \frac{\alpha_i \,\mathrm{NB}(x_k;\; \mu_i, r_i)}{\sum_{i=1}^{n} \alpha_i \,\mathrm{NB}(x_k;\; \mu_i, r_i)} \,. \tag{S37}$$

Note that the optimal $\tilde{\alpha}_{ik}$ is the posterior of condition i in the mixture model given data point x_k. $\alpha_i = p(i)$ is the prior for condition i and $p(x_k \mid i) = \mathrm{NB}(x_k;\; \mu_i, r_i)$ is the likelihood for condition i, and $\tilde{\alpha}_{ik} = p(i \mid x_k)$ the posterior for condition i. α_i can be decomposed into α_{ik}:

$$\begin{aligned} \alpha_i \;&=\; p(i) \;=\; \sum_{x=0}^{\infty} p(i, x) \;=\; \sum_{x=0}^{\infty} p(i \mid x)\, p(x) \;=\; \mathrm{E}_{p(x)}\left(p(i \mid x)\right) \\ &\approx\; \frac{1}{N} \sum_{k=1}^{N} p(i \mid x_k) \;=\; \frac{1}{N} \sum_{k=1}^{N} \alpha_{ik} \,. \end{aligned} \tag{S38}$$

Therefore, we estimate α_i by $\hat{\alpha}_i$:

$$\hat{\alpha}_i \;=\; \frac{1}{N} \sum_{k=1}^{N} \tilde{\alpha}_{ik} \,. \tag{S39}$$

This estimate $\hat{\alpha}_i$ is used in the update rules below.

M-step: Optimization w.r.t. Mean. In the M-step, we minimize the upper bound B Eq. (S30) on the negative log posterior with respect to all parameters $\boldsymbol{\mu}$, $\boldsymbol{r}$, and $\boldsymbol{\alpha}$.

First we minimize B with respect to μ_i, where only terms depending on μ_i are considered:

$$\min_{\mu_i} \quad -\frac{1}{N} \sum_{k=1}^{N} \sum_{i=1}^{n} \tilde{\alpha}_{ik} \,\log\left(\alpha_i \mathrm{NB}(x_k;\; \mu_i, r_i)\right) \,. \tag{S40}$$

The derivative of the log negative binomial distribution Eq. (S31) with respect to μ_i is

$$\begin{aligned} \frac{\partial \log \mathrm{NB}(x_k;\; \mu_i, r_i)}{\partial \mu_i} \;&=\; x_k \left(\frac{\mu_i + r_i}{\mu_i}\right) \left(\frac{r_i}{(\mu_i + r_i)^2}\right) \;-\; r_i \left(\frac{\mu_i + r_i}{r_i}\right) \frac{r_i}{(\mu_i + r_i)^2} \\ &=\; x_k \left(\frac{r_i \;-\; \mu_i \, r_i}{\mu_i \;(\mu_i \;+\; r_i)}\right) \;=\; \left(\frac{x_k \;-\; \mu_i}{\mu_i \;+\; r_i^{-1} \mu_i^2}\right) \,. \end{aligned} \tag{S41}$$

The derivative of the upper bound B Eq. (S30) with respect to μ_i is

$$\frac{\partial B}{\partial \mu_i} \;=\; -\,\frac{1}{N} \sum_{k=1}^{N} \tilde{\alpha}_{ik} \left(\frac{x_k \;-\; \mu_i}{\mu_i \;+\; r_i^{-1}\, \mu_i^2}\right) \,. \tag{S42}$$

At the minimum, this derivative must be zero. Setting the derivative equal to zero and solving for μ_i gives the update rule

$$\mu_i \;=\; \frac{\sum_{k=1}^{N} \tilde{\alpha}_{ik}\, x_k}{N\, \hat{\alpha}_i} \,. \tag{S43}$$

The update is simply a weighted mean, where the weight $\tilde{\alpha}_{ik}$ is the posterior of condition i for data x_k. $\tilde{\alpha}_{ik}$ reflects how likely x_k was generated under condition i. Note that the update does not depend on other parameters. Since we have introduced a uniform prior of μ_i on the compact interval $[\mu_{\min}, \max_k x_k]$, μ_i that exceed this interval after being updated are projected back to it.

M-step: Optimization w.r.t. Size Parameter Secondly we minimize B with respect to r_i. Only terms of B that depend on r_i are considered:

$$\min_{r_i} \quad -\frac{1}{N}\sum_{k=1}^{N}\sum_{i=1}^{n}\tilde{\alpha}_{ik}\,\log\left(\alpha_i\,\mathrm{NB}(x_k;\,\mu_i,r_i)\right)\;-\;\frac{1}{N}\,\log\mathrm{EXP}(\boldsymbol{r}) \tag{S44}$$

The derivative of the log negative binomial Eq. (S31) with respect to r_i is:

$$\frac{\partial}{\partial r_i}\log\mathrm{NB}(x;\,\mu_i,r_i)\;=\;\psi(x+r_i)\;-\;\psi(r_i)\;-\;\frac{x-\mu_i}{\mu_i+r_i}\;-\;\log\left(\frac{r_i}{\mu_i+r_i}\right)\,, \tag{S45}$$

where $\psi(x)$ is the digamma function. The derivative of the log exponential with respect to r_i is

$$\frac{\partial}{\partial r_i}\log\mathrm{EXP}(\boldsymbol{r})\;=\;\frac{\partial}{\partial r_i}\log\left(\prod_{i=1}^{n}\eta e^{-\eta r_i}\right)\;=\;-\,\eta\,. \tag{S46}$$

The derivative of the upper bound B Eq. (S30) with respect to r_i is

$$\frac{\partial B}{\partial r_i}\;=\;-\,\frac{1}{N}\sum_{k=1}^{N}\left[\tilde{\alpha}_{ik}\,\psi(x+r_i)\;-\;\psi(r_i)\;-\;\frac{x_k-\mu_i}{\mu_i+r_i}\;-\;\log\left(\frac{r_i}{\mu_i+r_i}\right)\right]\;+\;\frac{1}{N}\,\eta\,. \tag{S47}$$

This derivative depends on the parameter μ_i, where we have to use the new value for μ_i: $\mu_i = (\sum_{k=1}^{N}\tilde{\alpha}_{ik}x_k)/(N\hat{\alpha}_i)$. The term $\sum_{k=1}^{N}\tilde{\alpha}_{ik}(x_k-\mu_i)/(\mu_i+r_i)$ is zero, because $\sum_{k=1}^{N}\tilde{\alpha}_{ik}x_k = N\hat{\alpha}_i\mu_i$ according to the μ_i update Eq. (S43) and $\mu_i\sum_{k=1}^{N}\tilde{\alpha}_{ik} = N\hat{\alpha}_i\mu_i$.

At the minimum this derivative must be zero, which leads to

$$\sum_{k=1}^{N}\tilde{\alpha}_{ik}\,\psi(x_k+r_i)\;-\;\psi(r_i)\sum_{k=1}^{N}\tilde{\alpha}_{ik}\;+\;\log\left(\frac{r_i}{\mu_i+r_i}\right)\sum_{k=1}^{N}\tilde{\alpha}_{ik}\;-\;\eta\;=\;0\,. \tag{S48}$$

Inserting the new value for μ_i into this equation results in

$$\sum_{k=1}^{N}\tilde{\alpha}_{ik}\,\psi(x_k+r_i)\;-\;N\,\hat{\alpha}_i\,\psi(r_i)\;+\;N\,\hat{\alpha}_i\,\log\left(\frac{r_i\,\hat{\alpha}_i}{\frac{1}{N}\,\sum_{k=1}^{N}\tilde{\alpha}_{ik}\,x_k\;+\;r_i\,\hat{\alpha}_i}\right)\;-\;\eta\;=\;0\,. \tag{S49}$$

This equation cannot be solved for r_i in a closed form. However, the parameter r_i can be obtained by solving this equation numerically. Since it is an equation in one variable, we use a simple bisection procedure.

Without a prior on r_i, the term $-\frac{1}{N}\log\mathrm{EXP}(\boldsymbol{r})$ vanishes in Eq. (S44) and the equation which must be solved for r_i becomes

$$\sum_{k=1}^{N}\tilde{\alpha}_{ik}\,\psi(x_k+r_i)\;-\;N\,\alpha_i\,\psi(r_i)\;+\;N\,\hat{\alpha}_i\,\log\left(\frac{r_i}{\mu_i+r_i}\right)\;=\;0\,. \tag{S50}$$

Only the term $-\eta$ vanishes in comparison to Eq. (S49), i.e., the equation obtained with the prior. Since we have introduced an exponential prior of r_i on the compact interval $[0, r_{\max}]$, r_i for which $r_i > r_{\max}$ holds after being updated are set equal to $r_{\max}$.

M-step: Optimization of Mixture Weights Thirdly we minimize B with respect to $\boldsymbol{\alpha}$ with the constraint that the α_i sum to 1. Only terms depending on α are considered:

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & -\frac{1}{N}\sum_{k=1}^{N}\sum_{i=1}^{n}\tilde{\alpha}_{ik}\,\log\alpha_i \;-\; \frac{1}{N}\,\log \mathrm{D}(\boldsymbol{\alpha}) \\ \text{s.t.} \quad & \sum_{i=1}^{n}\alpha_i \;=\; 1\,. \end{aligned} \tag{S51}$$

The objective ensures that $\alpha_i > 0$. The Lagrangian is

$$\begin{aligned} L \;&=\; -\frac{1}{N}\sum_{k=1}^{N}\sum_{i=1}^{n}\tilde{\alpha}_{ik}\,\log\alpha_i \;-\; \frac{1}{N}\,\log p(\boldsymbol{\alpha}) \;+\; \rho\,\left(\sum_{i=1}^{n}\alpha_i - 1\right) \\ &=\; -\frac{1}{N}\sum_{k=1}^{N}\sum_{i=1}^{n}\tilde{\alpha}_{ik}\,\log\alpha_i \;-\; \frac{1}{N}\sum_{i=1}^{n}(\gamma_i-1)\,\log\alpha_i \;+\; \rho\,\left(\sum_{i=1}^{n}\alpha_i \;-\; 1\right)\,, \end{aligned} \tag{S52}$$

where ρ is the Lagrange multiplier for the constraint. The solution requires that the derivative of L with respect to α_i is zero:

$$\frac{\partial L}{\partial \alpha_i} \;=\; -\frac{1}{N}\sum_{k=1}^{N}\tilde{\alpha}_{ik}\,\frac{1}{\alpha_i} \;-\; \frac{1}{N}\,(\gamma_i-1)\,\frac{1}{\alpha_i} \;+\; \rho \;=\; 0\,. \tag{S53}$$

Multiplying this equation by α_i gives

$$-\frac{1}{N}\sum_{k=1}^{N}\tilde{\alpha}_{ik} \;-\; \frac{1}{N}\,(\gamma_i-1) \;+\; \rho\,\alpha_i \;=\; 0\,. \tag{S54}$$

Summation over i leads to

$$1 \;+\; \frac{1}{N}\,(\gamma_s - n) \;=\; \rho\,, \tag{S55}$$

where $\gamma_s = \sum_i \gamma_i$. Inserting this expression for ρ into Eq. (S54) gives

$$-\frac{1}{N}\sum_{k=1}^{N}\tilde{\alpha}_{ik} \;-\; \frac{1}{N}\,(\gamma_i-1) \;+\; \left(1+\frac{1}{N}\,(\gamma_s-n)\right)\,\alpha_i \;=\; 0\,. \tag{S56}$$

Solving this equation for α_i using $\hat{\alpha}_i = \frac{1}{N}\sum_{k=1}^{N}\tilde{\alpha}_{ik}$ leads to the update formula for α_i:

$$\alpha_i \;=\; \frac{\hat{\alpha}_i \;+\; \frac{1}{N}\,(\gamma_i-1)}{1 \;+\; \frac{1}{N}\,(\gamma_s-n)}\,. \tag{S57}$$

S3.2.2 Variance-To-Mean Ratio Approaching One

The variance-to-mean ratio of negative binomials is bounded from below by one, since $\sigma^2/\mu = (\mu+\mu^2/r)/\mu = 1+\mu/r$. For data with variance-to-mean ratio smaller than one, the size parameter r increases continuously during the EM algorithm. For numerical stability of the algorithm,

we approximate the distribution of the negative binomial for large values of r with a Poisson distribution. As mentioned before we use a truncated exponential function as a prior on r for which $r < r_{\max}$ (default $r_{\max} = 13.0$). If the r-update leads to an r larger equal $r_{\max}$ and $r_{\max}$ is set to a value higher than 10,000, then we switch to the Poisson distribution for the according condition.

For $r \to \infty$, the negative binomial converges to a Poisson distribution:

$$\begin{aligned} \mathrm{NB}(x;\ \mu, r) \ &= \ \frac{\Gamma(x+r)}{\Gamma(x+1)\Gamma(r)} \left(\frac{r}{r+\mu}\right)^r \left(\frac{\mu}{r+\mu}\right)^x \\ &= \ \frac{\mu^x}{\Gamma(x+1)} \frac{\Gamma(x+r)}{\Gamma(r)(r+\mu)^x} \left(\frac{1}{1+\frac{\mu}{r}}\right)^r \end{aligned} \tag{S58}$$

$$\lim_{r\to\infty} \mathrm{NB}(x;\ \mu, r) \ = \ \lim_{r\to\infty} \underbrace{\frac{\mu^x}{\Gamma(x+1)}}_{\mu^x/x!} \underbrace{\frac{\Gamma(x+r)}{\Gamma(r)(r+\mu)^x}}_{1} \underbrace{\left(\frac{1}{1+\frac{\mu}{r}}\right)^r}_{e^{-\mu}} \ = \ \frac{\mu^x}{x!}\, e^{-\mu} \ = \ \mathrm{P}(x;\ \mu)\,, \tag{S59}$$

where $\mathrm{P}(x;\ \mu)$ is the Poisson probability mass function with parameter μ evaluated at x. Note, that $\Gamma(x+1) = x!$ for integer x.

S3.2.3 Update Rules

We summarize the update rules for the EM algorithm. The update rules are:

- posterior estimate

$$\hat{\alpha}_i \ = \ \frac{1}{N} \sum_{k=1}^{N} \tilde{\alpha}_{ik}\,, \tag{S60}$$

- $\boldsymbol{\mu}$ update

$$\begin{aligned} \mu_i^{\mathrm{temp}} \ &= \ \frac{\sum_{k=1}^{N} \tilde{\alpha}_{ik}\, x_k}{N\ \hat{\alpha}_i}, \\ \mu_i \ &= \max\{\mu_i^{\mathrm{temp}},\ \mu_{\min}\}. \end{aligned} \tag{S61}$$

- $\boldsymbol{r}$ update
  The numeric solution f the following equationor r_i^{temp} of the equation:

$$\begin{aligned} &\sum_{k=1}^{N} \tilde{\alpha}_{ik}\ \psi(x_k + r_i^{\mathrm{temp}}) \ - \ N\ \hat{\alpha}_i\ \psi(r_i^{\mathrm{temp}}) \ + \\ &\qquad + \ N\ \hat{\alpha}_i\ \log\left(\frac{r_i^{\mathrm{temp}}\ \hat{\alpha}_i}{\frac{1}{N}\ \sum_{k=1}^{N} \tilde{\alpha}_{ik}\, x_k \ + \ r_i^{\mathrm{temp}}\ \hat{\alpha}_i}\right) \ - \ \eta \ = \ 0\,, \end{aligned} \tag{S62}$$

  where ψ is the digamma function. We use a bisection procedure to find the r_i^{temp}. We then have to map r_i^{temp} to the allowed parameter space:

$$r_i \ = \min\{r_i^{\mathrm{temp}}, r_{\max}\}. \tag{S63}$$

- $\boldsymbol{\alpha}$ update

$$\alpha_i \;=\; \frac{\hat{\alpha}_i \;+\; \frac{1}{N}\,(\gamma_i - 1)}{1 \;+\; \frac{1}{N}\,(\gamma_s - n)}\,. \tag{S64}$$

S3.2.4 Initialization

First we apply a k-means clustering algorithm (Hartigan and Wong 1979) with n centers ten times to the log read count data (adding a pseudo-count of 0.01 to avoid undefined values).

From these ten results, we select the result with minimal ratio between within-cluster distances to between-cluster distances of the log read counts. For each cluster $i = 1, \ldots, n$ we calculate the maximum likelihood estimators for μ_i and r_i of a negative binomial distribution using only read counts that belong to this cluster. These estimators are given in Eq. (S16) and Eq. (S17). After this estimations, the overdispersion parameter r_i bounded by $r_i \leq r_{\max}$. The values α_i are initialized by $\alpha_i = 1/n$, which is n-dimensional probability vector with maximum entropy. This initialization does not prefer any condition. Note that initializing an α_i close to zero would clamp condition i to zero.

S3.2.5 Adjusting the Hyperparameter for the Size Parameter Prior

As mentioned in Subsection S3.1.2, for finite sample size N, neither the bias nor the variance of the maximum likelihood estimator r_{ML} exists, because for data whose mean exceeds the variance r tends to infinity (Anscombe 1950). We empirically calculate a conditional bias, which is the bias under the condition that the mean is larger than the variance for the data set. For 10,000 experiments, we draw counts from a negative binomial distribution, removed experiments with mean larger than the variance, and computed the estimator r_{ML} for each experiment. Fig. S9 shows that the maximum likelihood estimator r_{ML} overestimates the true r for a small number of samples. Further it is shown that our truncated exponential prior on r reduces the effect of the overestimation for the maximum a posterior estimator. Both the bias and the variance of the maximum a posterior estimator is smaller than for the maximum likelihood estimator. The data underlying Fig. S9 were generated by drawing each r a normal distribution with mean 1.0 and standard deviation 0.1. Using this r and $\mu = 20$ for the parameters of a negative binomial distribution, five data points were drawn. With these five data points, for r the maximum likelihood estimator and the maximum a posterior estimator with $\eta = 0.8$ were calculated.

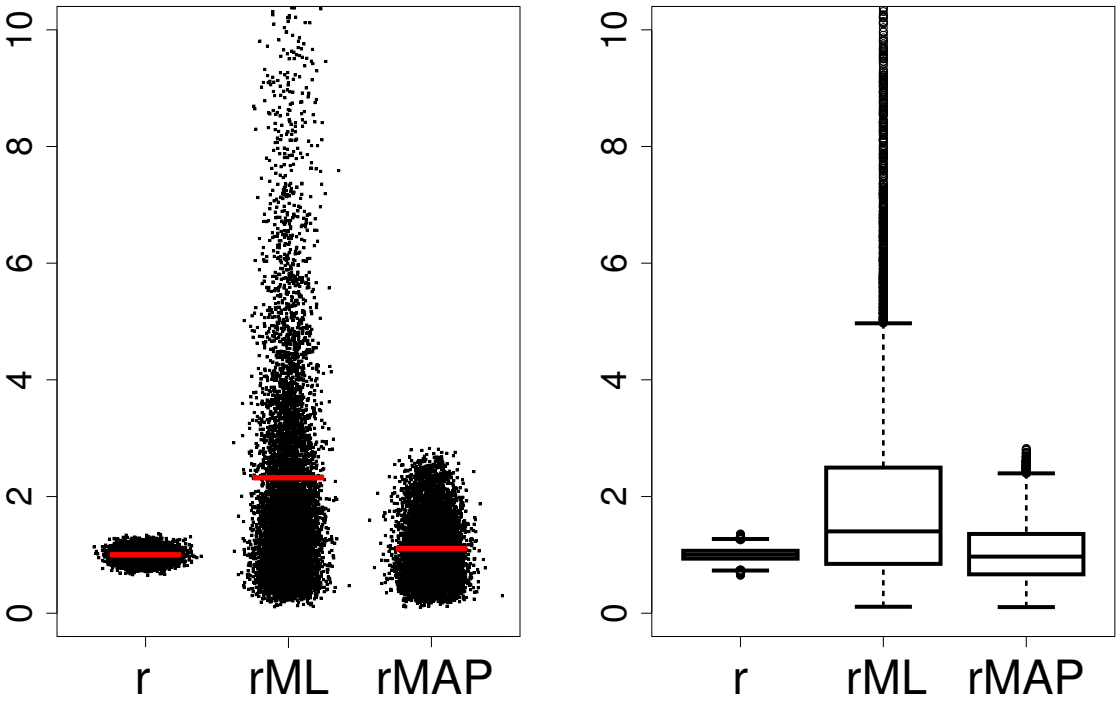

Figure S9: The bias of the maximum likelihood estimator r_{ML}. *Left panel:* In the left column "r" the true size parameters r are plotted. The middle column "rML" shows the maximum likelihood estimators for r. The right column "rMAP" gives the maximum a posterior estimators using our exponential prior on r. Red lines depict the mean of the according (estimated) parameters. *Right panel:* The same data as in the left panel but now presented as as boxplots. The maximum likelihood estimator overestimates the true r. Both the bias and the variance of the maximum a posterior estimator is smaller than for the maximum likelihood estimator.

In Subsection S3.2.1 we introduced a truncated exponential function with hyperparameter η as a prior on the size parameter r. This prior prefers small r and countermands the bias of the maximum likelihood estimator r_{ML} if using the maximum a posterior estimate. Note, that smaller estimates of r also reduce the variance of the estimator because r is bounded from below by zero. Thus, by adjusting η we can decrease both the bias and the variance of the maximum a posterior estimator, and hence the mean squared error (MSE). We analyzed the effect of different values for η for a large variety of values for μ, r, and number of samples N on the MSE, bias, and variance of the maximum a posterior estimator. Fig. S10 presents for the maximum a posterior estimator the mean over 10,000 experiments of the MSE, bias, and variance for different values of η. A data set of $N = 10$ data points is drawn from a negative binomial with parameters $\mu = 50$ and $r = 0.8$. The variance of the estimator decreases with increasing η, because of the lower bound at zero. The squared bias is minimal at $\eta = 1.6$. The MSE is minimal at 2.9.

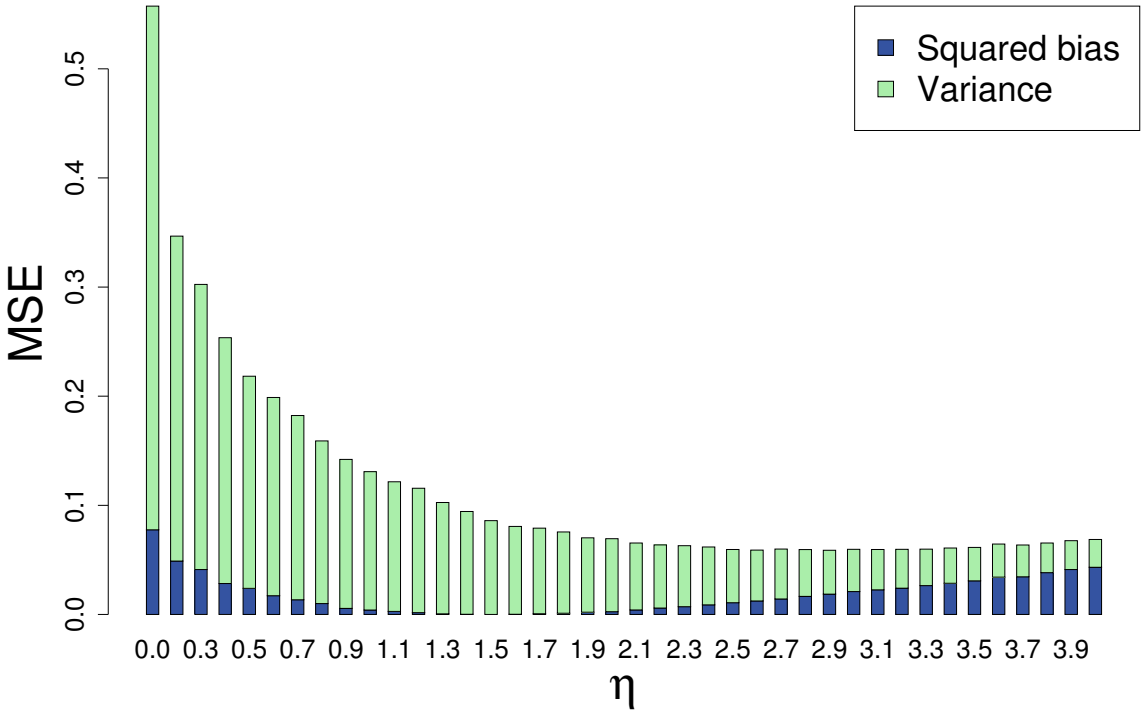

Figure S10: The mean squared error (MSE) of the maximum a posterior estimator for r. The hyperparameter η (x-axis) is plotted against the MSE (y-axis). Each box shows means of 10,000 experiments. One experiment consists of $N = 10$ data points, that were drawn from a negative binomial distribution with $r = 0.8$ and $\mu = 50$. The variance of the estimator decreases with with increasing η, whereas the squared bias is minimal at $\eta = 1.6$. The MSE is minimal at 2.9.

With increasing mean μ, both the variance and the mean squared error (MSE) of the maximum likelihood estimator decrease, as can be seen in Fig S11 for a mean over 10,000 experiments. One experiment of $N = 10$ data points is drawn from a negative binomial with $\mu = (4, 8, 16, \ldots, 40)$ and $r = 0.8$. Thus, larger μ needs less regularization by the prior to obtain a minimal MSE for the maximum posterior estimate. Therefore, η is selected depending on the μ of the particular gene or transcript that is analyzed. We compute an optimal η for each transcript using only a single hyperparameter θ for a data set. η is computed for each transcript with read counts x_i from θ by:

$$\eta \;=\; \frac{\theta}{1 + \mu_{\mathrm{ML}}}\,, \tag{S65}$$

where $\mu_{\mathrm{ML}} = 1/N \sum_{i=1}^{N} x_i$ is the mean read count for the transcript that is analyzed. μ_{ML} is the maximum likelihood estimator for μ of a negative binomial according to Eq. (S16).

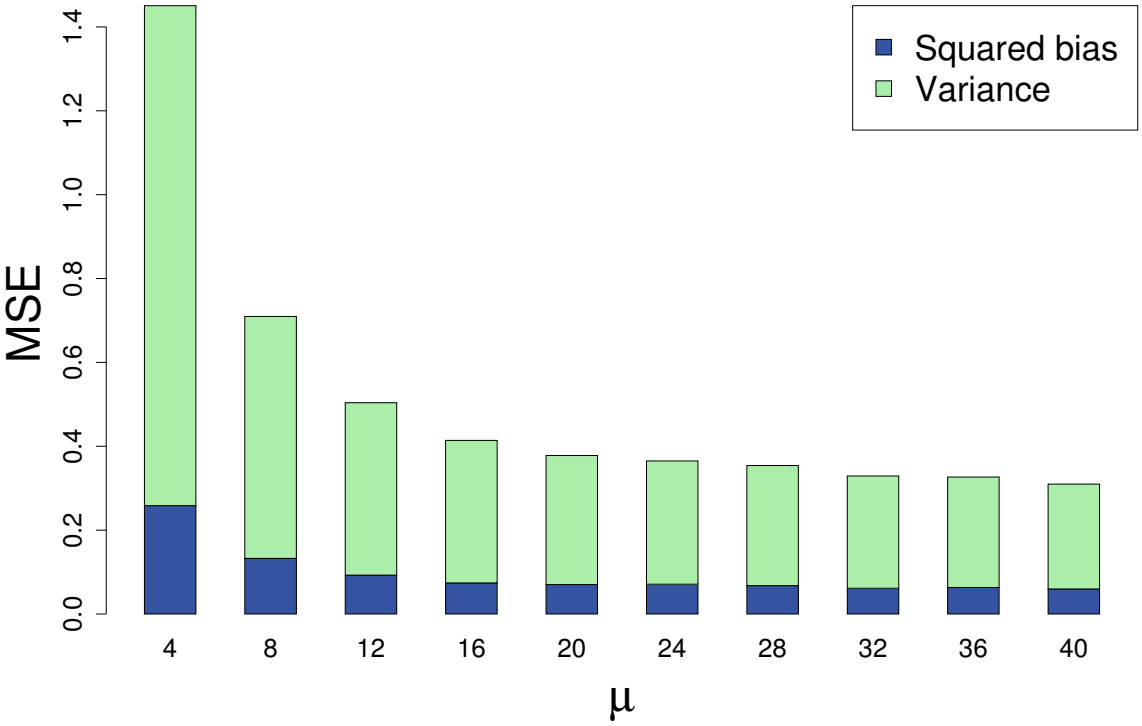

Figure S11: The mean squared error (MSE) of the maximum likelihood estimator r_{ML} for different means μ. The mean μ of the negative binomial distribution (x-axis) is plotted against the MSE (y-axis). Each box shows means of 10,000 experiments. One experiment consists of $N = 10$ data points, that were drawn from a negative binomial distribution with $r = 0.8$ and μ varying from 4 to 40. Both the mean squared error and the variance of the maximum likelihood estimator decrease with increasing μ. Thus, larger μ need less regularization by the prior to obtain an optimal maximum a posterior estimate.

S3.3 Calling Differentially Expressed Transcripts and I/NI Call

We suggest DEXUS for identifying differentially expressed transcripts in RNA-Seq data with unknown condition. In the E-step of the EM algorithm, α_{ik} estimates under which condition i read count x_k of a particular transcript was obtained.

In Subsection S3.2.1 in paragraph "E-step: Optimization w.r.t. Posterior Estimates" we noted that $\tilde{\alpha}_{ik}$ is the posterior of condition i in the mixture model given data point x_k. $\alpha_i = p(i)$ is the prior for condition i, $p(x_k \mid i) = \mathrm{NB}(x_k;\ \mu_i, r_i)$ is the likelihood for condition i, and $\tilde{\alpha}_{ik} = p(i \mid x_k)$ the posterior for condition i. According to the Bayes formula the posterior is

$$\alpha_{ik} \ = \ \frac{\alpha_i \ \mathrm{NB}(x_k;\ p_i, r_i)}{\sum_{i=1}^{n} \alpha_i \ \mathrm{NB}(x_k;\ p_i, r_i)} \ . \tag{S66}$$

In the Bayes interpretation, the prior $\alpha_i = p(i)$ gives the probability of drawing from condition i without seeing any data, while the posterior $\tilde{\alpha}_{ik} = p(i \mid x_k)$ is the probability of x_k being drawn from condition i. This means that the prior probability of the condition under which a read count is drawn (without seeing the read count) changes to the posterior probability after having observed the read count.

The posteriors $\tilde{\alpha}_{ik}$ are important to decide whether any two read counts are generated under the same or under different conditions. If any two read counts are generated under different conditions with different read count distributions, the according transcript is differentially expressed.

In the following, we have to distinguish between two cases: (i) data with known conditions and (ii) data with unknown conditions. For (i) data with known conditions, the $\tilde{\alpha}_{ik}$ are given. However the transcript may have the same read count distribution under the different conditions. To decide whether a transcript is differentially expressed in different given conditions, we have to determine whether read counts of different conditions arise from the same or from a different distribution. For (ii) data with unknown conditions, the EM algorithm ensures that different conditions have different read count distributions. The likelihood that a transcript is differentially expressed increases both with the likelihood that at least two conditions are observed and with the distance between the read count distributions of the conditions.

In the following subsections we consider first the case (i) data with known conditions and then case (ii) data with unknown conditions.

S3.3.1 Data with Known Conditions

For data with known conditions, the condition under which the read count x_k was generated is known. Therefore the α_{ik} values are binary: α_{ik} is one if x_k is generated under the i-th condition and zero otherwise.

$$\alpha_{ik} = \begin{cases} 1 & \text{if } x_k \text{ is drawn under condition } i \\ 0 & \text{if } x_k \text{ is not drawn under condition } i \end{cases} . \tag{S67}$$

The update rules in Subsection S3.2.3 simplify to the maximum likelihood estimators Eq. (S16) and Eq. (S17) from Subsection S3.1.2 for each condition. The regularization parameter η can be used to determine a maximum a posterior estimate for r.

Two conditions: An exact test for differential expression between two conditions. We use the test suggested by Robinson and Smyth (2008) and Anders and Huber (2010), which is implemented in the R package DESeq. We use the function `nbinomTestForMatrices` of the R package DESeq. It is a test of the null hypothesis that the means of read count distributions for the two conditions 1 and 2 are equal. Like Fisher's exact test, this test is a conditional test with the condition that the sum of all read counts has a particular value. We have N read counts x_k of which the N_1 read counts $x_1, \ldots, x_{N_1}$ are generated under condition 1 and the $N_2 = N - N_1$ read counts $x_{N_1+1}, \ldots, x_N$ are generated under condition 2. The test assumes that read counts of condition 1 are distributed according to a negative binomial $\mathrm{NB}(x;\ \mu_1, r_1)$ and read counts in condition 2 according to $\mathrm{NB}(x;\ \mu_2, r_2)$. The sum $S_1 = \sum_{k=1}^{N_1} x_k$ of N_1 read counts drawn from $\mathrm{NB}(x;\ \mu_1, r_1)$ is distributed according to $\mathrm{NB}(S_1;\ N_1\mu_1, N_1 r_1)$ (Bean 2001; Furman 2007). Analogously, the sum $S_2 = \sum_{k=N_1+1}^{N} x_k$ of N_2 read counts drawn from $\mathrm{NB}(x;\ \mu_2, r_2)$ is distributed according to $\mathrm{NB}(S_2;\ N_2\mu_2, N_2 r_2)$. The null hypothesis is that the mean μ_1 in the first condition is equal to the mean μ_2 in the second condition: $\mu = \mu_1 = \mu_2$. Using all $N = N_1 + N_2$ read counts x_k, the mean μ is estimated by $\mu = \frac{1}{N}\sum_{k=1}^{N} x_k$. Next, assuming $\mu_1 = \mu_2 = \mu$, the

values for r_1 and r_2 are estimated. Under the null hypothesis and with mutually independent read counts the probability p_S of observing the pair of sums (S_1, S_2) with N_1 summands in S_1 and N_2 summands in S_2 is:

$$p_S(S_1, S_2) \;=\; \mathrm{NB}(S_1;\; N_1\mu, N_1 r_1)\, \mathrm{NB}(S_2;\; N_2\mu, N_2 r_2)\;. \tag{S68}$$

Next we compute the probability of observing (S_1, S_2) or more extreme sum pairs (a, b) under the condition that $a + b = S$ with N_1 summands in a and N_2 summands in b. Further we assume that $a \sim \mathrm{NB}(x;\; N_1\mu, N_1 r_1)$ and that $b \sim \mathrm{NB}(x;\; N_2\mu, N_2 r_2)$. The probability of observing (S_1, S_2) or more extreme sum pairs (a, b) is:

$$p\left((S_1, S_2) \preceq (a, b) \mid a + b = S\right) \;=\; \frac{p\left((S_1, S_2) \preceq (a, b), a + b = S\right)}{p(a + b = S)}\;, \tag{S69}$$

where $(S_1, S_2) \preceq (a, b)$ means that (a, b) is equal or more extreme than (S_1, S_2). If $(S_1, S_2) \preceq (a, b) \Leftrightarrow p_s(a, b) \leq p_s(S_1, S_2)$ then the p-value can be calculated by:

$$p \;=\; \frac{\sum\limits_{a+b=S; p_s(a,b) \leq p_s(S_1,S_2)} p_S(a, b)}{\sum\limits_{a+b=S} p_S(a, b)}\;. \tag{S70}$$

Multiple conditions: Generalized Linear Model. For *multiple known conditions* we follow McCarthy *et al.* (2012) and fit a generalized linear model (GLM, Nelder and Wedderburn (1972)) for a negative binomial response using the logarithm as link function and the estimated dispersion parameters.

The GLM allows specifying any design and test for the significance of covariates. Without specifying a particular design, DEXUS will use a design that includes a covariate for each specified condition and compare it with a null hypothesis model that only includes an intercept term. The p-value from this comparison is used to rank transcripts according to the evidence for differential expression.

S3.3.2 Data with Unknown Conditions: I/NI Call

The Bayesian framework allows defining an informative/non-informative (I/NI) call (Hochreiter *et al.* 2006; Talloen *et al.* 2007, 2010; Clevert *et al.* 2011; Klambauer *et al.* 2012). An I/NI call reduces the false discovery rate at detecting differentially expressed transcripts because only those transcripts are called for which the evidence of being differentially expressed is high. DEXUS first computes the I/NI value (an evidence value) for differential expression. Subsequently, transcripts are called informative if the I/NI value is beyond a threshold.

In contrast to ϕ_i or r_i, which capture noise variation, $\boldsymbol{\alpha}$ captures variation arising from differentially expressed transcripts. Therefore, the posterior $\hat{\boldsymbol{\alpha}}$ of $\boldsymbol{\alpha}$ indicates differential expression in the data if at least two conditions have a probability larger than zero. First, we want to have evidence that at least two conditions are present. The larger the posterior value of a condition, the more read counts have this condition, the higher is the evidence that this condition was present for at least one read count. The model may explain one true condition by two model conditions and

differential expression would be falsely detected. Secondly, we want to have evidence that model conditions are different. The more the means μ_i of conditions differ (the means of the associated negative binomials), the higher is the evidence that these conditions are indeed different and the transcript is differentially expressed. Thus, the evidence on differential expression (the I/NI value) should consider two factors: (I) at least two non-zero posterior values for α_i, where larger values have more evidence that the conditions were indeed present; (II) differences of the means μ_i, where larger differences have more evidence that the conditions are indeed different.

We select the largest α_i (the condition with largest probability), and assume without loss of generality that this is the first condition ($i = 1$). Then we compare other conditions to the first condition. Factor (II), the differences between means, is expressed by the log differences $|\log(\mu_i) - \log(\mu_1)|$. Factor (I), two large non-zero posterior values, is included by weighting these differences by α_i. Thus, the I/NI value is

$$\begin{aligned} \mathrm{I/NI}(\boldsymbol{\alpha}, \boldsymbol{\mu}) &= \sum_{i=1}^{n} \alpha_i \left| \log\left(\frac{\mu_i}{\mu_1}\right) \right| \\ &= \sum_{i=1}^{n} \alpha_i \ |\log(\mu_i) - \log(\mu_1)| \ . \end{aligned} \tag{S71}$$

The I/NI value is the expected fold change of read counts relative to read counts of the most prominent condition given a noise-free model (all read counts are equal to the mean of the according condition). Another interpretation of the I/NI value is: "the information gain of the posterior $\hat{\boldsymbol{\alpha}}$ compared to the prior distribution $p(\boldsymbol{\alpha})$". The prior represents the null hypothesis that only one condition is present and the transcript is not differentially expressed. Therefore, the I/NI call measures the tendency to reject the null hypothesis based on the observed data.

The I/NI value can also be viewed as the distance between a multiple component model (differential expression) and the its closest single component model (no differential expression). The I/NI value is a distance measure between a model with parameters $M_1 = (\alpha_1^1, \ldots, \alpha_n^1, \mu_1^1, \ldots, \mu_n^1)$ and another model with $M_2 = (\alpha_1^2, \ldots, \alpha_n^2, \mu_1^2, \ldots, \mu_n^2)$:

$$\mathrm{d}(M_1, M_2) = \sum_{i=1}^{n} |\alpha_i^1 - \alpha_i^2| \, |\log \frac{\mu_i^1}{\mu_i^2}|. \tag{S72}$$

In the DEXUS method, the I/NI value measures the distance between the selected multiple component model with parameters $M = (\alpha_1, \ldots, \alpha_n, \mu_1, \ldots, \mu_n)$ to the closest single component model with parameters $M_0 = (1, 0, \ldots, 0, \mu_1, \mu_1, \ldots, \mu_1)$:

$$\mathrm{d}(M, M_0) = |\alpha_i - 1| \, |\log \frac{\mu_1}{\mu_1}| + \sum_{i=2}^{n} |\alpha_i - 0| \, |\log \frac{\mu_i}{\mu_1}| = \mathrm{I/NI}(\boldsymbol{\alpha}, \boldsymbol{\mu}). \tag{S73}$$

Note that other models than M_0 lead to larger distances to M.

S3.4 Sensitivity Analysis of the Hyperparameter for the Dirichlet Prior

For investigating the sensitivity of the hyperparameter G, that we introduced in Eq. (S27), we applied DEXUS to simulated data sets with unknown conditions (see Section S4.3). These data sets

were produced assuming three different library sizes (10^6,10^7, and 10^8), eight different settings with respect to the unkown conditions ($6/6$, $12/12$, $9/3$, $18/6$, $10/2$, $20/4$, $11/1$, and $22/2$). In total we had $3 \times 8 \times 100 = 2,400$ data sets and we used five different settings for G, that is $G = 0.1$, $G = 0.5$, $G = 1$, $G = 5$, and $G = 10$. We assess the average performance of DEXUS for different hyperparameters G in terms of the area under ROC curve ($\mathrm{AUC}_{\mathrm{ROC}}$) and the area under precision-recall curve ($\mathrm{AUC}_{\mathrm{PR}}$). The $\mathrm{AUC}_{\mathrm{ROC}}$ is determined by the ranking of the I/NI values, therefore it measures implicitely how much the I/NI value ranking change if different values for the hyperparameter G are used. Figures S12, S13, and S14 report the performance in terms of $\mathrm{AUC}_{\mathrm{ROC}}$ and $\mathrm{AUC}_{\mathrm{PR}}$ for the eight different settings and the three library sizes. The figures show that the performance (therefore the I/NI value ranking) is relatively insensitive to the setting of the hyperparameter G. For data sets with few samples in one of the conditions, $G > 1$ performs better with respect to $\mathrm{AUC}_{\mathrm{PR}}$. The improvement results from the reduced pressure on the minor condition towards zero condition weight — otherwise the condition would die out. Note, that in these settings only highly unbalanced number of samples in the conditions are present. However this is a quite unusual case in biological or medical studies. Only for very extreme cases, e.g. only one sample in one of the conditions, the performance improvement over $G = 1$ is notable. For data sets with the same number of samples in each condition, $G < 1$ performs better. However the performance improvement compared to $G = 1$ is minor. In conclusion, $G = 1$ has quite good performance for typical biological or medical studies.

Since the conditions and how many samples are in a condition are not known a priori, we average the performance over the different settings (number of samples in a condition). Tables S1 and S2 show the average performance of the model for different hyperparameters G in terms of the area under ROC curve ($\mathrm{AUC}_{\mathrm{ROC}}$) and the area under precision-recall curve ($\mathrm{AUC}_{\mathrm{PR}}$). The default value $G = 1$ is the best compromise for the different settings and gives for most library sizes the best average performance. The tables show that the performance is not very sensitive with respect to the value of G. A larger or smaller value of G than 1, leads for some settings to an performance increase but for other settings to a decrease which average out. This averaging out is also to be expected for real data sets, in which different settings are assumed to be present simultaneously.

library size	$G = 0.1$	$G = 0.5$	$G = 1$	$G = 5$	$G = 10$
10^6	0.74 ± 0.03	0.75 ± 0.02	0.75 ± 0.01	0.74 ± 0.02	0.73 ± 0.02
10^7	0.79 ± 0.05	0.80 ± 0.03	0.83 ± 0.02	0.77 ± 0.01	0.76 ± 0.01
10^8	0.89 ± 0.06	0.90 ± 0.03	0.91 ± 0.02	0.86 ± 0.02	0.85 ± 0.02

Table S1: The performance in terms of area under ROC curve ($\mathrm{AUC}_{\mathrm{ROC}}$) for three different library sizes and different choices of the hyperparameter G. The displayed values are the means over 800 data sets, that is 100 data sets for each of the eight different settings for the number of replicates in the conditions.

Though $G = 1$ is our recommendation, we still offer rules to set G for users who want to find better values of G:

- If conditions with few samples should be detected, large values of G, like $G = 5$ or $G = 10$, improve the $\mathrm{AUC}_{\mathrm{PR}}$ of DEXUS.

library size	$G = 0.1$	$G = 0.5$	$G = 1$	$G = 5$	$G = 10$
10^6	0.51 ± 0.08	0.54 ± 0.06	0.55 ± 0.05	0.56 ± 0.03	0.55 ± 0.02
10^7	0.62 ± 0.12	0.67 ± 0.09	0.70 ± 0.07	0.69 ± 0.03	0.68 ± 0.03
10^8	0.74 ± 0.15	0.80 ± 0.11	0.82 ± 0.08	0.81 ± 0.04	0.80 ± 0.03

Table S2: The performance in terms of area under precision-recall curve ($\mathrm{AUC}_{\mathrm{PR}}$) for three different library sizes and different choices of the hyperparameter G. The displayed values are the means over 800 data sets, that is 100 data sets for each of the eight different settings for the number of replicates in the conditions.

- For condition with equal number of samples, small G like $G = 0.1$ gives slightly better results than $G = 1$.
- If the number of samples in the conditions are unbalanced but not very extreme, values of G around one (the default) supply good performance.

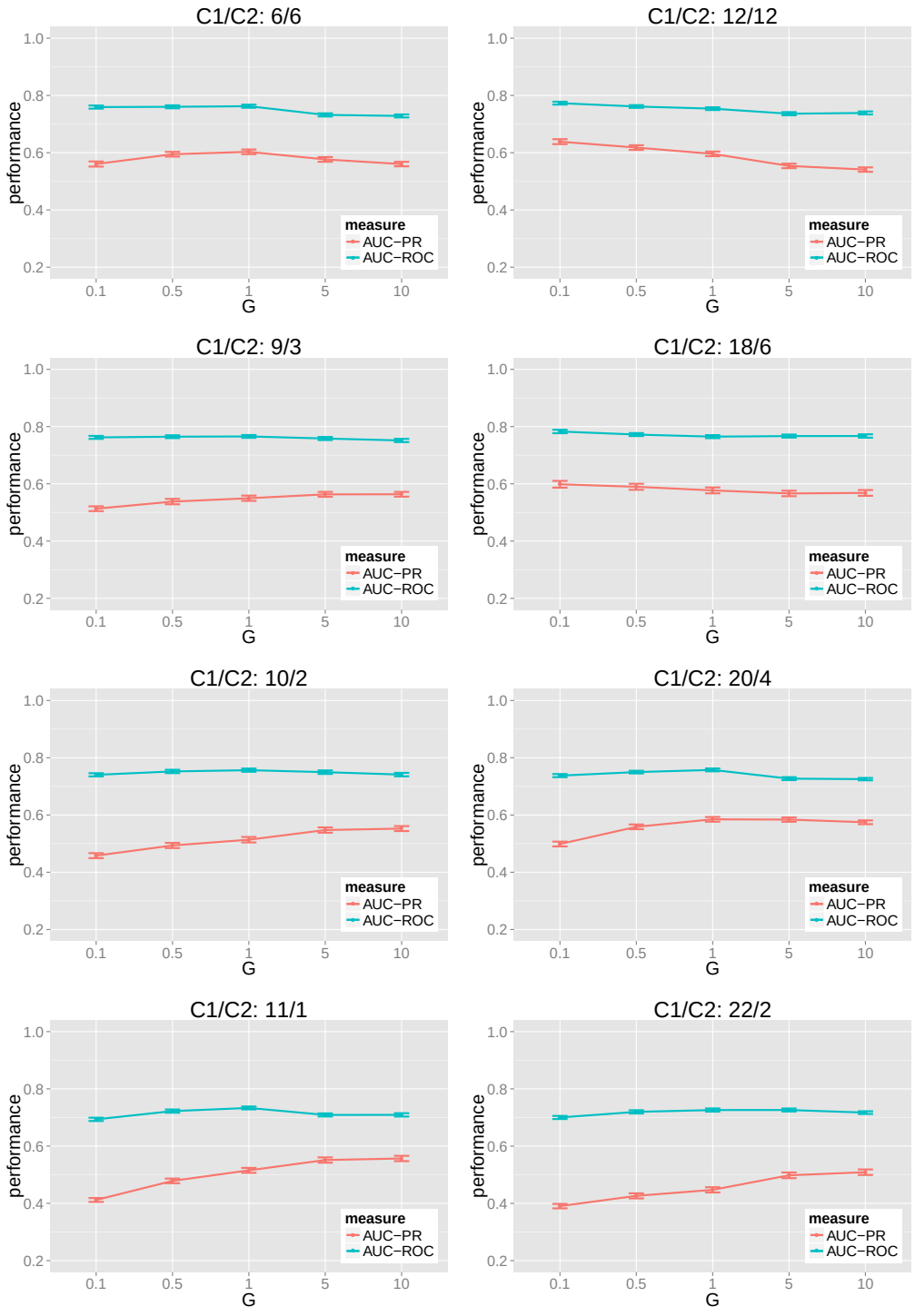

Figure S12: The performance of DEXUS in terms of $\mathrm{AUC}_{\mathrm{ROC}}$ and $\mathrm{AUC}_{\mathrm{PR}}$ for different values of the hyperparameter G. The library size is 10^6. The panels show the results for different number of replicates in the conditions displayed above the panel: $6/6$, $12/12$, $9/3$, $18/6$, $10/2$, $20/4$, $11/1$, and $22/2$. The $\mathrm{AUC}_{\mathrm{ROC}}$ and $\mathrm{AUC}_{\mathrm{PR}}$ are plotted against G values (x-axis). Each data point has an error bar that represents the standard deviation of the performance on 100 data sets.

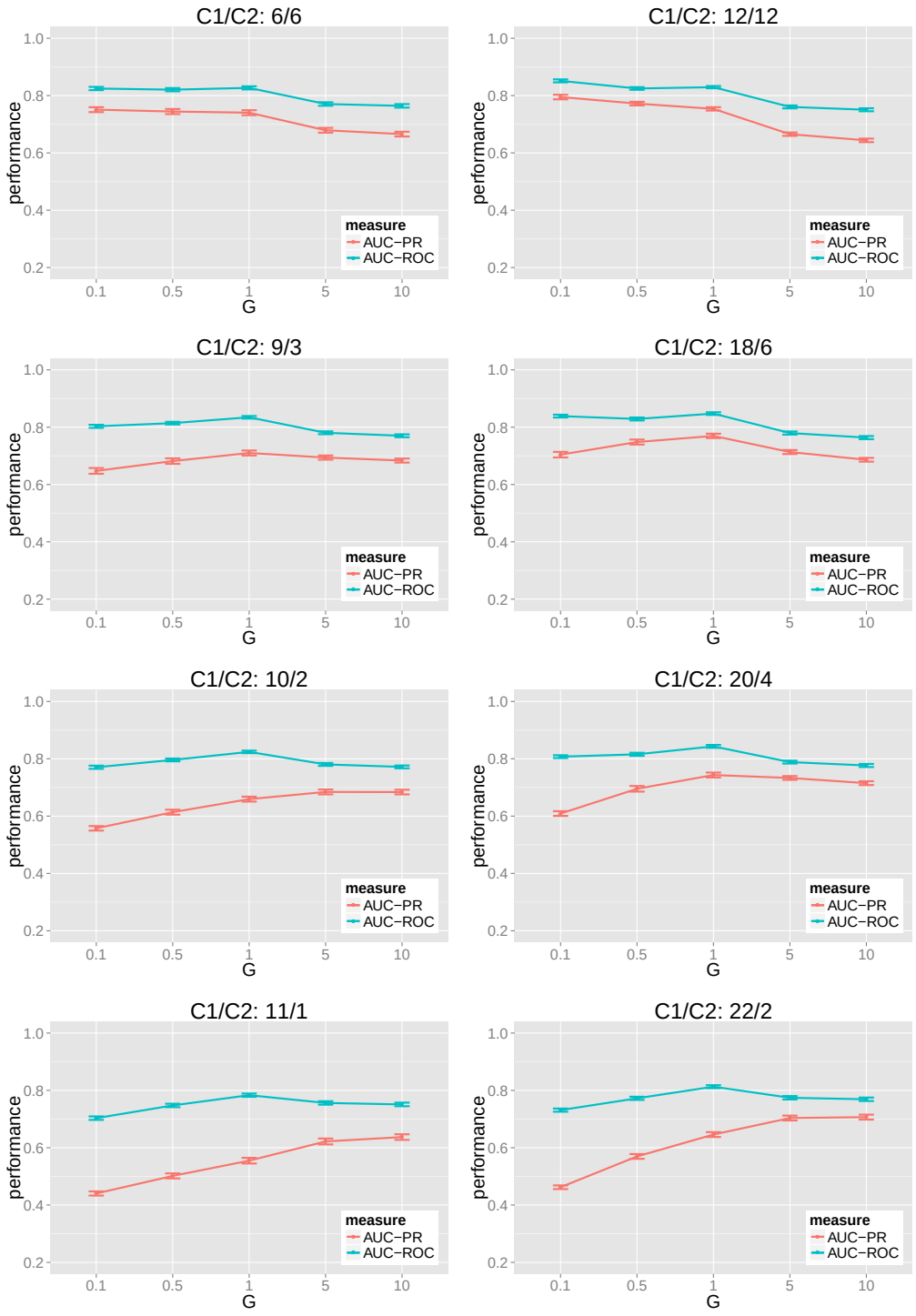

Figure S13: The performance of DEXUS in terms of $\mathrm{AUC}_{\mathrm{ROC}}$ and $\mathrm{AUC}_{\mathrm{PR}}$ for different values of the hyperparameter G. The library size is 10^7. The panels show the results for different number of replicates in the conditions displayed above the panel: $6/6$, $12/12$, $9/3$, $18/6$, $10/2$, $20/4$, $11/1$, and $22/2$. The $\mathrm{AUC}_{\mathrm{ROC}}$ and $\mathrm{AUC}_{\mathrm{PR}}$ are plotted against G values (x-axis). Each data point has an error bar that represents the standard deviation of the performance on 100 data sets.

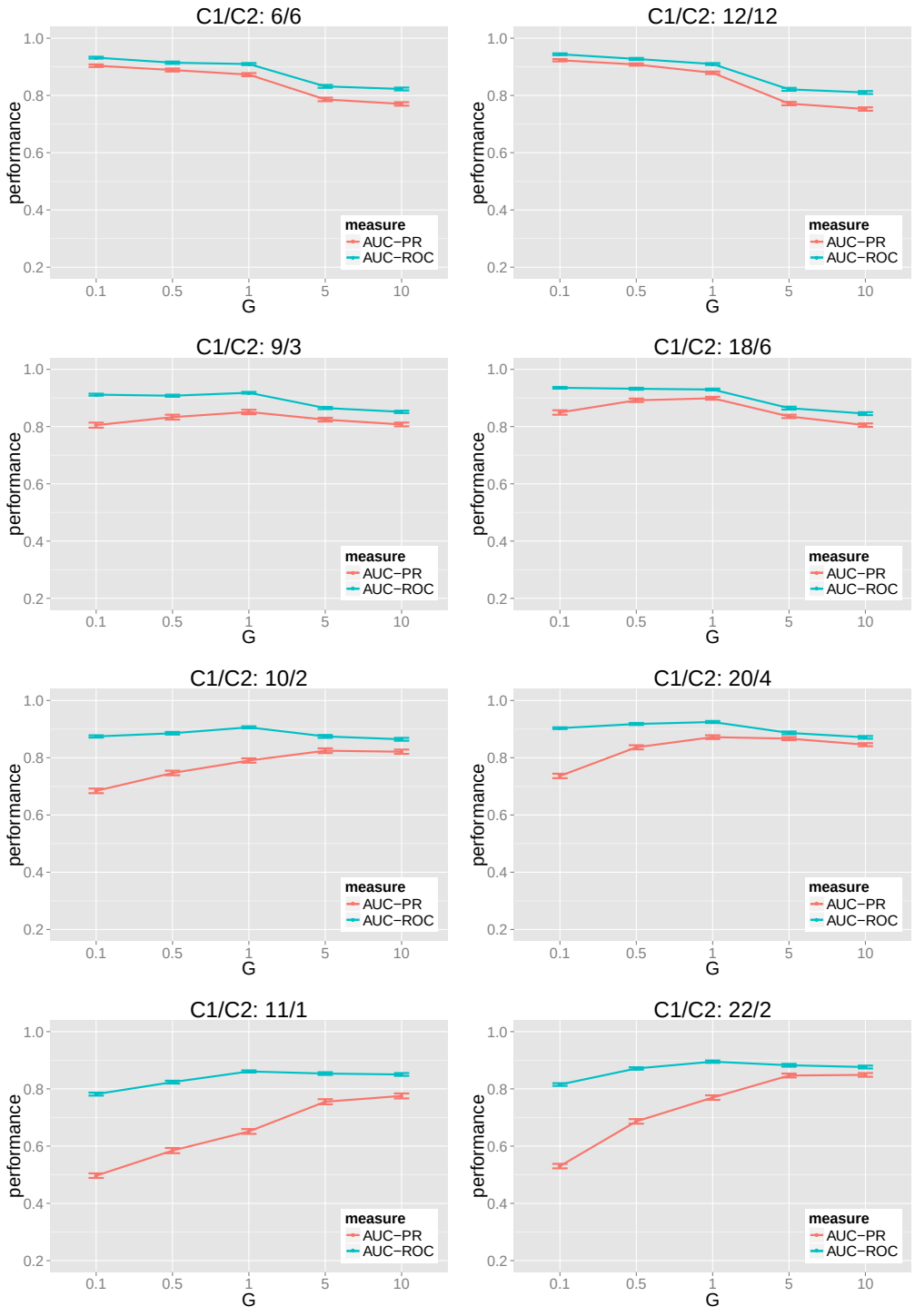

Figure S14: The performance of DEXUS in terms of $\mathrm{AUC}_{\mathrm{ROC}}$ and $\mathrm{AUC}_{\mathrm{PR}}$ for different values of the hyperparameter G. The library size is 10^8. The panels show the results for different number of replicates in the conditions displayed above the panel: $6/6$, $12/12$, $9/3$, $18/6$, $10/2$, $20/4$, $11/1$, and $22/2$. The $\mathrm{AUC}_{\mathrm{ROC}}$ and $\mathrm{AUC}_{\mathrm{PR}}$ are plotted against G values (x-axis). Each data point has an error bar that represents the standard deviation of the performance on 100 data sets.

S4 Experiments

S4.1 Evaluation Criteria for Simulated Data Sets

For simulated data sets with known ground truth, we formulate the detection of differential expression as a classification task. A method has to decide whether a gene or a transcript is differentially expressed (positive prediction) or not (negative prediction). For simulated data we know which genes are differentially expressed (the positives) and which are not (the negatives). Therefore, we can determine true positives, false positives, true negatives, and false negatives. The methods return a continuous value, like a p-value obtained from a test or the I/NI values with DEXUS, together with a threshold for deciding whether the gene is differentially expressed or not. This value allows to rank genes and to compute the receiver-operator characteristics (ROC), a standard measure to evaluate classification results. The area under the ROC curve ($\mathrm{AUC}_{\mathrm{ROC}}$) is a well-known classification performance criterion and is equivalent to a Mann-Whitney-Wilcoxon test of ranks.

Usually the number of differentially expressed genes is much smaller than the number of non-differentially expressed genes. For these unbalanced classes, i.e. one class is much larger then the other, the area under the precision recall curve ($\mathrm{AUC}_{\mathrm{PR}}$) is more appropriate as performance criterion, because it is independent of the true negatives. We report both the $\mathrm{AUC}_{\mathrm{ROC}}$ and the $\mathrm{AUC}_{\mathrm{PR}}$.

S4.2 RNA-Seq Data with Known Conditions

S4.2.1 Methods Compared

We compare the following methods (available as R packages) for differential expression in RNA-Seq data:

- DEXUS (our novel method using known conditions, see Section S3.3.1)
- DSS 1.0.0 (Wu *et al.* 2013)
- DESeq 1.8.1 (Anders and Huber 2010)
- baySeq 1.10.0 (Hardcastle and Kelly 2010)
- edgeR 2.6.0 (Robinson *et al.* 2010)
- DEGseq 1.10.0 (Wang *et al.* 2010)
- NOISeq 29-IV-2011 (Tarazona *et al.* 2011)
- PoissonSeq (Li *et al.* 2012)
- SAMseq samr 2.0 (Li and Tibshirani 2011)
- QuasiSeq 1.0-2 (Lund *et al.* 2012)
- NBPSeq 0.1.8 (Cumbie *et al.* 2011)

- TSPM version of 13th May 2011 (Auer and Doerge 2011)

- tweeDEseq 1.4.1 Unpublished method from Bioconductor (Gentleman *et al.* 2004)

- DESeq2 1.0.17 (Anders and Huber 2010)

We used the default settings of all methods. All methods supply a ranking criterion like a p-value.

S4.2.2 Simulated Data With Two Known Conditions

Data Simulation. We simulated datasets with 10^6, 10^7, or 10^8 reads per sample (the library size). For each library size 2, 6 or 15 replicates per condition were simulated. For each of these nine combinations we generated 100 datasets with 10,000 transcripts each. Under condition i the reads for a transcript distributed according to $\mathrm{NB}(x;\ \mu_i, r_i)$. For the selection of the mean μ_i and the size r_i ($r_i = \phi_i^{-1}$ with overdispersion ϕ_i) we sampled values from the from the "Mice Strains" RNA-Seq dataset (Bottomly *et al.* 2011), where we used only one biological condition. Following Wu *et al.* (2013), we sampled μ_i values from the median read counts of the transcripts. The overdispersion ϕ tends to decrease with increasing mean read counts as shown in Fig. S15. Therefore we fitted a regression line to overdispersions by least squares. After sampling the log μ_i values, we calculated the corresponding log ϕ_i values according to the regression line, added Gaussian noise ($\sigma^2 = 1$) to the log ϕ_i values and transformed the overdispersion into the size parameter $r_i = 1/\phi_i$. 30% of the genes were chosen to be differentially expressed. Differential expression was expressed by adjusting the means of the negative binomials to obtain fold changes of 0.5, 1 and 1.5 (randomly chosen) between these means.

Results. We first compared the methods on simulated data for two condition. Tabs. S3, S4, and S5 report the results for a library size of 10^6, 10^7, and 10^8. DEXUS estimates the dispersion parameter with comparable performance to other methods when the sample size is low. DEXUS outperforms the other methods when the sample size is medium, i.e. six replicates, or large, i.e. fifteen replicates.

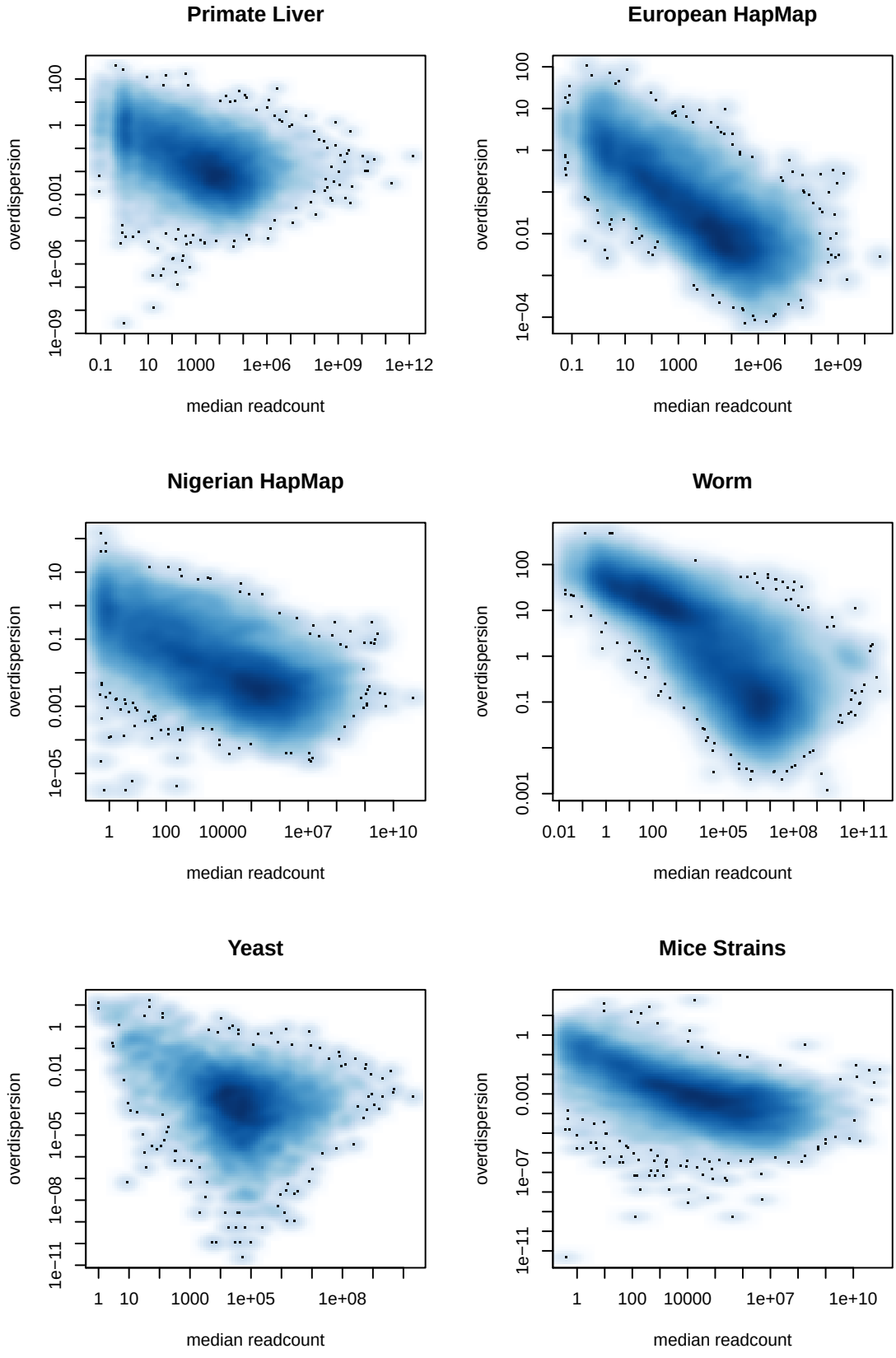

Figure S15: Scatterplots of overdispersion and median read counts of various datasets (see Tab. S20 in Subsection S5.1). For each transcript the median count was computed and the overdispersion estimated by maximum likelihood.

Libsize	Method	$\mathrm{AUC}_{\mathrm{ROC}}$	$\mathrm{AUC}_{\mathrm{PR}}$
10^6	DEXUS	0.765±0.0019	0.669±0.0025
10^6	DESeq	0.752±0.0022	0.687±0.0025
10^6	DESeq2	0.774±0.0056	0.693±0.0069
10^6	edgeR	0.753±0.0023	0.687±0.0026
10^6	baySeq	**0.800**±0.0017	**0.700**±0.0023
10^6	NOISeq	0.750±0.0021	0.677±0.0026
10^6	SAMseq	0.735±0.0024	0.572±0.0053
10^6	DSS	0.773±0.0021	0.695±0.0025
10^6	PoissonSeq	0.700±0.0024	0.597±0.0042
10^6	NBPSeq	*0.777*±0.0056	*0.698*±0.0074
10^6	QuasiSeq	0.713±0.0069	0.670±0.0075
10^6	TSPM	0.628±0.0096	0.515±0.0074
10^6	TweeDEseq	0.733±0.0062	0.649±0.0071
10^7	DEXUS	*0.896*±0.0014	0.839±0.0021
10^7	DESeq	0.882±0.0016	0.841±0.0020
10^7	DESeq2	0.887±0.0042	0.843±0.0055
10^7	edgeR	0.894±0.0015	*0.853*±0.0020
10^7	baySeq	*0.896*±0.0014	0.849±0.0020
10^7	NOISeq	0.851±0.0017	0.794±0.0021
10^7	SAMseq	0.836±0.0018	0.663±0.0043
10^7	DSS	**0.897**±0.0015	**0.856**±0.0019
10^7	PoissonSeq	0.821±0.0025	0.711±0.0056
10^7	NBPSeq	0.893±0.0043	0.845±0.0061
10^7	QuasiSeq	0.889±0.0045	0.852±0.0054
10^7	TSPM	0.783±0.0051	0.619±0.0087
10^7	TweeDEseq	0.839±0.0052	0.778±0.0060
10^8	DEXUS	0.961±0.0008	0.930±0.0014
10^8	DESeq	0.955±0.0009	0.934±0.0011
10^8	DESeq2	0.956±0.0025	0.935±0.0033
10^8	edgeR	*0.962*±0.0009	*0.942*±0.0011
10^8	baySeq	0.954±0.0010	0.933±0.0011
10^8	NOISeq	0.908±0.0011	0.860±0.0016
10^8	SAMseq	0.888±0.0012	0.705±0.0037
10^8	DSS	**0.965**±0.0008	**0.945**±0.0010
10^8	PoissonSeq	0.884±0.0018	0.787±0.0059
10^8	NBPSeq	0.960±0.0026	0.936±0.0037
10^8	QuasiSeq	0.959±0.0024	0.941±0.0030
10^8	TSPM	0.871±0.0041	0.686±0.0142
10^8	TweeDEseq	0.897±0.0045	0.847±0.0057

Table S3: Performance of methods for two known conditions with 2 replicates and a library size of 10^6, 10^7, and 10^8. The best methods with respect to $\mathrm{AUC}_{\mathrm{ROC}}$ are DSS, baySeq, and DEXUS. With respect to $\mathrm{AUC}_{\mathrm{PR}}$ DSS, baySeq and edgeR perform best.

Libsize	Method	$\mathrm{AUC}_{\mathrm{ROC}}$	$\mathrm{AUC}_{\mathrm{PR}}$
10^6	DEXUS	0.865±0.0017	0.825±0.0019
10^6	DESeq	0.855±0.0018	0.828±0.0020
10^6	DESeq2	0.864±0.0045	0.831±0.0052
10^6	edgeR	0.856±0.0019	0.830±0.0020
10^6	baySeq	**0.892**±0.0013	**0.845**±0.0018
10^6	NOISeq	0.812±0.0020	0.783±0.0023
10^6	SAMseq	0.847±0.0019	0.812±0.0020
10^6	DSS	*0.868*±0.0016	*0.835*±0.0019
10^6	PoissonSeq	0.803±0.0023	0.723±0.0040
10^6	NBPSeq	0.864±0.0050	0.826±0.0059
10^6	QuasiSeq	0.834±0.0063	0.820±0.0060
10^6	TSPM	0.828±0.0061	0.782±0.0077
10^6	TweeDEseq	0.852±0.0048	0.810±0.0056
10^7	DEXUS	**0.964**±0.0008	**0.949**±0.0010
10^7	DESeq	0.958±0.0009	*0.946*±0.0010
10^7	DESeq2	0.956±0.0024	0.943±0.0028
10^7	edgeR	*0.961*±0.0010	**0.949**±0.0011
10^7	baySeq	0.957±0.0009	0.942±0.0011
10^7	NOISeq	0.918±0.0013	0.889±0.0015
10^7	SAMseq	0.948±0.0010	0.934±0.0011
10^7	DSS	*0.961*±0.0008	**0.949**±0.0009
10^7	PoissonSeq	0.900±0.0018	0.828±0.0049
10^7	NBPSeq	0.959±0.0028	0.941±0.0036
10^7	QuasiSeq	0.959±0.0024	*0.946*±0.0028
10^7	TSPM	0.953±0.0024	0.935±0.0030
10^7	TweeDEseq	0.938±0.0037	0.926±0.0035
10^8	DEXUS	**0.993**±0.0003	**0.988**±0.0004
10^8	DESeq	0.990±0.0004	*0.986*±0.0005
10^8	DESeq2	0.989±0.0013	0.985±0.0015
10^8	edgeR	*0.992*±0.0004	**0.988**±0.0005
10^8	baySeq	0.985±0.0007	0.981±0.0008
10^8	NOISeq	0.958±0.0007	0.934±0.0010
10^8	SAMseq	0.986±0.0005	0.980±0.0007
10^8	DSS	*0.992*±0.0004	**0.988**±0.0004
10^8	PoissonSeq	0.941±0.0013	0.881±0.0041
10^8	NBPSeq	0.991±0.0012	0.984±0.0016
10^8	QuasiSeq	0.989±0.0014	0.985±0.0017
10^8	TSPM	0.987±0.0014	0.981±0.0018
10^8	TweeDEseq	0.965±0.0032	0.965±0.0027

Table S4: Performance of methods for two known conditions with 6 replicates and a library size of 10^6, 10^7, and 10^8. The best methods with respect to $\mathrm{AUC}_{\mathrm{ROC}}$ are DEXUS and DSS. With respect to $\mathrm{AUC}_{\mathrm{PR}}$ DEXUS, baySeq and DSS perform best.

Libsize	Method	$\mathrm{AUC}_{\mathrm{ROC}}$	$\mathrm{AUC}_{\mathrm{PR}}$
10^6	DEXUS	*0.928*±0.0014	0.910±0.0014
10^6	DESeq	0.922±0.0015	0.910±0.0014
10^6	DESeq2	0.925±0.0038	0.909±0.0040
10^6	edgeR	0.921±0.0017	0.908±0.0018
10^6	baySeq	**0.943**±0.0011	**0.919**±0.0014
10^6	NOISeq	0.851±0.0020	0.840±0.0026
10^6	SAMseq	0.910±0.0015	0.902±0.0062
10^6	DSS	0.927±0.0014	*0.912*±0.0014
10^6	PoissonSeq	0.861±0.0027	0.800±0.0047
10^6	NBPSeq	0.922±0.0049	0.903±0.0056
10^6	QuasiSeq	0.910±0.0048	0.894±0.0045
10^6	TSPM	0.916±0.0046	0.900±0.0047
10^6	TweeDEseq	0.921±0.0038	0.902±0.0041
10^7	DEXUS	**0.989**±0.0004	**0.985**±0.0005
10^7	DESeq	0.986±0.0005	*0.982*±0.0006
10^7	DESeq2	0.984±0.0016	0.979±0.0018
10^7	edgeR	0.986±0.0008	*0.982*±0.0009
10^7	baySeq	0.982±0.0008	0.977±0.0010
10^7	NOISeq	0.954±0.0008	0.938±0.0009
10^7	SAMseq	0.981±0.0006	0.978±0.0048
10^7	DSS	*0.987*±0.0005	*0.982*±0.0006
10^7	PoissonSeq	0.942±0.0013	0.886±0.0041
10^7	NBPSeq	0.985±0.0019	0.979±0.0022
10^7	QuasiSeq	0.981±0.0023	0.970±0.0032
10^7	TSPM	0.985±0.0015	0.980±0.0018
10^7	TweeDEseq	0.976±0.0025	0.973±0.0024
10^8	DEXUS	**0.999**±0.0001	**0.999**±0.0001
10^8	DESeq	*0.998*±0.0002	0.997±0.0002
10^8	DESeq2	0.997±0.0006	0.996±0.0007
10^8	edgeR	*0.998*±0.0003	0.997±0.0004
10^8	baySeq	0.995±0.0006	0.994±0.0007
10^8	NOISeq	0.981±0.0004	0.967±0.0006
10^8	SAMseq	0.997±0.0002	0.996±0.0003
10^8	DSS	*0.998*±0.0002	*0.998*±0.0002
10^8	PoissonSeq	0.969±0.0008	0.927±0.0029
10^8	NBPSeq	*0.998*±0.0007	0.997±0.0009
10^8	QuasiSeq	0.996±0.0011	0.991±0.0020
10^8	TSPM	*0.998*±0.0005	0.996±0.0006
10^8	TweeDEseq	0.979±0.0025	0.983±0.0019

Table S5: Performance of methods for two known conditions with 15 replicates and a library size of 10^6, 10^7, and 10^8. The best methods with respect to $\mathrm{AUC}_{\mathrm{ROC}}$ are DEXUS, baySeq, and DSS. With respect to $\mathrm{AUC}_{\mathrm{PR}}$ DEXUS and DSS perform best.

S4.2.3 Simulated Data With Multiple Known Conditions

Data Simulation. We simulated data for multi-class problems for three conditions with 2, 6 or 15 replicates each. The data was generated like for two known conditions as described in Subsection S4.2.2. If a transcript was selected to be differentially expressed, one group or two groups were given a log fold change of either 0.5, 1.0 or 1.5 (randomly chosen). We compared DEXUS to the multi-class versions of edgeR, baySeq, and SAMSeq.

Results. We first compared the methods on simulated data for two condition. Tabs. S6, S7, and S8 report the results for a library size of 10^6, 10^7, and 10^8. DEXUS outperforms the other methods when the sample size is medium, i.e. six replicates, or large, i.e. fifteen replicates.

Libsize	Method	$\mathrm{AUC}_{\mathrm{ROC}}$	$\mathrm{AUC}_{\mathrm{PR}}$
10^6	DEXUS	*0.830*±0.0023	0.745±0.0034
10^6	edgeR	0.827±0.0025	**0.755**±0.0029
10^6	baySeq	**0.833**±0.0023	0.745±0.0042
10^6	DESeq	0.820±0.0024	*0.753*±0.0029
10^6	SAMseq	0.780±0.0026	0.678±0.0038
10^7	DEXUS	**0.936**±0.0013	**0.896**±0.0020
10^7	edgeR	*0.931*±0.0014	**0.896**±0.0020
10^7	baySeq	0.920±0.0016	0.874±0.0047
10^7	DESeq	0.921±0.0016	*0.888*±0.0021
10^7	SAMseq	0.862±0.0044	0.713±0.0249
10^8	DEXUS	**0.979**±0.0006	*0.961*±0.0009
10^8	edgeR	*0.977*±0.0007	**0.962**±0.0010
10^8	baySeq	0.965±0.0010	0.939±0.0040
10^8	DESeq	0.972±0.0008	0.957±0.0010
10^8	SAMseq	0.877±0.0012	0.668±0.0114

Table S6: Performance of methods for three known conditions with 2 replicates and a library size of 10^6, 10^7, and 10^8. The best methods with respect to $\mathrm{AUC}_{\mathrm{ROC}}$ are baySeq and DEXUS. With respect to $\mathrm{AUC}_{\mathrm{PR}}$ edgeR and DEXUS perform best.

Libsize	Method	$\mathrm{AUC}_{\mathrm{ROC}}$	$\mathrm{AUC}_{\mathrm{PR}}$
10^6	DEXUS	*0.913*±0.0015	**0.882**±0.0018
10^6	edgeR	0.907±0.0017	*0.880*±0.0019
10^6	baySeq	**0.915**±0.0015	0.874±0.0035
10^6	DESeq	0.905±0.0016	0.877±0.0019
10^6	SAMseq	0.890±0.0017	0.877±0.0170
10^7	DEXUS	**0.982**±0.0006	**0.973**±0.0008
10^7	edgeR	*0.977*±0.0009	*0.969*±0.0011
10^7	baySeq	0.969±0.0011	0.951±0.0039
10^7	DESeq	0.975±0.0008	0.966±0.0010
10^7	SAMseq	0.967±0.0010	0.957±0.0034
10^8	DEXUS	**0.997**±0.0002	**0.995**±0.0002
10^8	edgeR	*0.996*±0.0003	*0.994*±0.0004
10^8	baySeq	0.988±0.0007	0.976±0.0040
10^8	DESeq	0.995±0.0003	0.992±0.0004
10^8	SAMseq	0.992±0.0005	0.989±0.0006

Table S7: Performance of methods for there known conditions with 6 replicates and a library size of 10^6, 10^7, and 10^8. The best methods with respect to $\mathrm{AUC}_{\mathrm{ROC}}$ are baySeq, edgeR, and DEXUS. With respect to $\mathrm{AUC}_{\mathrm{PR}}$ DEXUS performs best.

Libsize	Method	$\mathrm{AUC}_{\mathrm{ROC}}$	$\mathrm{AUC}_{\mathrm{PR}}$
10^6	DEXUS	**0.958**±0.0012	**0.945**±0.0013
10^6	edgeR	0.952±0.0015	0.939±0.0016
10^6	baySeq	*0.956*±0.0011	0.931±0.0041
10^6	DESeq	0.953±0.0013	*0.941*±0.0013
10^6	SAMseq	0.942±0.0013	0.931±0.0078
10^7	DEXUS	**0.996**±0.0003	**0.993**±0.0004
10^7	edgeR	0.992±0.0006	*0.990*±0.0006
10^7	baySeq	0.986±0.0008	0.973±0.0035
10^7	DESeq	*0.993*±0.0005	*0.990*±0.0005
10^7	SAMseq	0.990±0.0006	0.986±0.0007
10^8	DEXUS	**1.000**±0.0001	**1.000**±0.0001
10^8	edgeR	*0.999*±0.0002	*0.999*±0.0003
10^8	baySeq	0.994±0.0006	0.983±0.0043
10^8	DESeq	*0.999*±0.0001	*0.999*±0.0002
10^8	SAMseq	0.998±0.0002	0.998±0.0009

Table S8: Performance of methods for three known conditions with 15 replicates and a library size of 10^6, 10^7, and 10^8. The best method with respect to both $\mathrm{AUC}_{\mathrm{ROC}}$ and $\mathrm{AUC}_{\mathrm{PR}}$ is DEXUS.

S4.2.4 Real World Data with Two Known Conditions

We compare the methods on real-world data, the "Mice Strains" data set, which has already been used for benchmarking RNA-Seq methods. In Bottomly *et al.* (2011), two strains of mice, C57BL/6J (B6) and DBA/2J (D2), were compared using both RNA-Seq and microarrays. The dataset consists of 21 lanes from male mice (10 of the B6 strain and 11 of D2 strain), produced using an Illumina GAIIx sequencing machine. The dataset was provided by the ReCount repository (Frazee *et al.* 2011) that is based on Ensembl 61 gene definitions. DEXUS found 157 genes that were significant using the mentioned test for differentially expression after Bonferroni correction at a significance level of 0.01. Of these 157 genes 97.5% were confirmed by at least one of the eight other methods, 91% by at least two other methods, and 85% by at least three other methods. 8% were confirmed by all eight methods. To compare the result of DEXUS to the results of the original publication (Bottomly *et al.* 2011), we used the authors' read count data that is based on an older version of the Ensembl gene definitions (Ensembl 59). DEXUS identified 258 genes as differentially expressed. Of these 258 genes 245 were also identified in the original publication and confirmed by both Affymetrix and Illumina microarrays. The gene sets extracted by DEXUS are analyzed by the DAVID annotation tool (Huang *et al.* 2009b, a) for gene enrichment using gene ontology (Ashburner *et al.* 2000) and the INTERPRO data base (Hunter *et al.* 2012). Significant terms were "antigen processing and presentation" ($p = 9.7\text{e-}6$), "antigen processing an presentation of peptide antigen" ($p = 1.1\text{e-}5$)," Immunoglobulin/major histocompatibility complex, conserved site" ($p = 4.2\text{e-}4$), and "Immunoglobulin-like" ($p = 3.2\text{e-}4$). This shows that many transcripts that are differentially expressed between the two mice strains are related to the immune system.

S4.3 RNA-Seq Data with Unknown Conditions

The idea of DEXUS is to estimate the conditions and read counts belonging to them by a mixture model. This can also be done by a mixture of Gaussians. We compare DEXUS which is a mixture of negative binomials to a mixture of Gaussians to assess whether negative binomials are indeed the appropriate mixture components to model RNA-Seq data. We select mclust 4.0.0 (Fraley *et al.* 2012) as baseline method. It is used to model RNA-Seq data by a mixture of Gaussians. For the baseline method gene ranking was performed according to DEXUS' I/NI value.

S4.3.1 Methods compared

We compare the following methods for differential gene expression in RNA-Seq data with unknown conditions:

- DEXUS,
- baseline method: `mclust` (Fraley *et al.* 2012; Fraley and Raftery 2002).

The baseline method is mixture of Gaussians clustering algorithm. We model the data with one, two and three Gaussians. We use the DEXUS I/NI value (see Subsection S3.3) to rank the transcripts according to the evidence of being differentially expressed. The values $\boldsymbol{\alpha}$, $\boldsymbol{\mu}$,and α_{ik} are provided by the Gaussian mixture EM algorithm.

In principle it is possible to test all possible partitions of the samples into two or more conditions and then apply standard RNA-Seq methods. However this approach is not feasible because the number of partitions increases combinatorial. The number of partitions of a set with N elements is given by the Bell number B_N.

$$B_N = \frac{1}{e} \sum_{k=0}^{\infty} \frac{k^N}{k!} \tag{S74}$$

For multiple conditions, the number of data sets is the number of partions minus one $B_N - 1$ (the set of all data is not considered). For $N = 10$ samples, the Bell number is $B_{10} = 115,975$, therefore a method has to run 115,974 data sets. For $N = 20$ samples, a method has to run 51,724,158,235,371 data sets.

S4.3.2 Simulated Data Sets with Unknown Conditions

Data Simulation. We simulated datasets analogously to data with known conditions (see Subsection S3.3.1), except that the conditions are withhold from the methods. Furthermore, the conditions can have different number of replicates as expected for general study designs.

Results. Tab. S9, Tab. S10, and Tab. S11 show the results in terms of $\mathrm{AUC}_{\mathrm{ROC}}$ and $\mathrm{AUC}_{\mathrm{PR}}$ for DEXUS and mclust for library sizes 10^6, 10^7, and 10^8, respectively. In all experiments DEXUS outperforms the baseline method. This is not surprising as modeling with a negative binomial is supposed to perform better than modeling with a Gaussian distribution.

Performance at different I/NI thresholds Tab. S12, Tab. S13, and Tab. S14 show the results of DEXUS in terms of different performance measures like sensitivity and specificity at different I/NI thresholds and for library sizes 10^6, 10^7, and 10^8, respectively. Additional performance measures are given in Section S6.

Performance for different fold change categories We investigated whether DEXUS has a different performance on differentially expressed genes belonging to different fold change categories. Each data set in the simulated data consists of around 7,000 non differentially expressed genes, around 1,000 genes with a log fold change of 0.5, around 1,000 genes with a log fold change of 1, and around 1,000 genes with a log fold change of 1.5. We assess the performance of DEXUS in terms of specificity and sensitivity on a data sets of 8,000 genes (7,000 negatives and 1,000 positives), one data set for each fold change. The results for these different fold change categories are displayed in Tables S15, S16, and S17 for library sizes of 10^6, 10^7, and 10^8, respectively.

For the different fold change categories, the set of negatives (the 7,000 non differentially expressed genes) is the same and the number of the false positives is the same (as the I/NI threshold is the same), therefore also the specificity is the same. The sensitivity values increase with the log fold change. The reason is that genes with larger log fold changes lead to higher I/NI values (larger distances between the read count means), thus are easier to be detected. The lower the number of samples of the smaller condition, the lower the sensitivity. The signal of the smaller condition is more likely to be confounded with outliers. Table S shows that at a threshold of 0.05

C1/C2	Method	$\mathrm{AUC}_{\mathrm{ROC}}$	$\mathrm{AUC}_{\mathrm{PR}}$
6/6	DEXUS	0.772 ± 0.0036	0.580 ± 0.0073
6/6	baseline method	0.648 ± 0.0042	0.478 ± 0.0077
9/3	DEXUS	0.773 ± 0.0034	0.553 ± 0.0065
9/3	baseline method	0.768 ± 0.0037	0.401 ± 0.0058
10/2	DEXUS	0.764 ± 0.0038	0.517 ± 0.0068
10/2	baseline method	0.603 ± 0.0036	0.357 ± 0.0055
11/1	DEXUS	0.733 ± 0.0038	0.451 ± 0.0064
11/1	baseline method	0.522 ± 0.0031	0.274 ± 0.0039
12/12	DEXUS	0.758 ± 0.0034	0.598 ± 0.0058
12/12	baseline method	0.669 ± 0.0035	0.515 ± 0.0063
18/6	DEXUS	0.782 ± 0.0032	0.603 ± 0.0054
18/6	baseline method	0.645 ± 0.0038	0.447 ± 0.0061
20/4	DEXUS	0.764 ± 0.0034	0.587 ± 0.0059
20/4	baseline method	0.627 ± 0.0036	0.409 ± 0.0057
22/2	DEXUS	0.741 ± 0.0034	0.519 ± 0.0060
22/2	baseline method	0.591 ± 0.0032	0.356 ± 0.0042

Table S9: Results of DEXUS and the baseline method (`mclust`) for unknown conditions (two conditions). "C1/C2" reports the number of samples for each condition. "Method" gives the name of the method and $\mathrm{AUC}_{\mathrm{ROC}}$ and $\mathrm{AUC}_{\mathrm{PR}}$ the according performances. The library size was 10^6 for all experiments. DEXUS consistently outperforms the baseline method.

C1/C2	Method	$\mathrm{AUC}_{\mathrm{ROC}}$	$\mathrm{AUC}_{\mathrm{PR}}$
6/6	DEXUS	0.838 ± 0.0035	0.745 ± 0.0056
6/6	baseline method	0.728 ± 0.0041	0.662 ± 0.0070
9/3	DEXUS	0.843 ± 0.0027	0.714 ± 0.0060
9/3	baseline method	0.702 ± 0.0039	0.568 ± 0.0069
10/2	DEXUS	0.832 ± 0.0028	0.663 ± 0.0060
10/2	baseline method	0.673 ± 0.0036	0.495 ± 0.0056
11/1	DEXUS	0.792 ± 0.0041	0.559 ± 0.0070
11/1	baseline method	0.512 ± 0.0029	0.314 ± 0.0038
12/12	DEXUS	0.833 ± 0.0026	0.755 ± 0.0042
12/12	baseline method	0.764 ± 0.0035	0.706 ± 0.0068
18/6	DEXUS	0.851 ± 0.0032	0.771 ± 0.0054
18/6	baseline method	0.743 ± 0.0036	0.632 ± 0.0067
20/4	DEXUS	0.847 ± 0.0034	0.745 ± 0.0059
20/4	baseline method	0.719 ± 0.0032	0.573 ± 0.0062
22/2	DEXUS	0.817 ± 0.0035	0.648 ± 0.0060
22/2	baseline method	0.674 ± 0.0035	0.484 ± 0.0051

Table S10: Results of DEXUS and Gaussian mixtures (mclust) for unknown conditions (two conditions). "C1/C2" reports the number of samples for each condition. "Method" gives the name of the method and $\mathrm{AUC}_{\mathrm{ROC}}$ and $\mathrm{AUC}_{\mathrm{PR}}$ the according performances. The library size was 10^7 for all experiments. DEXUS consistently outperforms the baseline method.

C1/C2	Method	$\mathrm{AUC}_{\mathrm{ROC}}$	$\mathrm{AUC}_{\mathrm{PR}}$
6/6	DEXUS	0.914 ± 0.0025	0.874 ± 0.0038
6/6	baseline method	0.834 ± 0.0035	0.824 ± 0.0045
9/3	DEXUS	0.921 ± 0.0020	0.852 ± 0.0054
9/3	baseline method	0.813 ± 0.0035	0.762 ± 0.0065
10/2	DEXUS	0.908 ± 0.0025	0.791 ± 0.0053
10/2	baseline method	0.785 ± 0.0033	0.681 ± 0.0062
11/1	DEXUS	0.862 ± 0.0027	0.652 ± 0.0060
11/1	baseline method	0.513 ± 0.0026	0.328 ± 0.0038
12/12	DEXUS	0.912 ± 0.0023	0.880 ± 0.0031
12/12	baseline method	0.863 ± 0.0030	0.856 ± 0.0037
18/6	DEXUS	0.931 ± 0.0023	0.899 ± 0.0036
18/6	baseline method	0.849 ± 0.0031	0.816 ± 0.0050
20/4	DEXUS	0.926 ± 0.0024	0.872 ± 0.0047
20/4	baseline method	0.828 ± 0.0031	0.762 ± 0.0051
22/2	DEXUS	0.897 ± 0.0028	0.770 ± 0.0055
22/2	baseline method	0.786 ± 0.0032	0.654 ± 0.0070

Table S11: Results of DEXUS and Gaussian mixtures (mclust) for unknown conditions (two conditions). "C1/C2" reports the number of samples for each condition. "Method" gives the name of the method and $\mathrm{AUC}_{\mathrm{ROC}}$ and $\mathrm{AUC}_{\mathrm{PR}}$ the according performances. The library size was 10^8 for all experiments. DEXUS consistently outperforms the baseline method.

strong signals (log fold changes of 1.5) can still be reliably detected even if they appear in only a few samples ("11/1" or "22/2").

I/NI threshold	0.025		0.05		0.1	
C1/C2	specificity	sensitivity	specificity	sensitivity	specificity	sensitivity
6/6	0.938 ± 0.003	0.327 ± 0.009	0.955 ± 0.003	0.310 ± 0.009	0.976 ± 0.002	0.278 ± 0.010
9/3	0.938 ± 0.003	0.362 ± 0.008	0.955 ± 0.002	0.340 ± 0.007	0.976 ± 0.002	0.276 ± 0.007
10/2	0.938 ± 0.003	0.362 ± 0.010	0.955 ± 0.002	0.327 ± 0.009	0.976 ± 0.002	0.193 ± 0.007
11/1	0.938 ± 0.003	0.319 ± 0.009	0.955 ± 0.002	0.219 ± 0.009	0.976 ± 0.002	0.045 ± 0.004
12/12	0.959 ± 0.002	0.281 ± 0.009	0.978 ± 0.002	0.255 ± 0.008	0.993 ± 0.001	0.222 ± 0.008
18/6	0.959 ± 0.002	0.332 ± 0.010	0.978 ± 0.002	0.298 ± 0.009	0.993 ± 0.001	0.233 ± 0.009
20/4	0.959 ± 0.002	0.337 ± 0.009	0.979 ± 0.002	0.285 ± 0.009	0.993 ± 0.001	0.152 ± 0.008
22/2	0.959 ± 0.002	0.295 ± 0.008	0.978 ± 0.002	0.170 ± 0.007	0.993 ± 0.001	0.015 ± 0.003
Mean	$\mathbf{0.949 \pm 0.011}$	$\mathbf{0.327 \pm 0.029}$	$\mathbf{0.967 \pm 0.012}$	$\mathbf{0.275 \pm 0.058}$	$\mathbf{0.984 \pm 0.009}$	$\mathbf{0.177 \pm 0.100}$

Table S12: The performance of DEXUS in terms of sensitivity and specificity at the detection of differential expression with unknown conditions. The first row reports the different thresholds that were used for the I/NI value. The first column "C1/C2" reports the number of replicates for the first and second condition. The other columns report sensitivity and specificity of DEXUS at different I/NI thresholds. The library size was 10^6 for all experiments.

I/NI threshold	0.025		0.05		0.1	
C1/C2	specificity	sensitivity	specificity	sensitivity	specificity	sensitivity
6/6	0.896 ± 0.004	0.550 ± 0.011	0.940 ± 0.003	0.517 ± 0.011	0.976 ± 0.002	0.465 ± 0.010
9/3	0.897 ± 0.004	0.600 ± 0.009	0.941 ± 0.003	0.557 ± 0.009	0.976 ± 0.002	0.438 ± 0.009
10/2	0.898 ± 0.004	0.597 ± 0.009	0.941 ± 0.003	0.509 ± 0.009	0.976 ± 0.002	0.269 ± 0.009
11/1	0.898 ± 0.004	0.516 ± 0.010	0.941 ± 0.003	0.303 ± 0.009	0.976 ± 0.002	0.036 ± 0.003
12/12	0.940 ± 0.003	0.516 ± 0.009	0.975 ± 0.002	0.470 ± 0.009	0.993 ± 0.001	0.424 ± 0.008
18/6	0.941 ± 0.003	0.590 ± 0.010	0.976 ± 0.002	0.532 ± 0.010	0.993 ± 0.001	0.417 ± 0.010
20/4	0.940 ± 0.003	0.590 ± 0.010	0.975 ± 0.002	0.479 ± 0.010	0.993 ± 0.001	0.243 ± 0.008
22/2	0.940 ± 0.003	0.497 ± 0.010	0.975 ± 0.002	0.262 ± 0.009	0.993 ± 0.001	0.011 ± 0.002
Mean	$\mathbf{0.919 \pm 0.023}$	$\mathbf{0.557 \pm 0.042}$	$\mathbf{0.958 \pm 0.018}$	$\mathbf{0.454 \pm 0.110}$	$\mathbf{0.993 \pm 0.005}$	$\mathbf{0.171 \pm 0.177}$

Table S13: The performance of DEXUS in terms of sensitivity and specificity at the detection of differential expression with unknown conditions. The first row reports the different thresholds that were used for the I/NI value. The first column "C1/C2" reports the number of replicates for the first and second condition. The other columns report sensitivity and specificity of DEXUS at different I/NI thresholds. The library size was 10^7 for all experiments.

I/NI threshold	0.025		0.05		0.1	
C1/C2	specificity	sensitivity	specificity	sensitivity	specificity	sensitivity
6/6	0.893 ± 0.003	0.775 ± 0.009	0.951 ± 0.002	0.720 ± 0.009	0.985 ± 0.002	0.646 ± 0.009
9/3	0.893 ± 0.004	0.827 ± 0.006	0.951 ± 0.002	0.766 ± 0.007	0.985 ± 0.001	0.580 ± 0.008
10/2	0.893 ± 0.003	0.819 ± 0.008	0.950 ± 0.002	0.656 ± 0.009	0.985 ± 0.001	0.325 ± 0.009
11/1	0.893 ± 0.003	0.677 ± 0.009	0.951 ± 0.002	0.351 ± 0.008	0.985 ± 0.001	0.020 ± 0.003
12/12	0.945 ± 0.002	0.735 ± 0.008	0.982 ± 0.001	0.665 ± 0.008	0.996 ± 0.001	0.610 ± 0.009
18/6	0.945 ± 0.003	0.816 ± 0.008	0.982 ± 0.002	0.743 ± 0.009	0.996 ± 0.001	0.570 ± 0.011
20/4	0.945 ± 0.003	0.810 ± 0.008	0.982 ± 0.002	0.625 ± 0.009	0.996 ± 0.001	0.308 ± 0.009
22/2	0.946 ± 0.002	0.650 ± 0.009	0.982 ± 0.001	0.325 ± 0.008	0.996 ± 0.001	0.006 ± 0.002
Mean	$\mathbf{0.919 \pm 0.028}$	$\mathbf{0.764 \pm 0.069}$	$\mathbf{0.966 \pm 0.017}$	$\mathbf{0.606 \pm 0.172}$	$\mathbf{0.991 \pm 0.006}$	$\mathbf{0.383 \pm 0.261}$

Table S14: The performance of DEXUS in terms of sensitivity and specificity at the detection of differential expression with unknown conditions. The first row reports the different thresholds that were used for the I/NI value. The first column "C1/C2" reports the number of replicates for the first and second condition. The other columns report sensitivity and specificity of DEXUS at different I/NI thresholds. The library size was 10^8 for all experiments.

I/NI threshold	0.025		0.05		0.1	
C1/C2	specificity	sensitivity	specificity	sensitivity	specificity	sensitivity
6/6	0.94|0.94|0.94	0.10|0.28|0.45	0.95|0.95|0.95	0.08|0.27|0.45	0.98|0.98|0.98	0.04|0.24|0.43
9/3	0.94|0.94|0.94	0.13|0.32|0.48	0.95|0.95|0.95	0.09|0.30|0.47	0.98|0.98|0.98	0.03|0.22|0.45
10/2	0.94|0.94|0.94	0.13|0.32|0.47	0.95|0.95|0.95	0.08|0.29|0.47	0.98|0.98|0.98	0.03|0.08|0.38
11/1	0.94|0.94|0.94	0.10|0.28|0.43	0.95|0.95|0.95	0.06|0.12|0.38	0.98|0.98|0.98	0.03|0.04|0.06
12/12	0.96|0.96|0.96	0.07|0.27|0.44	0.98|0.98|0.98	0.04|0.24|0.42	0.99|0.99|0.99	0.01|0.20|0.40
18/6	0.96|0.96|0.96	0.10|0.32|0.49	0.98|0.98|0.98	0.06|0.29|0.47	0.99|0.99|0.99	0.01|0.20|0.43
20/4	0.96|0.96|0.96	0.10|0.33|0.49	0.98|0.98|0.98	0.03|0.27|0.47	0.99|0.99|0.99	0.01|0.03|0.37
22/2	0.96|0.96|0.96	0.06|0.29|0.45	0.98|0.98|0.98	0.03|0.07|0.37	0.99|0.99|0.99	0.01|0.01|0.02

Table S15: The performance of DEXUS in terms of sensitivity and specificity in detecting differential expression with unknown conditions. The results are separately reported for the three different fold change categories. The first column "C1/C2" contains the numbers of replicates for the first and second condition. The other columns list sensitivity and specificity of DEXUS at different I/NI thresholds as the average across 100 data sets. The first, second, and third value in each cell corresponds to a log fold change of 0.5, 1, and 1.5, respectively. The library size was 10^6 for all experiments.

I/NI threshold	0.025		0.05		0.1	
C1/C2	specificity	sensitivity	specificity	sensitivity	specificity	sensitivity
6/6	0.90|0.90|0.90	0.25|0.62|0.78	0.94|0.94|0.94	0.18|0.60|0.77	0.98|0.98|0.98	0.09|0.55|0.75
9/3	0.90|0.90|0.90	0.33|0.65|0.80	0.94|0.94|0.94	0.24|0.63|0.79	0.98|0.98|0.98	0.04|0.50|0.76
10/2	0.90|0.90|0.90	0.33|0.65|0.79	0.94|0.94|0.94	0.12|0.61|0.78	0.98|0.98|0.98	0.03|0.09|0.67
11/1	0.90|0.90|0.90	0.19|0.59|0.75	0.94|0.94|0.94	0.07|0.16|0.66	0.98|0.98|0.98	0.03|0.03|0.05
12/12	0.94|0.94|0.94	0.18|0.60|0.77	0.98|0.98|0.98	0.09|0.56|0.75	0.99|0.99|0.99	0.03|0.51|0.73
18/6	0.94|0.94|0.94	0.30|0.66|0.81	0.98|0.98|0.98	0.18|0.62|0.79	0.99|0.99|0.99	0.01|0.48|0.76
20/4	0.94|0.94|0.94	0.30|0.66|0.81	0.98|0.98|0.98	0.06|0.59|0.79	0.99|0.99|0.99	0.01|0.04|0.68
22/2	0.94|0.94|0.94	0.12|0.60|0.77	0.98|0.98|0.98	0.03|0.09|0.67	0.99|0.99|0.99	0.01|0.01|0.02

Table S16: The performance of DEXUS in terms of sensitivity and specificity in detecting differential expression with unknown conditions. The results are separately reported for the three different fold change categories. The first column "C1/C2" contains the numbers of replicates for the first and second condition. The other columns list sensitivity and specificity of DEXUS at different I/NI thresholds as the average across 100 data sets. The first, second, and third value in each cell corresponds to a log fold change of 0.5, 1, and 1.5, respectively. The library size was 10^7 for all experiments.

I/NI threshold	0.025		0.05		0.1	
C1/C2	specificity	sensitivity	specificity	sensitivity	specificity	sensitivity
6/6	0.89|0.89|0.89	0.47|0.89|0.97	0.95|0.95|0.95	0.33|0.87|0.96	0.99|0.99|0.99	0.16|0.83|0.95
9/3	0.89|0.89|0.89	0.60|0.91|0.97	0.95|0.95|0.95	0.44|0.89|0.97	0.99|0.99|0.99	0.03|0.76|0.95
10/2	0.89|0.89|0.89	0.59|0.90|0.97	0.95|0.95|0.95	0.14|0.86|0.96	0.98|0.98|0.98	0.02|0.07|0.88
11/1	0.89|0.89|0.89	0.24|0.85|0.95	0.95|0.95|0.95	0.07|0.14|0.85	0.99|0.99|0.99	0.02|0.02|0.02
12/12	0.95|0.95|0.95	0.36|0.88|0.97	0.98|0.98|0.98	0.20|0.84|0.96	1.00|1.00|1.00	0.09|0.79|0.94
18/6	0.95|0.95|0.95	0.56|0.91|0.98	0.98|0.98|0.98	0.38|0.88|0.97	1.00|1.00|1.00	0.01|0.75|0.95
20/4	0.95|0.95|0.95	0.55|0.91|0.98	0.98|0.98|0.98	0.06|0.85|0.96	1.00|1.00|1.00	0.00|0.03|0.89
22/2	0.95|0.95|0.95	0.14|0.85|0.96	0.98|0.98|0.98	0.02|0.07|0.88	1.00|1.00|1.00	0.00|0.01|0.01

Table S17: The performance of DEXUS in terms of sensitivity and specificity in detecting differential expression with unknown conditions. The results are separately reported for the three different fold change categories. The first column "C1/C2" contains the numbers of replicates for the first and second condition. The other columns list sensitivity and specificity of DEXUS at different I/NI thresholds as the average across 100 data sets. The first, second, and third value in each cell corresponds to a log fold change of 0.5, 1, and 1.5, respectively. The library size was 10^8 for all experiments.

S4.3.3 The "Nigerian HapMap" data set

Pickrell *et al.* (2010) sequenced RNA from 69 Nigerian HapMap individuals to study expression quantitative trait loci (eQTLs). The read count data was provided by the ReCount repository (Frazee *et al.* 2011). As in previous experiments, DEXUS was applied to this data with its default parameters and ranked genes according to the I/NI value. The read counts of top-ranked genes and the conditions identified by DEXUS are visualized as a heatmap in Fig. S16.

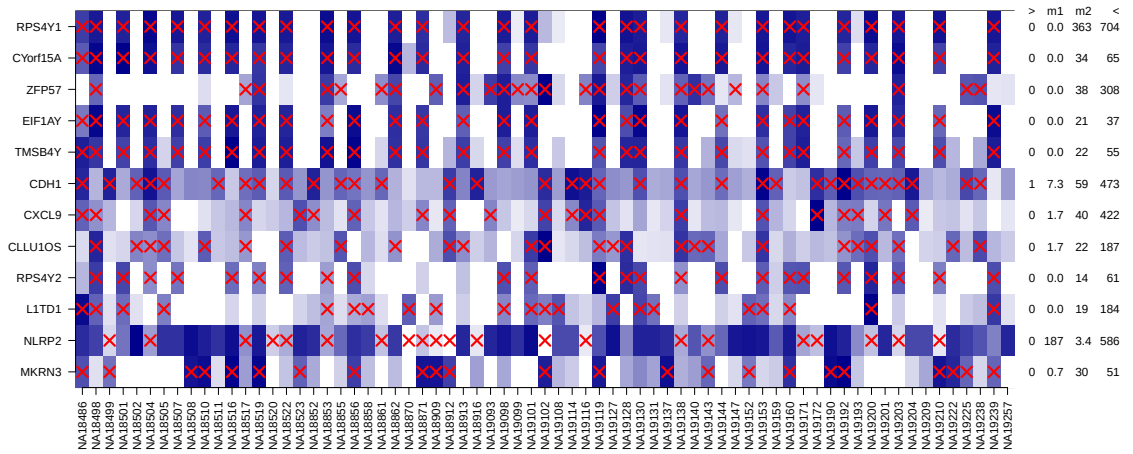

Figure S16: Heatmap of the normalized read counts of the twelve genes with the largest I/NI values for the "Nigerian HapMap" data set. Colors range from white for low expression to blue for high expression. The columns displays different HapMap individuals. The rows show the gene symbols of the top-ranked genes. Red crosses indicate that these samples belong to the minor condition. At the right hand side of the heatmap, each gene is annotated by the minimum (">"), the median of two conditions ("m1" and "m2"), and the maximum ("<") read count.

Five out of the twelve top-ranked genes are located on the Y chromosome (RPS4Y1, CYorf15A, EIF1AY, TMSB4Y, RPS4Y2). For these genes the conditions that DEXUS identified are related to the sex. For four of the twelve top-ranked genes at least one eQTL is known. For ZFP57 the eQTL is the single nucleotide polymorphism (SNP) rs1736924 with a minor allele frequency (MAF) of 0.14 (Pickrell *et al.* 2010). CDH1 has 6 eQTLs, one of which is SNP rs7196495 with a MAF of 0.22 (Zeller *et al.* 2010). CLLU1OS possesses the eQTL SNP rs12580153 with a MAF of 0.19 (Dimas *et al.* 2009). L1TD1 has 2 eQTLs, one of which is SNP rs12137088 with a MAF 0.30 (Veyrieras *et al.* 2008). Since the MAFs are large, it is plausible that the minor alleles are observed in the HapMap data set and that they lead to differential expressions of the associated genes. The conditions that were found by DEXUS correspond to the alleles of corresponding SNPs.

The HapMap samples are lymphoblastoid cells, therefore we confirmed that the genes detected by DEXUS are indeed expressed in lymphoblastoid cell lines. The gene NLRP2, ranked 11th by DEXUS, is expressed in lymphoblastoid cells but with large variability (Halbritter *et al.* 2011) as shown in Figure S17. NLRP2 is expressed in the HapMap individuals but in some very low. Schlattl *et al.* (2011) identified a copy number variable region that covers NLRP2 partially and may be the cause of differential expression. Therefore, the conditions that DEXUS identified for NLRP2 seem to be related to copy number states of the samples. Copy number states might also

cause differential expression of MKRN3 that was ranked 12th by DEXUS. Pinto *et al.* (2007) and Redon *et al.* (2006) identified a copy number variable region covering MKRN3. However, the interpretation of MKRN3's conditions is difficult since only the paternal copy of MKRN3 is expressed.

We analyzed DEXUS' I/NI value ranking of transcripts. Genes on the X chromosome were ranked significantly higher than other genes ($p = 3.0\text{e-}12$) which can be explained by sex related transcripts. An analog test for the Y chromosome was not significant, because too few genes were expressed. However, as already mentioned, out of the twelve top-ranked genes, five are located on the Y chromosome. At an I/NI threshold of 0.1, DEXUS called 366 differentially expressed genes. Gene enrichment analysis showed that the called genes are associated with the extracellular region. Significant GO terms were "extracellular space", "extracellular region part", and "extracellular region" with $p = 2.2\text{e-}5$, $p = 8.8\text{e-}5$, and $p = 0.01$, respectively (p-values were corrected for multiple testing by the Benjamini-Hochberg procedure). These GO terms are in agreement with characteristics of lymphoblastoid cells. Tab. S18 shows all significant GO terms of this data set.

Term	Count	p-value
GO:0005615 extracellular space	35	2e-5
GO:0044421 extracellular region part	41	9e-5
GO:0005529 sugar binding	16	0.001
GO:0042379 chemokine receptor binding	9	0.001
GO:0008009 chemokine activity	9	0.001
GO:0005125 cytokine activity	15	0.003
GO:0007267 cell-cell signaling	30	0.004
GO:0005886 plasma membrane	95	0.011
GO:0031982 vesicle	27	0.011
GO:0031988 membrane-bounded vesicle	24	0.012
GO:0044459 plasma membrane part	63	0.014
GO:0030246 carbohydrate binding	19	0.014
GO:0030054 cell junction	22	0.016
GO:0005576 extracellular region	59	0.016
GO:0031410 cytoplasmic vesicle	25	0.018
GO:0016023 cytoplasmic membrane-bounded vesicle	22	0.027
GO:0003002 regionalization	14	0.044
GO:0005865 striated muscle thin filament	4	0.046
GO:0008021 synaptic vesicle	7	0.049

Table S18: Significant GO terms of the differentially expressed genes of the "Nigerian HapMap" data set. The first column presents the GO identifier and the short name of the GO term. The second column the number of genes that belong to that GO term, and the last column shows the p-values after Benjamini-Hochberg correction.

S4.3.4 The "European HapMap" data set

We analyzed the RNA-Seq data of 60 individuals from the HapMap cohort from Montgomery *et al.* (2010) which were provided by the ReCount repository (Frazee *et al.* 2011). Again, DEXUS

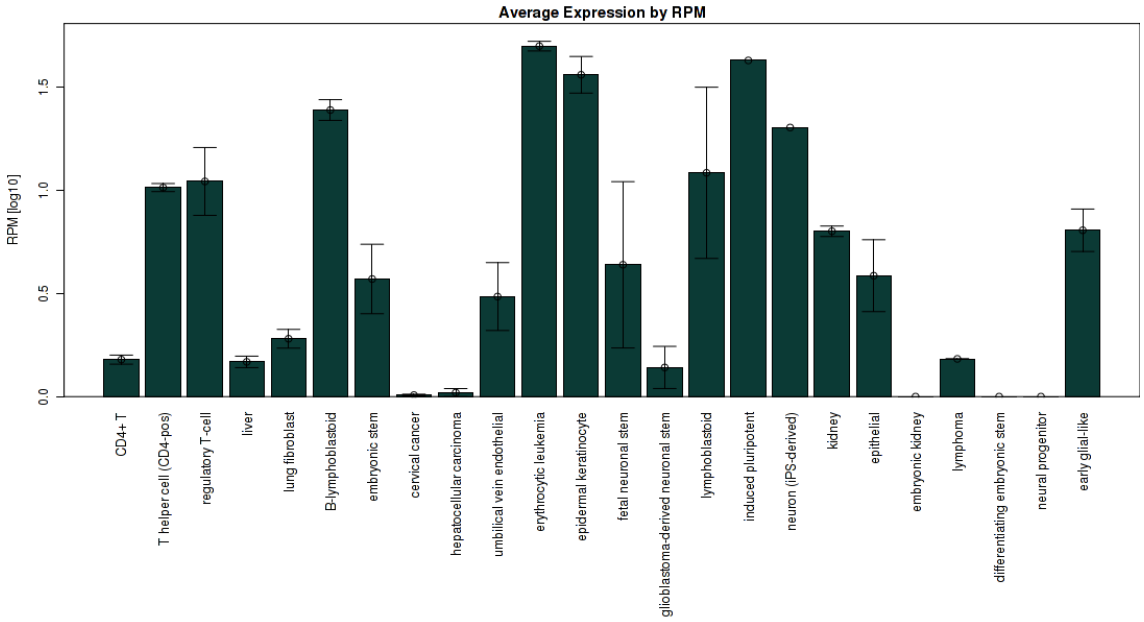

Figure S17: Expression values of gene NLRP2 in log10 RPM (reads per million mapped reads) format as provided by the GeneProf data base (Halbritter *et al.* 2011). The data is taken from 113 public data sets. The gene NLRP2 is expressed and highly variable in lymphoblastoid cell lines.

was applied to these data with its default parameters and ranked genes according to the I/NI value. The read counts of top-ranked genes and the identified conditions are visualized as a heatmap in Fig. S18.

RPS4Y1 is the gene with the largest I/NI value, differentially expressed between males and females, and located on the Y chromosome. The genes CYorf15A and TMSB4Y, ranked fourth and fifth according to the I/NI value, are located on the Y chromosome, too. As in "Nigerian HapMap" data set, ZFP57 was detected as being differentially expressed. Two of the twelve top-ranked genes have eQTLs. CLLU1OS has as eQTL the SNP rs12580153 with a minor allele frequency of 0.19 (Dimas *et al.* 2009). POU2F3 has as eQTL the SNP rs2847497 with a MAF of 0.14 (Schadt *et al.* 2008). As in the "Nigerian HapMap" data set some top ranked genes, like NLRP2 (rank 11, again), were differentially expressed due to variable copy numbers (Schlattl *et al.* 2011). Again the conditions are associated with copy numbers. For the genes T, PRSS21, and RASSF10 DEXUS identified two conditions the interpretation of which is yet to be found. We could neither interpret the conditions by sex, nor allele, nor copy number state. DEXUS hints at a new source of variability in gene expression. The second ranked gene T, the third ranked gene PRSS21, and the twelfth ranked gene RASSF10 are expressed in B-lymphoblastoid cells (Wu *et al.* 2009; The ENCODE Project Consortium 2012), the cell type of the HapMap samples. The high expression variability of T and PRSS21 in the B-lymphoblastoid cell line was already reported by the ENCODE Project (The ENCODE Project Consortium 2012). The ENCODE Project expression values for the genes T, PRSS21, and RASSF10 are visualized in Fig. S19, S20, and S21.

When analyzing the I/NI value ranking, we found that genes on the X chromosome are ranked

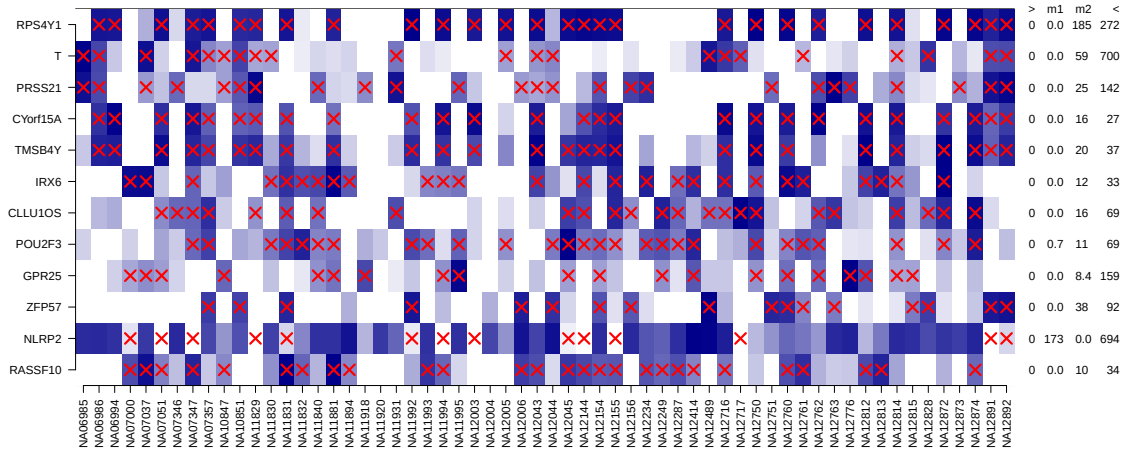

Figure S18: Heatmap of the read counts of the twelve genes with the largest I/NI values for the "European HapMap" data set. Colors range from white for low expression to blue for high expression. The columns displays different HapMap individuals. The rows show the genes symbols of the top-ranked genes. Red crosses indicate that these samples belong to the minor condition. At the right hand side of the heatmap, each gene is annotated by the minimum ("$>$"), the median of two conditions ("m1" and "m2"), and the maximum ("$<$") read count.

significantly higher ($p = 8.0\text{e-}6$, Wilcoxon test). The analogous test for the Y chromosome was not significant as too few genes were expressed. However, three out of the twelve top-ranked genes with the largest I/NI value are located on the Y chromosome.

At an I/NI threshold of 0.1, DEXUS called 680 differentially expressed genes. Gene enrichment analysis showed that the called genes are associated with ion transport. Significant GO terms were "ion transport", "potassium ion transport", and "plasma membrane part" with $p = 0.04$, $p = 4.3\text{e-}03$, and $p = 0.027$, respectively (p-values were corrected for multiple testing by the Benjamini-Hochberg procedure). These GO terms are in agreement with characteristics of lymphoblastoid cells. Tab. S19 shows all significant GO terms of this data set.

S4.3.5 The "Primate Liver" data set

Blekhman *et al.* (2010) investigated the differences in alternative splicing in liver tissue between humans, chimpanzees and rhesus macaques. For this purpose, they sequenced the RNA of three male and three female liver samples from each species. They focused on the expression values of exons that had reliably determined orthologs in all species. Read counts for exons were provided by the original publication which used gene models from Ensemble (Release 50). After pooling technical replicates, DEXUS ranked genes according to the I/NI value using its default parameters. The ten top-ranked genes are visualized in Fig. S22 which shows strong differential expression between the species. For all these genes DEXUS determined one of the three species as minor condition without having been provided with this information. Interestingly, out of the ten top-ranked genes, six are human pseudogenes: AC010591.10, AC105383.3, AC093874.3-1, AC105383.3, AL132855.4, and UOX. These genes are inactive in humans because of recent structural rearrangements (Balasubramanian *et al.* 2009). Since the rearrangements are recent, their

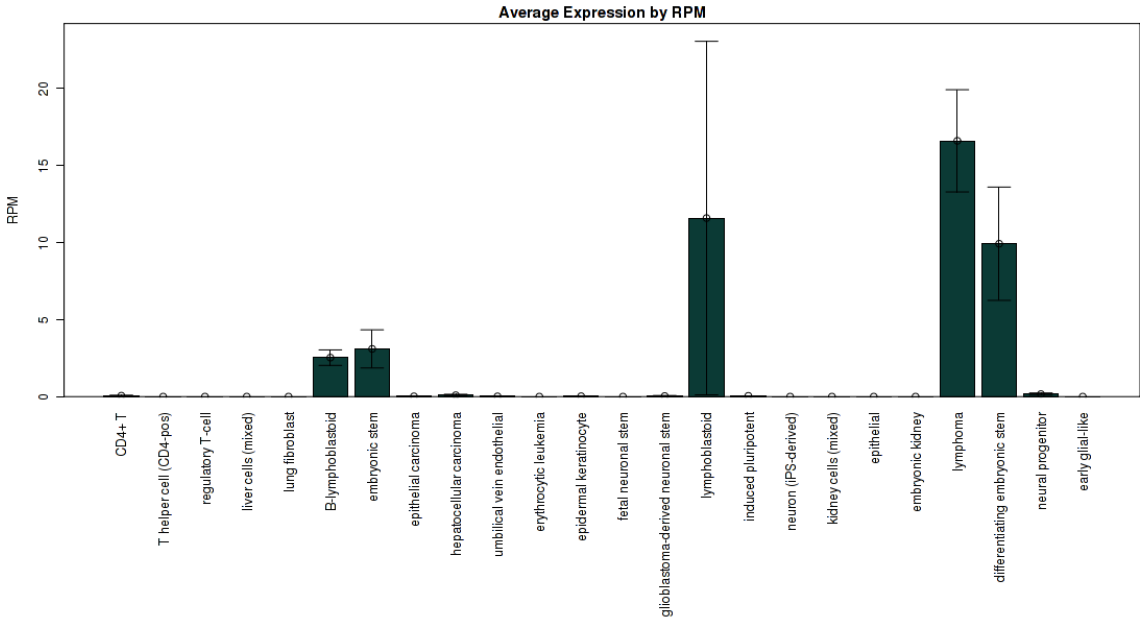

Figure S19: Expression values of gene T in RPM (reads per million mapped reads) format as provided by the GeneProf data base (Halbritter *et al.* 2011). The data is taken from 113 public data sets. The gene T is expressed and highly variable in lymphoblastoid cell lines.

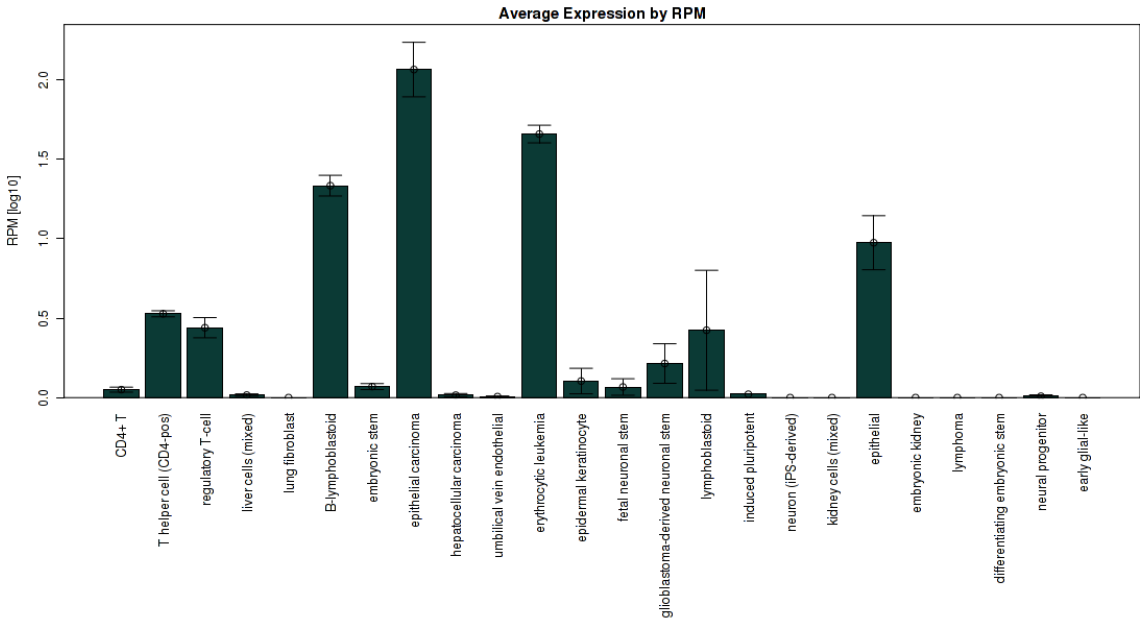

Figure S20: Expression values of PRSS21 in log10 RPM (reads per million mapped reads) format as provided by the GeneProf data base (Halbritter *et al.* 2011). The data is taken from 113 public data sets. The gene PRSS21 is expressed and highly variable in lymphoblastoid cell lines.

Term	Count	p-value
GO:0005261 cation channel activity	28	0.001
GO:0006813 potassium ion transport	20	0.004
GO:0022838 substrate specific channel activity	31	0.005
GO:0022843 voltage-gated cation channel activity	17	0.005
GO:0005267 potassium channel activity	16	0.005
GO:0046873 metal ion transmembrane transporter activity	29	0.005
GO:0022803 passive transmembrane transporter activity	32	0.005
GO:0005244 voltage-gated ion channel activity	20	0.006
GO:0022832 voltage-gated channel activity	20	0.006
GO:0005216 ion channel activity	30	0.006
GO:0015267 channel activity	32	0.007
GO:0030955 potassium ion binding	16	0.008
GO:0031420 alkali metal ion binding	22	0.008
GO:0022836 gated channel activity	27	0.009
GO:0005249 voltage-gated potassium channel activity	13	0.011
GO:0044459 plasma membrane part	103	0.027
GO:0051254 positive regulation of RNA metabolic process	34	0.030
GO:0034702 ion channel complex	19	0.031
GO:0030001 metal ion transport	33	0.031
GO:0045944 positive regulation of transcription from RNA polymerase II promoter	28	0.032
GO:0051173 positive regulation of nitrogen compound metabolic process	41	0.033
GO:0045941 positive regulation of transcription	38	0.033
GO:0031328 positive regulation of cellular biosynthetic process	43	0.034
GO:0015672 monovalent inorganic cation transport	27	0.034
GO:0009891 positive regulation of biosynthetic process	43	0.035
GO:0045893 positive regulation of transcription, DNA-dependent	34	0.036
GO:0006811 ion transport	48	0.042
GO:0010628 positive regulation of gene expression	39	0.043
GO:0034703 cation channel complex	15	0.046

Table S19: Significant GO terms of the differentially expressed genes of the "European HapMap" data set. The first column presents the GO identifier and the short name of the GO term. The second column the number of genes that belong to that GO term, and the last column shows the p-values after Benjamini-Hochberg's correction.

orthologs can reliably be identified in other primates. Differential expression is detected because these orthologs are still transcribed in chimpanzees or in rhesus macaques.

Many of the ten top-ranked genes are associated with liver pathways. Differential expression of these genes between species might have arisen from different diets. Examples of such genes are the human pseudogene UOX that is required to catalyze the oxidation of uric acid to allantoin in *Macaca mulatta*, ABP1 and GSTM5 which participate in degradation and detoxification pathways, VNN3 which helps to recycle vitamin B5, and CHFR2 which is associated with lipoproteins.

Thresholding the I/NI call at 0.1, DEXUS called 3384 genes (16% of all genes) as differentially expressed. A gene set enrichment analysis found GO-Terms "intrinsic to plasma membrane" ($p = 7.9\text{e-}7$) and "integral to plasma membrane"($p = 4.0\text{e-}6$) to be significant. Thus, genes that encode membrane proteins seem to be more often differentially expressed between species than other genes. Interestingly also "response to extracellular stimulus", "response to nutrient", and

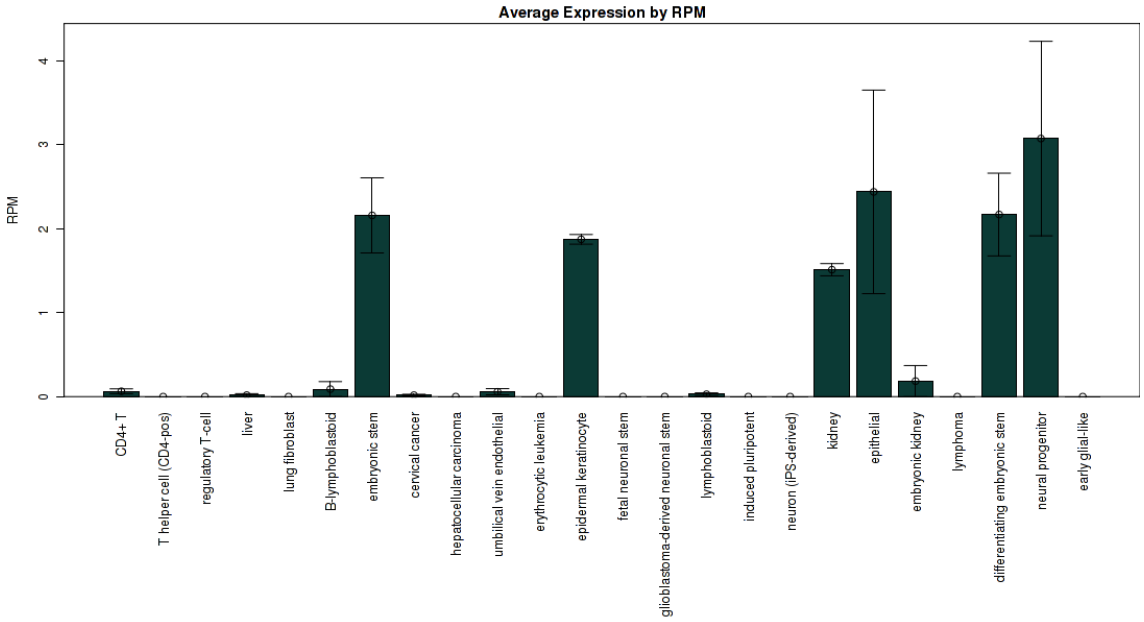

Figure S21: Expression values of gene RASSF10 in RPM (reads per million mapped reads) format as provided by the GeneProf data base (Halbritter *et al.* 2011). The data is taken from 113 public data sets. The gene T is expressed and highly variable in lymphoblastoid cell lines.

"response to nutrient levels" were significant (all p-values below 7.6e-5), which supports the hypothesis that some genes are differentially expressed due to the different diets of the species. All p-values were corrected by the Benjamini-Hochberg procedure.

S4.3.6 The "Maize Leafs" data set

Li *et al.* (2010) studied the developmental dynamics of the maize transcriptome using RNA-Seq data from different locations of maize plant leafs. For each location two biological replicates were sequenced with Illumina's Genome Analyzer II. The reads were mapped to the TE-masked *Zea maize* ZmB73 reference genome version 2 (AGPv2), release 5a using the GSNAP splicing short read mapper (Wu and Nacu 2010). We counted the overlaps between mapped reads and the *Zea maize* gene definitions from the Ensemble Plants database (Release 14). Reads that have multiple possible alignments or that overlap with more than one gene are discarded. DEXUS was applied to this data with its default parameters.

Fig. S23 shows the genes with the largest I/NI value and the conditions that were identified by DEXUS. DEXUS found differentially expressed genes between different tissues, therefore distinguished them without having been provided with any information on the tissue type. DEXUS almost always assigned the two replicates to the same condition without knowing replicates or tissue types. Thus, DEXUS assigns conditions very reliable.

Eight of the ten top ranked genes were also measured by microarrays across different tissues of *Zea mays* (Sekhon *et al.* 2011). In this microarray experiment all eight genes show an absolute

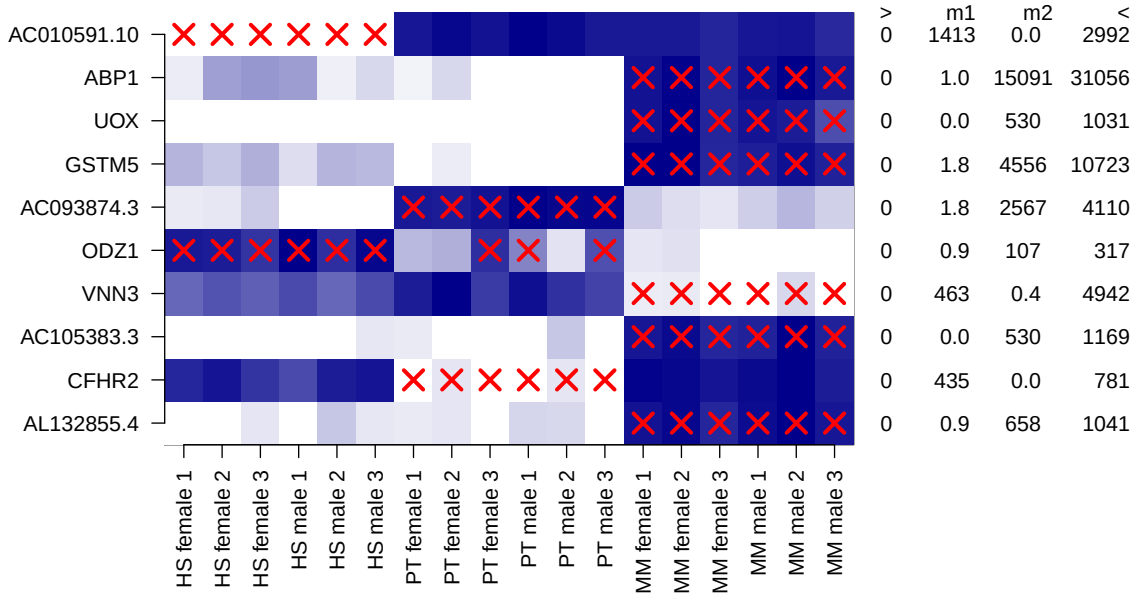

Figure S22: Heatmap of the normalized read counts of the ten genes with the largest I/NI values for the "Primate Liver" data set. Colors range from white for low expression to blue for high expression. The columns give female and male individuals from the three species human *Homo sapiens* (HS), chimpanzee *Pan troglodytes* (PT), and rhesus macaques *Macaca mulatta* (MM). The rows display the gene symbols of the top-ranked genes. Red crosses mark samples that were assigned to the minor condition. At the right hand side of the heatmap, each gene is annotated by the minimum (">"), the median of two conditions ("m1" and "m2"), and the maximum ("<") read count.

log fold change of at least 1 between base and tip. Six of these eight genes show an absolute log fold change greater than 4.

The two remaining genes, GRMZM2G331518 and AC213612.3_FG001, were not annotated on the microarray. The function of the top ranked gene GRMZM2G331518 is not known. However, the associated peptide is similar to the defensin-like protein 91 of *Arabidopsis thaliana* that plays a role in immune response. The gene AC213612.3_FG001 was ranked ninth. It is a glycine-rich cell wall structural protein which hints at the fact that cell walls at different locations have different structure.

At a threshold of 0.1 for the I/NI call, DEXUS called 15,756 differentially expressed genes. Gene set enrichment analysis using the R package goseq (Young *et al.* 2010) led to the significant GO terms "chloroplast"($p = 1.8\text{e-}92$), and "plasma membrane" ($p = 1.3\text{e-}34$). Further the GO terms "cytosolic ribosome" ($p = 9.8\text{e-}32$), "chloroplast thylakoid membrane" ($p = 5.4\text{e-}31$), and "chloroplast stroma" ($p = 1.8\text{e-}30$) were significant. All p-values were corrected by the Benjamini-Hochberg procedure. It is plausible that different locations of the maize plant leaf are different with respect to chloroplasts. Moreover the GO term "cell wall" was highly significant ($p = 3.9\text{e-}18$) which supports the above mentioned hypothesis that the cell walls differ at the different locations of the plant leaf.

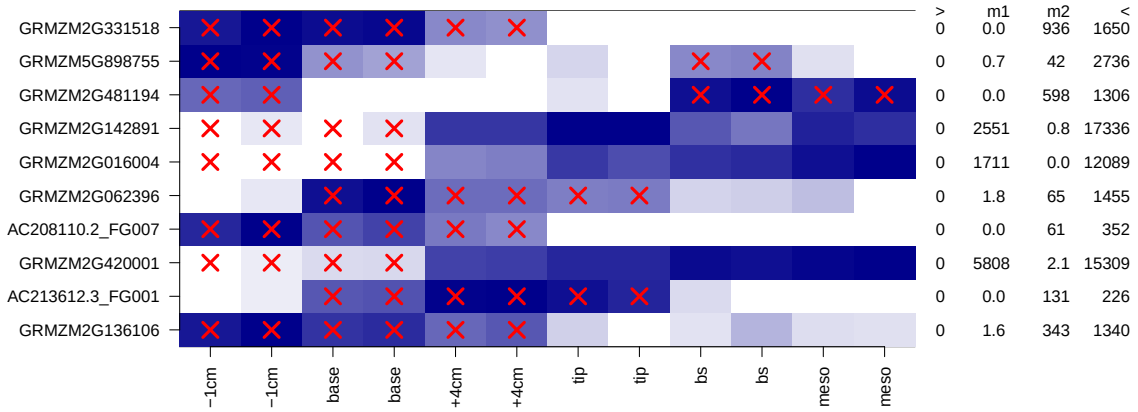

Figure S23: Heatmap of the normalized read counts of the ten genes with the largest DEXUS I/NI values for the "Maize Leafs" data set. Colors range from white for low expression to blue for high expression. The columns show samples from different locations of the maize plant leaf. The rows display the gene symbols of the top-ranked genes. Red crosses indicate that the according samples belong to the minor condition. At the right hand side of the heatmap, each gene is annotated by the minimum (">"), the median of two conditions ("m1" and "m2"), and the maximum ("<") read count.

S4.4 RNA-Seq Data with Subconditions

We demonstrate that DEXUS is capable of detecting subconditions in data sets with known major conditions, which are typically the study conditions. Either (a) the higher level conditions are given or (b) the hierarchy of the conditions is unknown. In both cases we can explain the hierarchy by following model:

$$p(x) \;=\; \sum_{j=1}^{l} \beta_j \sum_{i=1}^{k_j} \alpha_{ij}\, p(x \mid C=j, D_j=i) \quad \text{with} \tag{S75}$$
$$p(x \mid C=j, D_j=i) \;=\; \mathrm{NB}(x; \mu_{ij}, r_{ij})\;,$$

where $\beta_j = p(C=j)$ with $\sum_{j=1}^{l} \beta_j = 1$ are the probabilities for the higher level condition j, C is the random variable for the higher level conditions, D_j is the random variable for the lower level conditions for higher level condition j, and $\alpha_{ij} = p(D_j = i | C = j)$ with $\sum_{i=1}^{k_j} \alpha_{ij} = 1$ is the probability to observe a lower level condition i given the higher level condition j. We obtain the mixture of negative binomials for condition j:

$$\sum_{i=1}^{k_j} \alpha_{ij}\, p(x \mid C=j, D_j=i) \;=\; \sum_{i=1}^{k_j} p(D_j=i \mid C=j)\, p(x \mid C=j, D_j=i) \tag{S76}$$
$$= \sum_{i=1}^{k_j} p(x, D_j=i \mid C=j) \;=\; p(x \mid C=j)\;.$$

We define the probabilities for the lower level conditions

$$\pi_{ij} \;=\; \alpha_{ij}\,\beta_j \;=\; p(D_j=i \mid C=j)\, p(C=j) \;=\; p(D_j=i, C=j) \quad \text{and} \quad \alpha_{ij} = \frac{\pi_{ij}}{\beta_j}\;. \tag{S77}$$

Thus,

$$\beta_j \;=\; \beta_j \sum_{i=1}^{k_j} \alpha_{ij} \;=\; \sum_{i=1}^{k_j} \alpha_{ij}\, \beta_j \;=\; \sum_{i=1}^{k_j} \pi_{ij} \tag{S78}$$

and $\sum_{j=1}^{l} \sum_{i=1}^{j_k} \pi_{ij} = 1$. We have as full model using the π_{ij}:

$$p(x) \;=\; \sum_{j=1,i=1}^{l,k_j} \pi_{ij}\, p(x \mid C = j, D_j = i)\;, \tag{S79}$$

which is just a mixture of negative binomials with index set $\{(i,j)\}$.

The posterior of a condition j after observing read count x is given via the Bayes formula by

$$p(C = j \mid x) \;=\; \frac{p(x \mid C = j)\, p(C = j)}{p(x)} \tag{S80}$$

Using this model we first consider case (a), in which the higher level conditions are given. For each read count x_k, its higher level condition ι is known:

$$p(C = j \mid x_k) \;=\; \begin{cases} 1 & \text{if } j = \iota \\ 0 & \text{otherwise} \end{cases}\;, \tag{S81}$$

and, therefore, the β_j can be approximated:

$$\begin{aligned} \beta_j \;&=\; p(C = j) \;=\; \sum_x p(x)\, p(C = j \mid x) \;=\; \mathrm{E}_x\left(p(C = j \mid x)\right) \\ &\approx\; \frac{1}{N} \sum_{k=1}^{N} p(C = j \mid x_k) \end{aligned} \tag{S82}$$

To estimate the α_{ij}, we perform model selection on the mixture of negative binomials of Eq. (S76) using only the $\boldsymbol{x}_k$ that belong to condition j. If we do model selection for each higher level condition j then all parameters of the hierarchical model are known. We just apply our standard mixture of negative binomials model to each of the higher level conditions j.

Next we consider case (b), in which the higher level conditions are not kown. DEXUS is applied to the full data set using the mixture of negative binomials model in Eq. (S79). Model selection supplies the $\theta_s = \pi_{ij}$, such that we do not know the index j of the π_{ij}. To identify j and θ_s, which belong to the same higher level condition j, the lower level conditions s can be joined by agglomerative clustering. In such a way we obtain a hierarchy of the conditions. The variable β_j is obtained by summing up the θ_s which belong to higher level condition j, that are the $\theta_s = \pi_{ij}$: $\beta_j = \sum_{i=1}^{k_j} \pi_{ij}$. Then the α_{ij} are obtained by $\alpha_{ij} = \frac{\pi_{ij}}{\beta_j}$. This approach is just our standard mixture of negative binomials model applied to all data followed by a agglomerative clustering to obtain a hierarchy of conditions.

In our experiments, in which the higher level conditions were known, both approaches led to similar results, as we show in Figure S24. This figure shows using four genes of the "Primate Liver" data set as exemplars for a hierarchy of conditions (or groups with subgroups of samples).

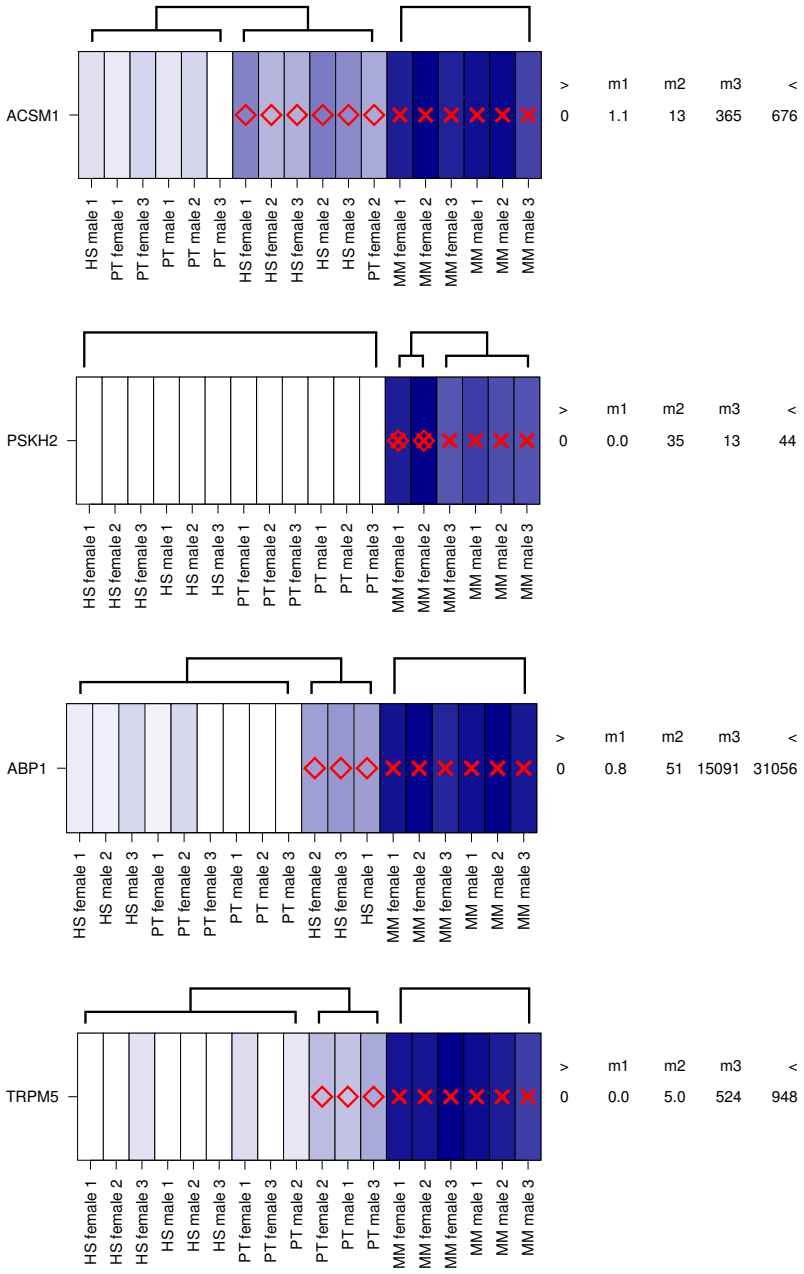

Figure S24: Heatmap of the normalized read counts of four exemplar genes of the "Primate Liver" data set that contain subconditions. Colors range from white for low expression to blue for high expression. Different individuals are denoted along the x-axis, while genes are denoted by their gene symbols along the y-axis. Red crosses indicate that the according samples belong to the minor condition. Red diamonds indicate that the according samples belong to the minor subcondition. At the right hand side of the heatmap, each gene is annotated by the minimum ("$>$"), the median of three conditions ("m1", "m2", and "m3"), and the maximum ("$<$") read count. The three conditions found in both cases if the high level conditions are unknown and if the high level conditions are known. In the former case, a DEXUS model with multiple unknown conditions finds the three conditions and then two of them are merged by agglomerative clustering to a high level condition. In the latter case DEXUS is applied to data from one high level condtion and then finds the subconditions.

S5 Additional Information

S5.1 Data set overview

Tab. S20 gives an overview over the data sets used in the manuscript and this supplement. Except for the "Primate Liver" data set, all count matrices were downloaded from the ReCount (Frazee *et al.* 2011) repository. The count matrix for the "Primate Liver" data was taken from GEO (Accession number GSE17274). The raw counts were normalized using UpperQuartile normalization.

Name	**Reference**	**Organism**	**S**	**R**	**Counts**	**C**
Primate Liver	Blekhman *et al.* (2010)	*H.s./M.m./P.T.*	18	2	Pub.	U
European HapMap	Montgomery *et al.* (2010)	*H. sapiens*	60	1	ReCount	U
Nigerian HapMap	Pickrell *et al.* (2010)	*H. sapiens*	69	1	ReCount	U
Worm	Hillier *et al.* (2009)	*C. elegans*	46	–	ReCount	U
Yeast	Nagalakshmi *et al.* (2008)	*S. cervisiae*	4	1	ReCount	U
Mice Strains	Bottomly *et al.* (2011)	*M. musculus*	21	1	ReCount	K
Maize Leafs	Li and Tibshirani (2011)	*Z. mays*	12	2	Mapped	U

Table S20: Overview of the data sets used in the manuscript. "Name" gives the name used for the data set in the manuscript, "Reference" lists the according publications, "Organism" gives the organism from which the RNA-Seq data was obtained ("H.s./M.m./P.T." means *Homo sapiens*/*Pan troglodytes*/*Macaca mulatta* which is human, chimpanzee, and rhesus macaques), "S" reports the number of samples, "R" gives the number of replicates for each condition, "Counts" reports the way the read counts are obtained ("Pub." means from the publication, "ReCount" means the mapped reads are counted per transcript, "Mapped" we preprocessed the data ourselves (read mapping and counting), the column "C" lists whether the conditions were known (K) or unknown (U).

S5.2 Alternative Way to Derive the Update Rule for Mixture Weights

The update rule Eq. (S57) can be obtained in an alternative way. The Dirichlet distribution is conjugate to the multinomial distribution, that is the posterior $p(\boldsymbol{\alpha} \mid \{\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_k, \ldots, \boldsymbol{\alpha}_N\})$ is a Dirichlet distribution as is the prior $p(\boldsymbol{\alpha})$ with $\boldsymbol{\alpha}_k = p(\boldsymbol{\alpha} \mid x_k)$. The Dirichlet prior $p(\boldsymbol{\alpha}) = \mathrm{D}(\boldsymbol{\alpha}; \boldsymbol{\gamma})$ with parameters $\boldsymbol{\gamma}$ leads to the conjugate posterior $p(\boldsymbol{\alpha} \mid \{\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_k, \ldots, \boldsymbol{\alpha}_N\})$ with parameters

$$\hat{\boldsymbol{\gamma}} \;=\; \boldsymbol{\gamma} \;+\; \sum_{k=1}^{N} \boldsymbol{\alpha}_k \;=\; \boldsymbol{\gamma} \;+\; N\,\boldsymbol{\alpha}\,, \tag{S83}$$

where we used Eq. (S38). We obtain update rule Eq. (S57) from Eq. (S83) component-wise by first replacing the unknown values α_{ik} by their estimates $\tilde{\alpha}_{ik}$ and then computing the posterior's mode because we search for the maximum posterior.

S5.3 Posteriors in Our Framework

In our Bayesian framework, we introduced two different posterior distributions: (i) in Eq. (S37) the posterior $\alpha_{ik} = p(i \mid x_k, \boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{r})$ of the data x_k arising from the i-th condition with prior $\alpha_i = p(i)$ — this posterior is defined for fixed model parameters $(\boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{r})$; (ii) in Eq. (S20) the parameter posterior $p(\boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{r} \mid x)$ with priors $p(\boldsymbol{\alpha})$, $p(\boldsymbol{\mu})$, and $p(\boldsymbol{r})$ — this posterior is the objective that we maximize during model selection. In previous subsection we introduced another posterior, the posterior $p(\boldsymbol{\alpha} \mid \{\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_k, \ldots, \boldsymbol{\alpha}_N\})$ used in Eq. (S83) with prior $p(\boldsymbol{\alpha})$. In contrast to (ii) this posterior is not the posterior for the full mixture of negative binomials model but only for the multinomial distribution given by $\boldsymbol{\alpha}$, where the posteriors $\alpha_{ik} = p(i \mid x_k, \boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{r})$ from (i) serve as data.

S5.4 Maximum A Posterior for the Size Parameter of a Negative Binomial

The maximum likelihood solution r_{ML} Eq. (S17) for the negative binomial tends to overestimate the true size parameter r (Piegorsch 1990). Therefore we introduce a prior $p(r)$ on r, which prefers small r-values. An appropriate prior distribution is the exponential distribution $p(r) = \mathrm{EXP}(r) = \eta e^{-\eta r}$.

Using a Bayesian approach, we obtain the posterior $p(r \mid \boldsymbol{x})$ for a data point x as the normalized product between the likelihood $p(\boldsymbol{x} \mid r)$ and the prior $p(r)$. We want to maximize the posterior

$$p(r \mid \boldsymbol{x}) \;=\; \frac{p(\boldsymbol{x} \mid r)\, p(r)}{\int p(\boldsymbol{x} \mid r)\, p(r)\, dr}\,. \tag{S84}$$

The logarithm of the posterior is

$$\log p(r \mid \boldsymbol{x}) \;=\; \log p(\boldsymbol{x} \mid r) \;+\; \log p(r) \;-\; \log(c(\boldsymbol{x}))\,, \tag{S85}$$

where $c(\boldsymbol{x})$ is a function of $\boldsymbol{x}$. Using the negative binomial distribution

$$p(\boldsymbol{x} \mid r) \;=\; \prod_{k=1}^{N} \frac{\Gamma(x_k + r)}{\Gamma(x_k + 1)\Gamma(r)} \left(\frac{\mu}{\mu + r}\right)^{x_k} \left(\frac{r}{\mu + r}\right)^{r} \tag{S86}$$

and the exponential prior $\mathrm{EXP}(r) = \eta e^{-\eta r}$ on r, we obtain

$$\begin{aligned}\log p(r \mid \boldsymbol{x}) \;=\; \sum_{k=1}^{N} \Bigg[&\log\left(\Gamma(x_k \;+\; r)\right) \;-\; \log\left(\Gamma(x_k \;+\; 1)\right) \;-\; \log\left(\Gamma(r)\right) \;+\; r\, \log\left(\frac{r}{\mu + r}\right) \\ &+\; x_k\, \log\left(\frac{\mu}{\mu + r}\right)\Bigg] \;+\; \log(\eta) \;-\; \eta\, r \;-\; \log(c(\boldsymbol{x}))\,.\end{aligned} \tag{S87}$$

In order to maximize the posterior, we set the derivative with respect to r to zero:

$$\frac{\partial}{\partial r} \log(p(r \mid \boldsymbol{x})) \;=\; \sum_{k=1}^{N} \psi(x_k + r) \;-\; N\, \psi(r) \;+\; N\, \log\left(\frac{r}{\mu + r}\right) \;-\; \eta \;=\; 0\,, \tag{S88}$$

where ψ is the digamma function. We call the solution of the above equation "maximum a posterior estimator" r_{MAP} for the size parameter of the negative binomial distribution. Note that this is identical to the maximum-likelihood solution Eq. (S17) without prior except for the additional term $-\eta$. η is the parameter of the exponential prior.

Note the similarity of Eq. (S88) for a single negative binomial distribution to Eq. (S49) for the whole mixture model. The difference is that, for the whole mixture model, each data point is weighted by its contribution to component i, that is, $\tilde{\alpha}_{ik}$.

S5.5 Summary of the parameters and input values of DEXUS

S5.5.1 Unknown Conditions

Input values and parameters:

$\boldsymbol{X}$ The input matrix of read counts. Rows are assumed to be genes and columns samples. An entry is the read count of sample k in gene g.

n Number of conditions. For further information see Subsection S3.1. *Default setting:* $n = 2$.

$\boldsymbol{\alpha}^{\mathrm{INIT}}$ The initial values for α_i. For further information see Subsection S3.2.4. *Default setting:* $\alpha_i^{\mathrm{INIT}} = 1/n$.

normalization We implemented "RLE" (relative log expression) that is used by DESeq (Anders and Huber 2010) and "UpperQuartile" normalization (Bullard *et al.* 2010). *Default setting:* normalization = RLE .

kmeansIter The number of iterations of the kmeans algorithm for initializing. For further information see Subsection S3.2.4. *Default setting:* kmeansIter $= 10$.

cyc The number of cycles of the EM algorithm. Convergence is usually reached after 5 to 10 cycles. For further information see Subsection S3.2.1. *Default setting:* cyc $= 20$.

Hyperparameters:

G The weight of the prior of $\boldsymbol{\alpha}$. The parameter of the Dirichlet distribution is set to $\boldsymbol{\gamma} = (1 + G, 1, ..., 1)$. For further information see Subsection S3.2.1. *Default setting:* $G = 1$.

θ The hyperparameter that governs the setting of the regularization parameter η on the size parameter r. For further information see Subsection S3.1.2 and S3.2.5. *Default setting:* $\theta = 2.5$.

$r_{\max}$ The upper bound for the size parameter r of the negative binomial distribution. Corresponds to a lower bound of $1/r_{\max}$ for the overdispersion. *Default setting:* $r_{\max} = 13.0$.

$\mu_{\min}$ The minimal value for μ_i that is the mean parameter of the negative binomial distribution For further information see Subsection S3.3. *Default setting:* $\mu_{\min} = 0.5$.

S5.5.2 Known Conditions

Input values and parameters:

$\boldsymbol{X}$: The input matrix of read counts. Rows are assumed to be genes and columns samples. An entry is the read count of sample k in gene g.

labels: A vector containing the condition for each sample. Must be the same length as the number of rows of $\boldsymbol{X}$. For further information see Subsection S3.3.1.

normalization: We implemented "RLE" (relative log expression) that is used by DESeq (Anders and Huber 2010) and "UpperQuartile" normalization (Bullard *et al.* 2010). *Default setting:* normalization = RLE .

Hyperparameters:

θ The hyperparameter that governs the setting of the regularization parameter η on the size parameter r. For further information see Subsection S3.1.2 and S3.2.5. *Default setting:* $\theta = 2.5$.

$r_{\max}$ The upper bound for the size parameter r of the negative binomial distribution. Corresponds to a lower bound of $1/r_{\max}$ for the overdispersion. *Default setting:* $r_{\max} = 13.0$.

$\mu_{\min}$ The minimal value for μ_i that is the mean parameter of the negative binomial distribution For further information see Subsection S3.3. *Default setting:* minMu $= 0.5$.

S5.6 Software Details of DEXUS and Experiments

- In case of two known conditions we use the function `nbinomTestForMatrices` of the R package DESeq.
- To detect differential expression for multiple known conditions, DEXUS fits a generalized linear model with the R package statmod.
- The Gaussian clustering method mclust is available as package for the R . We used the most recent stable version mclust 4.0 of the implementations as provided by the original authors.
- For initialization of the EM algorithms the k-means clustering algorithm as implemented in kmeans of the R base package is used.
- We calculated the $\mathrm{AUC}_{\mathrm{ROC}}$ with the function of the R package ROCR (Sing *et al.* 2005) and the $\mathrm{AUC}_{\mathrm{PR}}$ with the algorithm suggested by Davis and Goadrich (2006).
- A function to calculate the maximum a posterior estimator r_{MAP} for the size parameter of a negative binomial is efficiently implemented in the function `getRNBbisection` of the DEXUS software package.

S6 Performance Tables

I/NI threshold 0.025								
C1/C2	precision	FDR	recall	F-score	accuracy	specificity	sensitivity	b. accuracy
6/6	0.64	0.36	0.33	0.43	0.78	0.94	0.33	0.63
9/3	0.67	0.33	0.36	0.47	0.79	0.94	0.36	0.65
10/2	0.66	0.34	0.36	0.47	0.79	0.94	0.36	0.65
11/1	0.63	0.37	0.32	0.42	0.78	0.94	0.32	0.63
12/12	0.72	0.28	0.28	0.40	0.77	0.96	0.28	0.62
18/6	0.75	0.25	0.33	0.46	0.79	0.96	0.33	0.65
20/4	0.76	0.24	0.34	0.47	0.79	0.96	0.34	0.65
22/2	0.73	0.27	0.30	0.42	0.78	0.96	0.30	0.63
Mean	**0.70**	**0.30**	**0.33**	**0.44**	**0.79**	**0.95**	**0.33**	**0.64**

I/NI threshold 0.05								
C1/C2	precision	FDR	recall	F-score	accuracy	specificity	sensitivity	b. accuracy
6/6	0.70	0.30	0.31	0.43	0.79	0.96	0.31	0.63
9/3	0.72	0.28	0.34	0.46	0.80	0.96	0.34	0.65
10/2	0.71	0.29	0.33	0.45	0.80	0.96	0.33	0.64
11/1	0.62	0.38	0.22	0.32	0.77	0.96	0.22	0.59
12/12	0.82	0.18	0.25	0.39	0.78	0.98	0.25	0.62
18/6	0.84	0.16	0.30	0.44	0.79	0.98	0.30	0.64
20/4	0.83	0.17	0.29	0.42	0.79	0.98	0.29	0.63
22/2	0.75	0.25	0.17	0.28	0.76	0.98	0.17	0.57
Mean	**0.75**	**0.25**	**0.28**	**0.40**	**0.78**	**0.97**	**0.28**	**0.62**

I/NI threshold 0.1								
C1/C2	precision	FDR	recall	F-score	accuracy	specificity	sensitivity	b. accuracy
6/6	0.80	0.20	0.28	0.41	0.80	0.98	0.28	0.63
9/3	0.80	0.20	0.28	0.41	0.80	0.98	0.28	0.63
10/2	0.73	0.27	0.19	0.31	0.78	0.98	0.19	0.58
11/1	0.39	0.61	0.05	0.08	0.74	0.98	0.05	0.51
12/12	0.92	0.08	0.22	0.36	0.78	0.99	0.22	0.61
18/6	0.92	0.08	0.23	0.37	0.79	0.99	0.23	0.61
20/4	0.89	0.11	0.15	0.26	0.76	0.99	0.15	0.57
22/2	0.43	0.57	0.01	0.03	0.73	0.99	0.01	0.50
Mean	**0.73**	**0.27**	**0.18**	**0.28**	**0.77**	**0.98**	**0.18**	**0.58**

I/NI threshold 0.15								
C1/C2	precision	FDR	recall	F-score	accuracy	specificity	sensitivity	b. accuracy
6/6	0.87	0.13	0.25	0.39	0.80	0.99	0.25	0.62
9/3	0.82	0.18	0.18	0.29	0.78	0.99	0.18	0.58
10/2	0.55	0.45	0.05	0.08	0.75	0.99	0.05	0.52
11/1	0.36	0.64	0.02	0.04	0.75	0.99	0.02	0.50
12/12	0.96	0.04	0.20	0.34	0.78	1.00	0.20	0.60
18/6	0.94	0.06	0.14	0.24	0.76	1.00	0.14	0.57
20/4	0.62	0.38	0.02	0.03	0.73	1.00	0.02	0.51
22/2	0.37	0.63	0.01	0.01	0.73	1.00	0.01	0.50
Mean	**0.69**	**0.31**	**0.11**	**0.18**	**0.76**	**0.99**	**0.11**	**0.55**

Table S21: Results of DEXUS for unknown conditions (two conditions). "C1/C2" reports the number of samples for each condition. Each line represents one experiment that consists of 100 data sets. The column names give the different performance measures. The I/NI thresholds are given in table headings. The library size was 10^6 for all experiments.

I/NI threshold 0.025

C1/C2	precision	FDR	recall	F-score	accuracy	specificity	sensitivity	b. accuracy
6/6	0.69	0.31	0.55	0.61	0.79	0.90	0.55	0.72
9/3	0.71	0.29	0.60	0.65	0.81	0.90	0.60	0.75
10/2	0.71	0.29	0.60	0.65	0.81	0.90	0.60	0.75
11/1	0.68	0.32	0.52	0.59	0.78	0.90	0.52	0.71
12/12	0.79	0.21	0.52	0.62	0.81	0.94	0.52	0.73
18/6	0.81	0.19	0.59	0.68	0.84	0.94	0.59	0.77
20/4	0.81	0.19	0.59	0.68	0.84	0.94	0.59	0.77
22/2	0.78	0.22	0.50	0.61	0.81	0.94	0.50	0.72
Mean	**0.75**	**0.25**	**0.56**	**0.64**	**0.81**	**0.92**	**0.56**	**0.74**

I/NI threshold 0.05

C1/C2	precision	FDR	recall	F-score	accuracy	specificity	sensitivity	b. accuracy
6/6	0.79	0.21	0.52	0.62	0.81	0.94	0.52	0.73
9/3	0.80	0.20	0.56	0.66	0.83	0.94	0.56	0.75
10/2	0.78	0.22	0.51	0.62	0.81	0.94	0.51	0.72
11/1	0.68	0.32	0.30	0.42	0.75	0.94	0.30	0.62
12/12	0.89	0.11	0.47	0.62	0.82	0.98	0.47	0.72
18/6	0.90	0.10	0.53	0.67	0.84	0.98	0.53	0.75
20/4	0.89	0.11	0.48	0.62	0.83	0.98	0.48	0.73
22/2	0.82	0.18	0.26	0.40	0.76	0.98	0.26	0.62
Mean	**0.82**	**0.18**	**0.45**	**0.58**	**0.81**	**0.96**	**0.45**	**0.71**

I/NI threshold 0.1

C1/C2	precision	FDR	recall	F-score	accuracy	specificity	sensitivity	b. accuracy
6/6	0.89	0.11	0.46	0.61	0.82	0.98	0.46	0.72
9/3	0.89	0.11	0.44	0.59	0.82	0.98	0.44	0.71
10/2	0.83	0.17	0.27	0.41	0.77	0.98	0.27	0.62
11/1	0.39	0.61	0.04	0.07	0.70	0.98	0.04	0.51
12/12	0.96	0.04	0.42	0.59	0.82	0.99	0.42	0.71
18/6	0.96	0.04	0.42	0.58	0.82	0.99	0.42	0.71
20/4	0.94	0.06	0.24	0.39	0.77	0.99	0.24	0.62
22/2	0.42	0.58	0.01	0.02	0.70	0.99	0.01	0.50
Mean	**0.78**	**0.22**	**0.29**	**0.41**	**0.78**	**0.98**	**0.29**	**0.64**

I/NI threshold 0.15

C1/C2	precision	FDR	recall	F-score	accuracy	specificity	sensitivity	b. accuracy
6/6	0.94	0.06	0.42	0.58	0.82	0.99	0.42	0.71
9/3	0.90	0.10	0.25	0.39	0.77	0.99	0.25	0.62
10/2	0.57	0.43	0.03	0.07	0.70	0.99	0.03	0.51
11/1	0.37	0.63	0.02	0.03	0.70	0.99	0.02	0.50
12/12	0.98	0.02	0.39	0.56	0.82	1.00	0.39	0.70
18/6	0.97	0.03	0.23	0.38	0.77	1.00	0.23	0.62
20/4	0.65	0.35	0.01	0.02	0.70	1.00	0.01	0.50
22/2	0.35	0.65	0.00	0.01	0.70	1.00	0.00	0.50
Mean	**0.72**	**0.28**	**0.17**	**0.26**	**0.75**	**0.99**	**0.17**	**0.58**

Table S22: Results of DEXUS for unknown conditions (two conditions). "C1/C2" reports the number of samples for each condition. Each line represents one experiment that consists of 100 data sets. The column names give the different performance measures. The I/NI thresholds are given in table headings. The library size was 10^7 for all experiments.

I/NI threshold 0.025

C1/C2	precision	FDR	recall	F-score	accuracy	specificity	sensitivity	b. accuracy
6/6	0.76	0.24	0.78	0.77	0.86	0.89	0.78	0.83
9/3	0.77	0.23	0.83	0.80	0.87	0.89	0.83	0.86
10/2	0.77	0.23	0.82	0.79	0.87	0.89	0.82	0.86
11/1	0.73	0.27	0.68	0.70	0.83	0.89	0.68	0.79
12/12	0.85	0.15	0.74	0.79	0.88	0.94	0.74	0.84
18/6	0.86	0.14	0.82	0.84	0.91	0.95	0.82	0.88
20/4	0.86	0.14	0.81	0.84	0.90	0.95	0.81	0.88
22/2	0.84	0.16	0.65	0.73	0.86	0.95	0.65	0.80
Mean	**0.80**	**0.20**	**0.76**	**0.78**	**0.87**	**0.92**	**0.76**	**0.84**

I/NI threshold 0.05

C1/C2	precision	FDR	recall	F-score	accuracy	specificity	sensitivity	b. accuracy
6/6	0.86	0.14	0.72	0.78	0.88	0.95	0.72	0.84
9/3	0.87	0.13	0.77	0.81	0.90	0.95	0.77	0.86
10/2	0.85	0.15	0.66	0.74	0.86	0.95	0.66	0.80
11/1	0.75	0.25	0.35	0.48	0.77	0.95	0.35	0.65
12/12	0.94	0.06	0.67	0.78	0.89	0.98	0.67	0.82
18/6	0.95	0.05	0.74	0.83	0.91	0.98	0.74	0.86
20/4	0.94	0.06	0.63	0.75	0.87	0.98	0.63	0.80
22/2	0.89	0.11	0.32	0.48	0.78	0.98	0.32	0.65
Mean	**0.88**	**0.12**	**0.61**	**0.71**	**0.86**	**0.97**	**0.61**	**0.79**

I/NI threshold 0.1

C1/C2	precision	FDR	recall	F-score	accuracy	specificity	sensitivity	b.accuracy
6/6	0.95	0.05	0.65	0.77	0.88	0.99	0.65	0.82
9/3	0.94	0.06	0.58	0.72	0.86	0.99	0.58	0.78
10/2	0.90	0.10	0.32	0.48	0.79	0.98	0.32	0.65
11/1	0.37	0.63	0.02	0.04	0.70	0.99	0.02	0.50
12/12	0.99	0.01	0.61	0.75	0.88	1.00	0.61	0.80
18/6	0.98	0.02	0.57	0.72	0.87	1.00	0.57	0.78
20/4	0.97	0.03	0.31	0.47	0.79	1.00	0.31	0.65
22/2	0.40	0.60	0.01	0.01	0.70	1.00	0.01	0.50
Mean	**0.81**	**0.19**	**0.38**	**0.49**	**0.81**	**0.99**	**0.38**	**0.69**

I/NI threshold 0.15

C1/C2	precision	FDR	recall	F-score	accuracy	specificity	sensitivity	b. accuracy
6/6	0.98	0.02	0.58	0.73	0.87	0.99	0.58	0.79
9/3	0.96	0.04	0.31	0.47	0.79	0.99	0.31	0.65
10/2	0.57	0.43	0.02	0.04	0.70	0.99	0.02	0.51
11/1	0.35	0.65	0.01	0.01	0.70	0.99	0.01	0.50
12/12	1.00	0.00	0.56	0.72	0.87	1.00	0.56	0.78
18/6	0.99	0.01	0.30	0.46	0.79	1.00	0.30	0.65
20/4	0.64	0.36	0.01	0.01	0.70	1.00	0.01	0.50
22/2	0.35	0.65	0.00	0.00	0.70	1.00	0.00	0.50
Mean	**0.73**	**0.27**	**0.22**	**0.31**	**0.76**	**1.00**	**0.22**	**0.61**

Table S23: Results of DEXUS for unknown conditions (two conditions). "C1/C2" reports the number of samples for each condition. Each line represents one experiment that consists of 100 data sets. The column names give the different performance measures. The I/NI thresholds are given in table headings. The library size was 10^8 for all experiments.

References

Anders, S. and Huber, W. (2010). Differential expression analysis for sequence count data. *Genome Biology*, **11**(10), R106.

Anscombe, F. J. (1950). Sampling theory of the negative binomial and logarithmic series distributions. *Biometrika*, **37**(3/4), 358–382.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. (2000). Gene ontology: tool for the unification of biology. *Nature Genetics*, **25**(1), 25–29.

Auer, P. L. and Doerge, R. W. (2011). A Two-Stage Poisson Model for Testing RNA-Seq Data. *Statistical applications in genetics and molecular biology*, **10**(1), 1–26.

Balasubramanian, S., Zheng, D., Liu, Y.-J., Gang Fang, A. F., Carriero, N., Robilotto, R., and Philip Cayting, M. G. (2009). Comparative analysis of processed ribosomal protein pseudogenes in four mammalian genomes. *Genome Biology*, **10**, R2.

Bean, M. A. (2001). *Probability: The Science of Uncertainty: with Applications to Investments, Insurance, and Engineering*. American Mathematical Society.

Blekhman, R., Marioni, J. C., Zumbo, P., Stephens, M., and Gilad, Y. (2010). Sex-specific and lineage-specific alternative splicing in primates. *Genome Research*, **20**(2), 180–189.

Bottomly, D., Walter, N. A. R., Hunter, J. E., Darakjian, P., Kawane, S., Buck, K. J., Searles, R. P., Mooney, M., McWeeney, S. K., and Hitzemann, R. (2011). Evaluating gene expression in C57BL/6J and DBA/2J mouse striatum using RNA-Seq and microarrays. *PLoS One*, **6**(3), e17820.

Bullard, J. H., Purdom, E., Hansen, K. D., and Dudoit, S. (2010). Evaluation of statistical methods for normalization and differential expression in mRNA-seq experiments. *BMC Bioinformatics*, **11**, 94.

Clevert, D.-A., Mitterecker, A., Mayr, A., Klambauer, G., Tuefferd, M., Bondt, A. D., Talloen, W., Göhlmann, H., and Hochreiter, S. (2011). cn.FARMS: a latent variable model to detect copy number variations in microarray data with a low false discovery rate. *Nucleic Acids Research*, **39**(12), e79.

Cumbie, J. S., Kimbrel, J. A., Di, Y., Schafer, D. W., Wilhelm, L. J., Fox, S. E., Sullivan, C. M., Curzon, A. D., Carrington, J. C., Mockler, T. C., and Chang, J. H. (2011). GENE-Counter: A Computational Pipeline for the Analysis of RNA-Seq Data for Gene Expression Differences. *PLoS ONE*, **6**(10), e25279.

Davis, J. and Goadrich, M. (2006). The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd international conference on Machine learning*, ICML '06, pages 233–240, New York, NY, USA. ACM.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, **39**(1), 1–38.

Dimas, A. S., Deutsch, S., Stranger, B. E., Montgomery, S. B., Borel, C., Attar-Cohen, H., Ingle, C., Beazley, C., Arcelus, M. G., Sekowska, M., Gagnebin, M., Nisbett, J., Deloukas, P., Dermitzakis, E. T., and Antonarakis, S. E. (2009). Common regulatory variation impacts gene expression in a cell type-dependent manner. *Science*, **325**(5945), 1246–1250.

Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis and density estimation. *Journal of the American Statistical Association*, **97**, 611–631.

Fraley, C., Raftery, A. E., Murphy, T. B., and Scrucca, L. (2012). *mclust Version 4 for R: Normal Mixture Modeling for Model-Based Clustering, Classification, and Density Estimation*.

Frazee, A. C., Langmead, B., and Leek, J. T. (2011). ReCount: a multi-experiment resource of analysis-ready RNA-seq gene count datasets. *BMC Bioinformatics*, **12**, 449.

Furman, E. (2007). On the convolution of the negative binomial random variables. *Statistics & Probability Letters*, **77**(2), 169–172.

Gentleman, R. C., Carey, V. J., Bates, D. M., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge, Y., Gentry, J., Hornik, K., Hothorn, T., Huber, W., Iacus, S., Irizarry, R., Leisch, F., Li, C., Maechler, M., Rossini, A. J., Sawitzki, G., Smith, C., Smyth, G., Tierney, L., Yang, J. Y., and Zhang, J. (2004). Bioconductor: open software development for computational biology and bioinformatics. *Genome biology*, **5**(10), R80+.

Halbritter, F., Vaidya, H. J., and Tomlinson, S. R. (2011). GeneProf: analysis of high-throughput sequencing experiments. *Nature Methods*, **9**(1), 7–8.

Hansen, K. D., Wu, Z., Irizarry, R. A., and Leek, J. T. (2011). Sequencing technology does not eliminate biological variability. *Nature Biotechnology*, **29**(7), 572–573.

Hardcastle, T. J. and Kelly, K. A. (2010). baySeq: empirical Bayesian methods for identifying differential expression in sequence count data. *BMC Bioinformatics*, **11**, 422.

Hartigan, J. A. and Wong, M. A. (1979). Algorithm AS 136: A K-Means Clustering Algorithm. *Applied Statistics*, **28**(1), 100–108.

Hillier, L. W., Reinke, V., Green, P., Hirst, M., Marra, M. A., and Waterston, R. H. (2009). Massively parallel sequencing of the polyadenylated transcriptome of C. elegans. *Genome Research*, **19**(4), 657–666.

Hochreiter, S., Clevert, D.-A., and Obermayer, K. (2006). A new summarization method for Affymetrix probe level data. *Bioinformatics*, **22**(8), 943–949.

Huang, D. W., Sherman, B. T., and Lempicki, R. A. (2009a). Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucleic Acids Research*, **37**(1), 1–13.

Huang, D. W. a. . W., Sherman, B. T., and Lempicki, R. A. (2009b). Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nature protocols*, **4**(1), 44–57.

Hunter, S., Jones, P., Mitchell, A., Apweiler, R., Attwood, T. K., Bateman, A., Bernard, T., Binns, D., Bork, P., Burge, S., de Castro, E., Coggill, P., Corbett, M., Das, U., Daugherty, L., Duquenne, L., Finn, R. D., Fraser, M., Gough, J., Haft, D., Hulo, N., Kahn, D., Kelly, E., Letunic, I., Lonsdale, D., Lopez, R., Madera, M., Maslen, J., McAnulla, C., McDowall, J., McMenamin, C., Mi, H., Mutowo-Muellenet, P., Mulder, N., Natale, D., Orengo, C., Pesseat, S., Punta, M., Quinn, A. F., Rivoire, C., Sangrador-Vegas, A., Selengut, J. D., Sigrist, C. J. A., Scheremetjew, M., Tate, J., Thimmajanarthanan, M., Thomas, P. D., Wu, C. H., Yeats, C., and Yong, S.-Y. (2012). InterPro in 2011: new developments in the family and domain prediction database. *Nucleic Acids Research*, **40**(D1), D306–D312.

Klambauer, G., Schwarzbauer, K., Mayr, A., Clevert, D.-A., Mitterecker, A., Bodenhofer, U., and Hochreiter, S. (2012). cn.MOPS: mixture of Poissons for discovering copy number variations in next-generation sequencing data with a low false discovery rate. *Nucleic Acids Research*, **40**(9), e69.

Leek, J. T. and Storey, J. D. (2007). Capturing Heterogeneity in Gene Expression Studies by Surrogate Variable Analysis. *PLoS Genetics*, **3**(9), e161.

Li, J. and Tibshirani, R. (2011). Finding consistent patterns: A nonparametric approach for identifying differential expression in RNA-seq data. *Statistical Methods in Medical Research*, **x**.

Li, J., Witten, D. M., Johnstone, I. M., and Tibshirani, R. (2012). Normalization, testing, and false discovery rate estimation for RNA-sequencing data. *Biostatistics*, **13**(3), 523–538.

Li, P., Ponnala, L., Gandotra, N., Wang, L., Si, Y., Tausta, S. L., Kebrom, T. H., Provart, N., Patel, R., Myers, C. R., Reidel, E. J., Turgeon, R., Liu, P., Sun, Q., Nelson, T., and Brutnell, T. P. (2010). The developmental dynamics of the maize leaf transcriptome. *Nature Genetics*, **42**(12), 1060–1067.

Lund, S. P., Nettleton, D., McCarthy, D. J., and Smyth, G. K. (2012). Detecting Differential Expression in RNA-sequence Data Using Quasi-likelihood with Shrunken Dispersion Estimates. *Statistical applications in genetics and molecular biology*, **11**(5).

Marioni, J. C., Mason, C. E., Mane, S. M., Stephens, M., and Gilad, Y. (2008). RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Research*, **18**(9), 1509–1517.

McCarthy, D. J., Chen, Y., and Smyth, G. K. (2012). Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic Acids Research*, **40**(10), 4288–4297.

Montgomery, S. B., Sammeth, M., Gutierrez-Arcelus, M., Lach, R. P., Ingle, C., Nisbett, J., Guigo, R., and Dermitzakis, E. T. (2010). Transcriptome genetics using second generation sequencing in a caucasian population. *Nature*, **464**(7289), 773–777.

Nagalakshmi, U., Wang, Z., Waern, K., Shou, C., Raha, D., Gerstein, M., and Snyder, M. (2008). The transcriptional landscape of the yeast genome defined by RNA sequencing. *Science*, **320**(5881), 1344–1349.

Nelder, J. A. and Wedderburn, R. W. M. (1972). Generalized Linear Models. *Journal of the Royal Statistical Society. Series A (General)*, **135**(3), 370–384.

Pickrell, J. K., Marioni, J. C., Pai, A. A., Degner, J. F., Engelhardt, B. E., Nkadori, E., Veyrieras, J.-B., Stephens, M., Gilad, Y., and Pritchard, J. K. (2010). Understanding mechanisms underlying human gene expression variation with rna sequencing. *Nature*, **464**(7289), 768–772.

Piegorsch, W. W. (1990). Maximum likelihood estimation for the negative binomial dispersion parameter. *Biometrics*, **46**(3), 863–867.

Pinto, D., Marshall, C., Feuk, L., and Scherer, S. W. (2007). Copy-number variation in control population cohorts. *Human Molecular Genetics*, **16**(R2), R168–R173.

Redon, R., Ishikawa, S., Fitch, K. R., Feuk, L., Perry, G. H., Andrews, T. D., Fiegler, H., Shapero, M. H., Carson, A. R., Chen, W., Cho, E. K., Dallaire, S., Freeman, J. L., GonzÃąlez, J. R., GratacÃšs, M., Huang, J., Kalaitzopoulos, D., Komura, D., MacDonald, J. R., Marshall, C. R., Mei, R., Montgomery, L., Nishimura, K., Okamura, K., Shen, F., Somerville, M. J., Tchinda, J., Valsesia, A., Woodwark, C., Yang, F., Zhang, J., Zerjal, T., Zhang, J., Armengol, L., Conrad, D. F., Estivill, X., Tyler-Smith, C., Carter, N. P., Aburatani, H., Lee, C., Jones, K. W., Scherer, S. W., and Hurles, M. E. (2006). Global variation in copy number in the human genome. *Nature*, **444**(7118), 444–454.

Robinson, M. D. and Smyth, G. K. (2008). Small-sample estimation of negative binomial dispersion, with applications to SAGE data. *Biostatistics*, **9**(2), 321–332.

Robinson, M. D., McCarthy, D. J., and Smyth, G. K. (2010). edgeR: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, **26**(1), 139–140.

Schadt, E. E., Molony, C., Chudin, E., Hao, K., Yang, X., Lum, P. Y., Kasarskis, A., Zhang, B., Wang, S., Suver, C., Zhu, J., Millstein, J., Sieberts, S., Lamb, J., GuhaThakurta, D., Derry, J., Storey, J. D., Avila-Campillo, I., Kruger, M. J., Johnson, J. M., Rohl, C. A., van Nas, A., Mehrabian, M., Drake, T. A., Lusis, A. J., Smith, R. C., Guengerich, F. P., Strom, S. C., Schuetz, E., Rushmore, T. H., and Ulrich, R. (2008). Mapping the genetic architecture of gene expression in human liver. *PLoS Biol*, **6**(5), e107.

Schlattl, A., Anders, S., Waszak, S. M., Huber, W., and Korbel, J. O. (2011). Relating CNVs to transcriptome data at fine resolution: assessment of the effect of variant size, type, and overlap with functional regions. *Genome Research*, **21**(12), 2004–2013.

Sekhon, R. S., Lin, H., Childs, K. L., Hansey, C. N., Buell, C. R., de Leon, N., and Kaeppler, S. M. (2011). Genome-wide atlas of transcription during maize development. *The Plant Journal*, **66**(4), 553–563.

Sing, T., Sander, O., Beerenwinkel, N., and Lengauer, T. (2005). ROCR: visualizing classifier performance in R. *Bioinformatics*, **21**(20), 3940–3941.

Talloen, W., Clevert, D.-A., Hochreiter, S., Amaratunga, D., Bijnens, L., Kass, S., and Göhlmann, H. (2007). I/NI-calls for the exclusion of non-informative genes: a highly effective filtering tool for microarray data. *Bioinformatics*, **23**(21), 2897–2902.

Talloen, W., Hochreiter, S., Bijnens, L., Kasim, A., Shkedy, Z., and Amaratunga, D. (2010). Filtering data from high-throughput experiments based on measurement reliability. *Proc. Natl. Acad. Sci. U.S.A.*, **107**(46), 173–174.

Tarazona, S., García-Alcalde, F., Dopazo, J., Ferrer, A., and Conesa, A. (2011). Differential expression in RNA-seq: A matter of depth. *Genome Research*, **21**, 2213–2223.

The ENCODE Project Consortium (2012). An integrated encyclopedia of DNA elements in the human genome. *Nature*, **489**(7414), 57–74.

Veyrieras, J.-B., Kudaravalli, S., Kim, S. Y., Dermitzakis, E. T., Gilad, Y., Stephens, M., and Pritchard, J. K. (2008). High-Resolution Mapping of Expression-QTLs Yields Insight into Human Gene Regulation. *PLoS Genet*, **4**(10), e1000214.

Wang, L., Feng, Z., Wang, X., Wang, X., and Zhang, X. (2010). DEGseq: an R package for identifying differentially expressed genes from RNA-seq data. *Bioinformatics*, **26**(1), 136–138.

Wu, C., Orozco, C., Boyer, J., Leglise, M., Goodale, J., Batalov, S., Hodge, C. L., Haase, J., Janes, J., Huss, J. W., and Su, A. I. (2009). BioGPS: an extensible and customizable portal for querying and organizing gene annotation resources. *Genome Biology*, **10**(11), R130.

Wu, H., Wang, C., and Wu, Z. (2013). A new shrinkage estimator for dispersion improves differential expression detection in rna-seq data. *Biostatistics*, **14**(2), 232–243.

Wu, T. D. and Nacu, S. (2010). Fast and SNP-tolerant detection of complex variants and splicing in short reads. *Bioinformatics*, **26**(7), 873–881.

Yakowitz, S. J. and Spragins, J. D. (1968). On the identifiability of finite mixtures. *The Annals of Mathematical Statistics*, **39**, 209–214.

Young, M. D., Wakefield, M. J., Smyth, G. K., and Oshlack, A. (2010). Gene ontology analysis for RNA-seq: accounting for selection bias. *Genome Biology*, **11**, R14.

Zeller, T., Wild, P., Szymczak, S., Rotival, M., Schillert, A., Castagne, R., Maouche, S., Germain, M., Lackner, K., Rossmann, H., Eleftheriadis, M., Sinning, C. R., Schnabel, R. B., Lubos, E., Mennerich, D., Rust, W., Perret, C., Proust, C., Nicaud, V., Loscalzo, J., Hübner, N., Tregouet, D., Münzel, T., Ziegler, A., Tiret, L., Blankenberg, S., and Cambien, F. (2010). Genetics and beyond–the transcriptome of human monocytes and disease susceptibility. *PloS One*, **5**(5), e10693.