In high throughput sequencing all normalizations are wrong but some are often useful

true

Scale in high throughput sequencing

High throughput sequencing generates data where the total number of reads is not informative about the environment. These data are compositional, that is, the data is constrained to a fixed upper limit and can be represented by their corresponding proportions (or probabilities) without losing any information. These data must be normalized and many of the normalizations in widespread use can be represented as ratios, further the normalized data are almost universally log-transformed prior to analysis. Thus, all of the aforementioned normalizations are actually log-ratios of some sort. There has been much debate in the literature, and much misunderstanding by the user base as to which normalization is the best because until now the actual assumptions of those normalizations could not be understood inside a common framework.

Recently, Silverman and colleagues proved that log-ratios could be understood as implicit assumptions about the actual scale (size) of the environment. Informally, they recognized that an underlying environment W containing D parts in N samples could be represented by a count matrix decomposed into proportional (p) and total (t) sub-matrices for each sample n

$$W_{dn} = W^p_{dn} \cdot W^t_n \tag{1}$$

or the logarithmic form

$$\log W_{dn} = \log W^p_{dn} + \log W^t_n, \tag{2}$$

where

$$W^t_n = \sum W_{1\ldots D,n} \tag{3}$$

and

$$W^p_{dn} = \frac{W_{dn}}{W^t_n}. \tag{4}$$

The logarithmic form in equation 2 is strikingly similar to the logarithmic formula for the centred log-ratio (CLR) transformation used in compositional data analysis (CoDa). Here the observed data matrix Y can be transformed by the CLR [1] by a taking the log-ratio of each part with the sample geometric mean which we will denote as h:

$$CLR(Y_{dn}) = \log Y_{dn} + h_n, \tag{5}$$

where

$$h_n = -\frac{1}{D} \sum log Y_{1\ldots D,n}. \tag{6}$$

[1] The formula for the CLR gives the same answer if the matrix Y is the raw count values or if its values have been converted to proportions, or indeed by any other linear transformation. This is the scale-invariance property of CoDa in action.

In other words, if Y is a sample of W, then h_n is an implicit estimate of the scale $\log W_n^t$. Of course, just because the equations are similar does not guarantee that the values estimated are correct. They went on to show that in the rare instance when the geometric mean was a good estimate of the scale then analyses were trustworthy, but more commonly that the geometric mean was a biased scale estimator of of the data. Nixon et al made two further important conclusions: first, that any estimate that used the logarithm of a ratio approach was also estimating the scale; and second, that the only the relative scale was important. We will discuss the implications of these conclusions in a later section.

All CoDa based approaches are constrained to real vectors greater than 0

$$X = [x_1, x_2 \ldots, x_D], x \in \mathbb{R} | x > 0, \sum_X = 1. \tag{7}$$

Despite this limitation some advantages of the CoDa framework are that it has true distance metrics and it has a solid geometric interpretation. The usual ways of dealing with 0 values in CoDa approaches are to either impute 0 values, to add a prior to all values, or to restrict the analysis to the parts that include only non-0 values. All these methods restrict the generality of CoDa methods and are rightly pointed out as being weaknesses of the approaches.

Information theory is another framework in which proportional vectors can be examined. Information theory developed as a way to examine the amount of information needed to encode, transmit and decode information. Thus the concept of size was build in from the beginning and this has been one of the most useful advances in all of computation and statistics. The domain of information theory is subtly different from CoDa in that 0 values are permitted:

$$X = [x_1, x_2 \ldots, x_D], x \in \mathbb{R} | x \geq 0, \sum_X = 1. \tag{8}$$

In information theory, the elemental amount of information or surprisal for x_i is the inverse of the logarithm of the elemental probability (1). This measure is often called self-information or the surprisal and we will denote this as I:

$$I(x_i) = \log_2(\frac{1}{x_i}) = -\log_2 x_i. \tag{9}$$

Logarithms of probabilities are additive so that for multiple independent probabilities the entropy H which is the expected information content of the random variable x is

$$H = \sum_X x_i I(x_i) = -\sum_X x_i \log_2(x_i). \tag{10}$$

In this formula Shannon defined $x_i \log_2(x_i) = 0, x_i = 0$ and this effectively removed the 0 constraint when using information theory. Less well known is that a sample average entropy was also defined

$$h = -\frac{1}{D} \sum log_2(x_i), \tag{11}$$

and this is simply the geometric mean. Usefully, h measure converges on H as the number of measurements from a "typical subset" of the data approaches infinity; this is the Asymptotic Equipartition Property of information as described in (2, 3). As shown in Chapter 4 of Mackay (2) the typical subset is approximately Gaussian and so if both h and H are drawn from this distribution they should be identical in the limit. Another way to think about this is that the two measures converge as the distribution of $p(x_i$ approaches the uniform $\frac{1}{N}$. Thus the relationship between H and h demonstrates independently that the log of the geometric mean can be interpreted as a size of a probability vector.

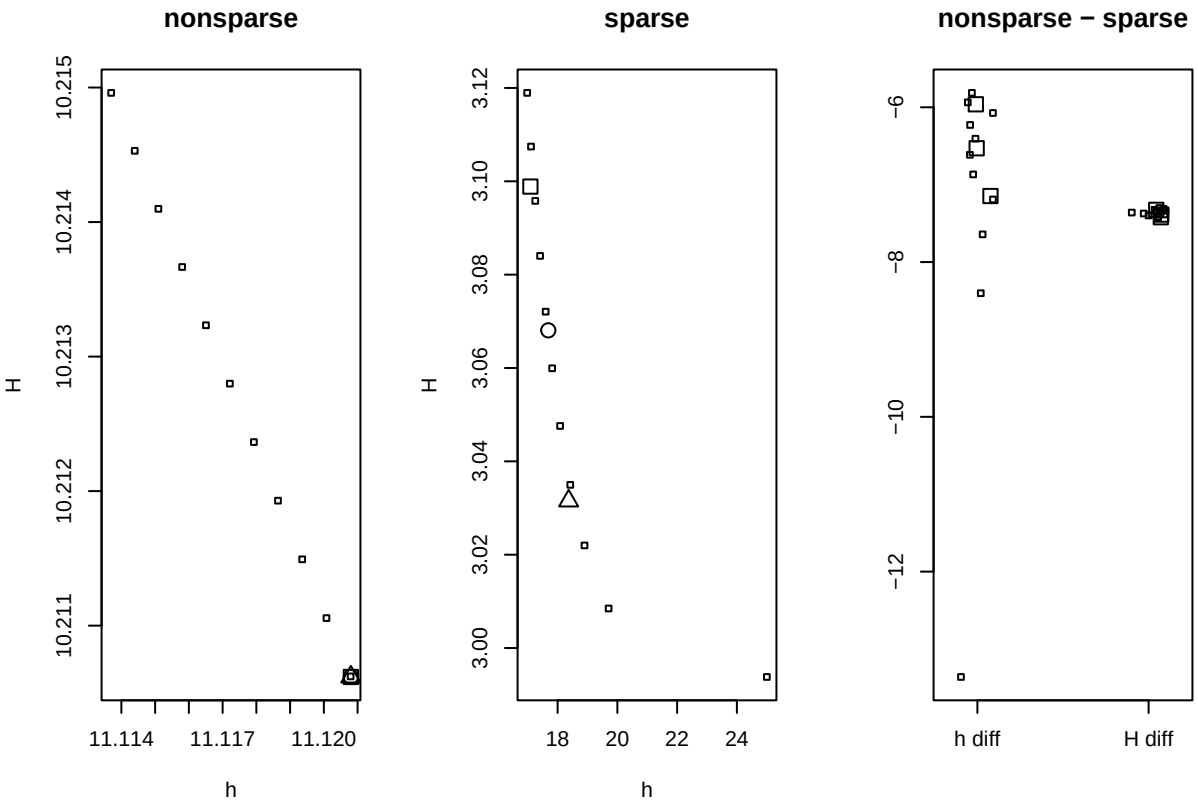
One the challenges of the scale-based approach has been to identify appropriate base scale values for informed scale models. It has been noted before that h can be a good or a poor initial estimate. Recall that the actual scale value is not important, but that the ratio between scale values in each group is important (or the difference in log(scale) values). Figure 1 shows that one potential confounder is that the value of h is very sensitive to small changes in initial assumptions when the data are sparse. For example, one method of treating 0 values is to add a small prior to each value in the matrix. Figure 1 shows how H and h vary for values of this prior in the range of 0.01 to 1. Note that entropy is essentially not affected by these prior values, but that h can produce a very different initial estimate of the scale with different priors. Another method of treating 0s is to replace the 0 values with a pseudocount, this approach produces a result that is similar to adding a relatively large prior in sparse data and similar to adding a small prior in non-sparse data. A final method is to impute 0 values using a CoDa-based approach. This method

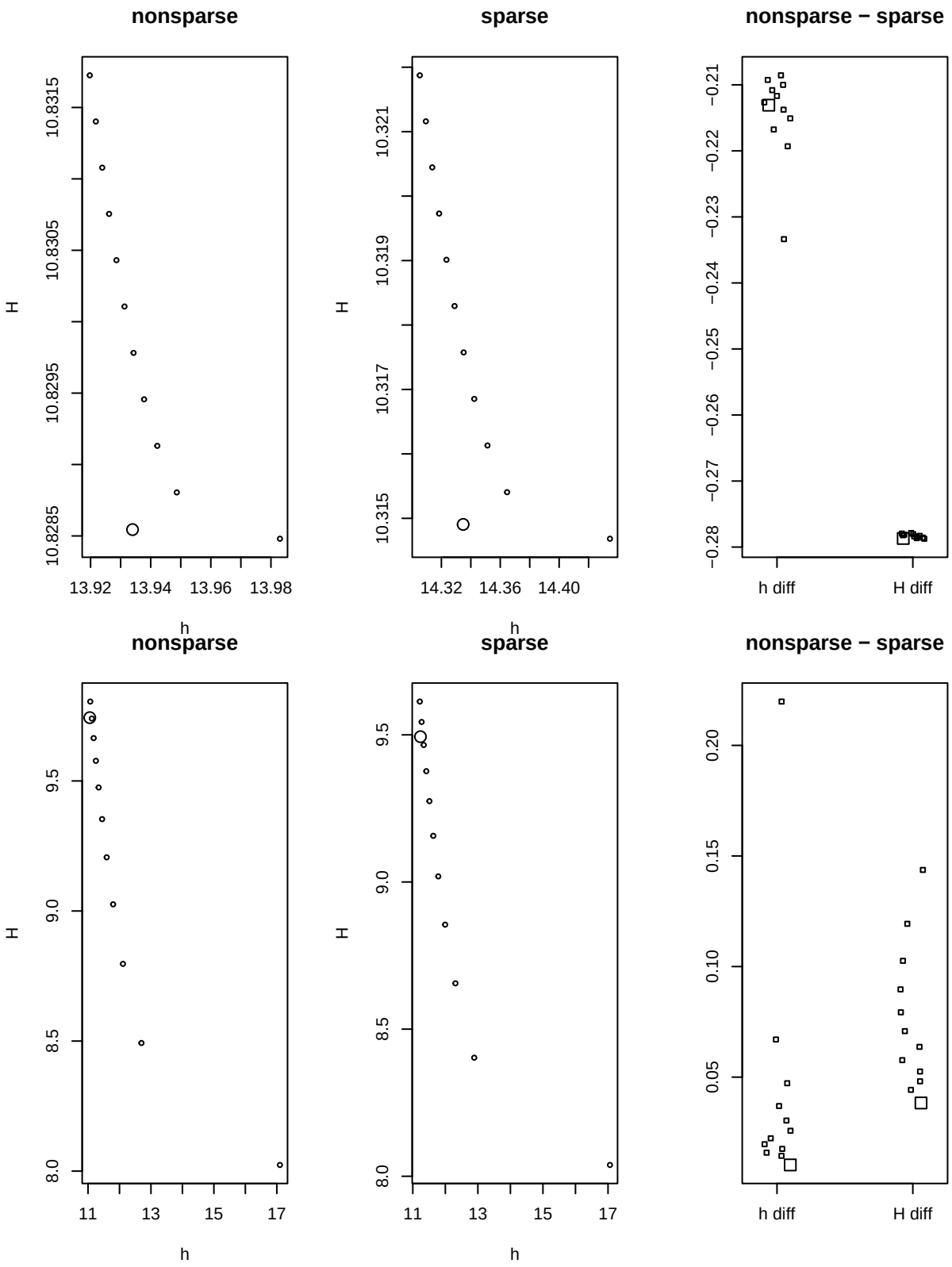
As shown in Figure 1, one reason for this is that the geometric mean cannot be calculated when one or more of the entries in the probability vector is 0. Thus, Furthermore, the tight relationship between can be exploited

```
## Warning in cmultRepl(t(selex), label = 0, method = "GBM"): Row no. 9 containing >80% zeros/unobserve
```

```
## No. adjusted imputations:  1883
```

```
## Warning in cmultRepl(t(selex), label = 0, method = "CZM"): Row no. 9 containing >80% zeros/unobserve
```





Many such normalizations have been developed and are in widespread use. These include simple proportions, relative log expression (RLE), trimmed mean of M values (TMM), cumulative sum scaling (CSS), LVHA, IQLR and others. In the case of proportions the assumption being made is that the total sum of Y_n is the same (or at least directly proportional to) the total sum of W_n. All the other normalizations make the

assumption that a subset of the parts in Y_n are the same (or at least directly proportional to) the same subset of parts in W_n. Thus, while the actual details of how the normalization is conducted vary, the end result is always the same: analyses are conducted with a log of a ratio that can be expressed as a function of the observed data

$$N(Y_{dn}) = \log Y_{dn} + f(\cdot n), \tag{12}$$

where $\cdot n$ represents the whole or a subset of the parts in sample n.

Returning to equation (12) we are now in a position to think about how the concept of scale allows us to understand the assumptions made by different normalizations and to understand how scale normalization supersedes these individual normalizations, including the CLR.

The simplest normalization is the proportion calculated as in equation (4). For proportions the values in the output all sum to 1, and so it is easy to see that the assumption being made is that the scale of all the n samples is the same. This holds if we replace 0 values, add priors or remove 0 count parts. In all cases the total remains 1 and the assumption holds. In many cases the assumption that the scale of all samples is nearly constant is likely good enough. For example, it likely holds approximately in data taken from closely controlled experiments where only a small perturbation was made.

However, early on it was noted that simple proportions introduced bias into the analyses and so alternative normalizations were developed. These normalizations were explicitly developed to attempt to scale the output data so that it more closely reflected the environment. They did this by making assumptions that appropriate reference parts, or subsets of parts, could be chosen that would allow a good estimate of W_d^n from Y_d^n. Multiple such normalizations exist but they differ only in how the subset is chosen. For example the RLE as

There are several ways of determining the appropriate denominator for a composition. The most general is to determine a matrix of all possible pairwise log-ratios; this moves the data onto a real space that is completely interpretable and symmetric (4). However, for large datasets such as those generated by high-throughput sequencing or mass spectroscopy of biological samples where there are thousands of parts, the number of possible ratios becomes computationally infeasible. Thus, a number of other approaches have been proposed that involve different ways of choosing appropriate denominators for the log-ratio. The main problem now being that the choice of denominator can affect the interpretation of the result.

The earliest method used by Aitchison was the ALR (additive log ratio), where a presumed invariant part was chosen as the reference for the denominator. The log-ratio was calculated between each part and the reference resulting in a dataset with 1 less entry than the original. The ALR has the advantage of simplicity and ease of interpretation (5, 6). The primary disadvantage of the ALR is that it is not isometric, that is, the ALR does not by default recapitulate the geometry of the all-vs-all log-ratio. Greenacre (6) showed that it was possible to identify a reference for most datasets that was isometric for all practical purposes. It is noteworthy that the ALR has long been used, without naming it such, in fields such as molecular biology and mass spectrometry where it is common to use a single reference as a standard.

Another approach proposed was the CLR (centred log-ratio), where it is presumed that the geometric mean of the parts in each sample is invariant. The CLR is isometric, and captures all of the ratio information of the all-vs-all approach and has the further advantage of being computationally tractable.

Intermediate denominators have also been proposed. For example, the LVHA (low variance, high abundance) log-ratio

choices of reference give slightly different H (CLR, ALR, hybrid - choice of denominator affects interpretation - Greenacre

Silverman and colleagues introduced concept of scale which still uses a log-ratio. The actual method of choosing the reference is nearly immaterial, instead it is the relationship between the denominators for each group that determines if the log-ratio is useful. The key finding was that the denominator chosen was a proxy

for the underlying size or scale of the environment. The original work by Silverman and colleagues proved this relationship, here we provide an intuitive demonstration based on information theory.

Information theory and CoDa are two ways to deal with compositional data, but information theory is more general. CoDa approaches are applicable to any real vector of numbers

$$X = [x_1, x_2 \ldots, x_D], x \in \mathbb{R} | x > 0, \sum_X = 1. \tag{13}$$

Information Theory approaches are subtly different in that here 0 values are permitted:

$$X = [x_1, x_2 \ldots, x_D], x \in \mathbb{R} | x \geq 0, \sum_X = 1. \tag{14}$$

This s in the l to

Scale is related to entropy - Shannon's entropy measures size of the data - define entropy - define size or scale - H is insensitive to 0 - G $\sim$ H - can use diff H as base

GM is related to Information and Shannon's entropy in HTS datasets

Shannon's entropy has a volume or size

Information content is a fundamental property of all measured systems. Shannon defined the information properties of discrete probability vectors in the field of communications and launched the field of information theory. Information theory concerns itself with how to encode a dataset optimally to minimize its size when transmitted.

The following example is adapted from Chapter 2 of (3). If we have information encoded in bits–i.e., either a "0" or "1"–we can imagine a uniform encoding scheme for four symbols

$$x = [A, C, G, T] \tag{15}$$

where the underlying probability of observing each symbol is

$$Pr_x = [\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}]. \tag{16}$$

Now we can define two encodings; a uniform encoding

$$U_x = [00, 01, 10, 11] \tag{17}$$

and a non-uniform encoding

$$N_x = [0, 10, 110, 111]. \tag{18}$$

In the uniform encoding the number of bits per symbol is always 2 and in the non-uniform encoding it varies from one to three, with the most common symbol encoded by one bit and the rarest symbols encoded by three bits. Lets now examine the amount of information needed to encode a string of characters that has a similar frequency to the true background. The 11 character string is:

$$aacbdabbaab. \tag{19}$$

In the uniform encoding this needs 22 bits because each symbol requires two bits:

$$\sum U_x = 5a + 4b + c + d = 10 + 8 + 2 + 2 = 22. \tag{20}$$

In the non-uniform encoding this needs 19 bits because the number of bits for each symbol is encoded by a variable number of bits:

$$\sum N_x = 5a + 4b + c + d = 5(1) + 4(2) + 1(3) + 1(3) = 19. \tag{21}$$

Going back to our known background frequencies we can calculate the expected number of bits for a given set of symbols and frequencies. For the uniform encoding we get

$$\frac{1}{2}(2), \frac{1}{4}(2), \frac{1}{8}(2), \frac{1}{8}(2) = \frac{16}{8} = \frac{8}{4} = 2 \tag{22}$$

which is 2 bits per symbol. While for the variable length encoding we get

$$p_i = \frac{1}{2}(1), \frac{1}{4}(2), \frac{1}{8}(3), \frac{1}{8}(3) = \frac{14}{8} = \frac{7}{4} \tag{23}$$

which is smaller.

We are now able to understand the formula for entropy. For notational simplicity assume we have a single discrete random variable to represent a probability distribution with d elements; i.e. $X = \mathbf{p}_{i=(1\ldots d)}$. In information theory, the elemental amount of information or surprisal for p_i is the inverse of the logarithm of the elemental probability (1). This measure is often called self-information or the surprisal and I will denote this as I:

$$I(X_i) = \log_2(\frac{1}{X_i}) = -\log_2 X_i. \tag{24}$$

Logarithms of probabilities are additive so that for multiple independent probabilities the expected information content of the random variable x is

$$\sum_X p(X_i)I(X_i) = -\sum_X p(X_i)\log_2(X_i). \tag{25}$$

And this is the formula for entropy H! For any encoding of information in a probability vector Shannon showed that H is the expected information content of that probability vector. We can demonstrate this property if we remember that the base 2 logarithms of the fractions from equation 2 are -1, -2, -3, -3. The expected entropy to base 2 is thus:

$$H(x) = -\Big(\frac{1}{2}(-1) + \frac{1}{4}(-2) + \frac{1}{8}(-3) + \frac{1}{8}(-3)\Big) = \frac{7}{4} \tag{26}$$

which is precisely the same result as in equation 9 where we calculated the average bit length of the infinite length message. Thus, entropy measures the average information encoded by each symbol in a message. One conclusion of the above discussion is that information has a size and that we can measure that size as the average number of bits needed to encode the information.

Less well known is that a sample average entropy was also defined

$$h = -\frac{1}{d}\sum log_2(p), \tag{27}$$

and that this measure converges on H as the number of measurements from a "typical subset" of the data approaches infinity; this is the Asymptotic Equipartition Property of information as described in (2, 3). As shown in Chapter 4 of Mackay (2) the typical subset is approximately Gaussian and so if both h and H are drawn from this distribution they should be identical in the limit. Another way to think about this is that the two measures converge as the distribution of p_i approaches $\frac{1}{n}$.

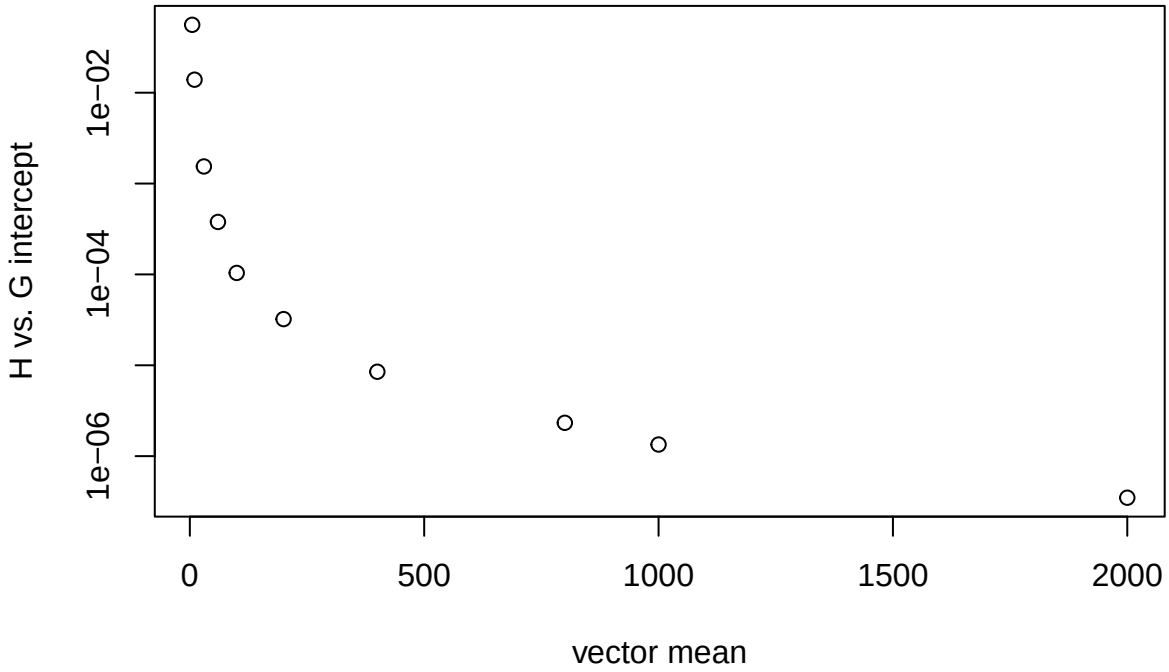


Figure 1: Plot showing that H and h converge as the size of the dataset increases.

Compositional data analysis is concerned with the analysis of compositions; datasets that are strictly positive and that have an arbitrary upper limit. These can always be reduced to a proportion (or probability) without loss of information. Compositional approaches have become popular for the analysis of high throughput sequencing data, and other high dimensional data generated by 'omics platforms. In compositional data analysis (CoDa) one popular data transformation is the centred log-ratio (CLR) transform, which is:

$$clr_x = log_2(x_i) - mean(log_2(x) = log_2(x_i) + h_x, \tag{28}$$

and we can see immediately that the second term is equivalent to $h(x)$. Thus, information theory and compositional data analysis intersect through the geometric mean of a probability vector.

This intersection of information theory and CoDa can help us understand the scale of a system as defined by Nixon and Silverman which can help in the interpretation of many high throughput sequencing datasets. Since H can be defined as a volume for discrete distributions it is reasonable to treat h treated similarly.

Thus, the use of the geometric mean as a measure of size is acceptable both from the information theoretic and from a CoDa perspective.

Biological context of H - average amount of information in each entry. CLR is diff between sample avg entropy and each i(x)

We can think about scale from an information theoretic point of view as a measure of how much information, or total uncertainty, is encoded in a particular sample (7, 8). In the geometric interpretation of information theory used in quantum information theory of information (3, 9), entropy can be interpreted as the volume occupied by a probability distribution relative to the maximum total entropy. See chapters 4 of the PhD thesis of Lecamwasam (10) for a more complete explanation of this. Furthermore, the Asymptotic Equipartition

Property says that the expected value of h(X) for a large number of random variables approaches H(X) as the number of variables approaches infinity (3, 9).

The weighted average entropy of the system $H(X)$ is the weighted sum of the elemental information;

$$H(X) = -\sum_{i=1}^{d} p_i \log_2 p_i \tag{29}$$

$H(X)$ corresponds to the mean amount of information in each entry that is needed to reproduce X. We can also calculate the unweighted expected amount of information for each observation in the random variable, and this is the mean of the elemental probability. This measure is also called the sample average of the information (3);

$$h(X) = -\frac{1}{d}\sum_{i=1}^{d} \log_2 p_i \tag{30}$$

The linkage between compositional analysis, scale inference and information theory comes when we realize that the logarithm of the geometric mean calculated in base(2) is:

$$l2G(X) = \log_2 G(X) = \frac{1}{d}\sum_{i=1}^{d} \log_2 p_i \tag{31}$$

;

We see that $h(X) = -l2G(x)$. Furthermore, $l2G$ is used as the basis for the centred log-ratio transform and is the starting point for scale-based inference:

$$CLR(X) = log2(p_i) - l2G(X) = log2(p_i) + h(X)$$

Thus we see that the geometric mean used in the centred log ratio (CLR), often used for Compositional Data Analysis (CoDa) (5) is related to entropy or H. Indeed, we can rearrange the CLR formula to show that it can be interpreted as computing the difference between the elemental information and the mean information content:

$$CLR(X) = -log2(p_i) - (-l2G(X)) = -(-log2(p_i) - h(X))$$

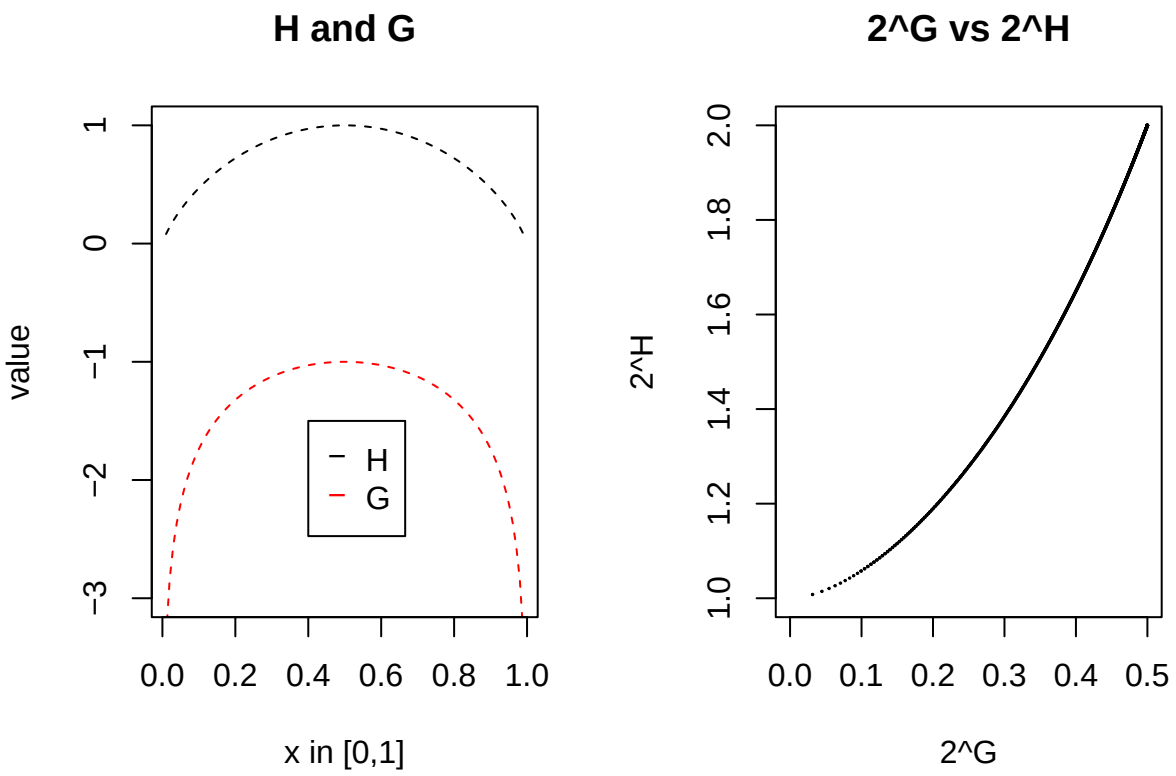
As defined in (11), the scale is the inverse of $l2G$, which is $h(X)$. Thus, one way we can understand scale is that it is measuring the total complexity of the system, and this is expected to increase with absolute size in most cases. Moreover, $H(X)$ and $G(X)$ share similar shapes in the continuum between 0-1 for a bivariate distribution as shown below:

```
par(mfrow=c(1,2))
curve(mf.G, from=1e-2, to =.99,  col='red', lty=2, ylim=c(-3,1),
   xlab="x in [0,1]", ylab="value", main="H and G")
curve(mf.H, from=1e-2, to =.99,  col='black', lty=2, add=T )

legend(.4,-1.5, legend=c('H','G'), col=c('black','red'), pch="-")

vals <- seq(from=.001, to=0.99, by=0.001)

plot(mf.2G(vals),mf.2H(vals), xlim=c(0,0.5), ylim=c(1,2), pch=19, cex=0.1,
  xlab="2^G", ylab="2^H", main="2^G vs 2^H")
```



The difference being that entropy is constructed to have a value of 0 at the margins because Shannon defined $0log(0) = 0$ while the geometric mean approaches negative infinity.

Now let's think about the idea of entropy as a volume which allows us to identify what amount of the available entropy space a given observation fills. The following is taken and modified from (10), to which you should refer for a more fulsome discussion, and it is covered in Chapter 2 of Wilde(3) and Chapter 3 of Cover and Thomas (9). If we start with a four part system $X1 = [A, C, G, T]$ where the frequencies are equally and identically distributed, then $p_A = p_C = p_G = p_T = \frac{1}{4}$. $H(X1) = -1 * 4 * (\frac{1}{4} * log2(\frac{1}{4})) = 2$. This is the maximum entropy possible. We can obtain the "volume" of $X1$ by exponentiating $H1$ using the same base as was used to calculate the entropy; $V1 = 2^{H1} = 4$. This is the same as the number of letters in the system; so the volume needed to explain the system is 4 units (in this case bits). But what happens in another system, $X2$ where A occurs with a much higher probability, say 0.7, and the other three are distributed equiprobably amongst the remainder with a probability of 0.1; i.e. $p_C = p_G = p_T = 0.1$. In this case $H2 = -1 * ((\frac{7}{10} * log2(\frac{7}{10})) + (3 * (\frac{1}{10} * log2(\frac{1}{10})))) = 1.358$. Here the volume of $X2 = 2^{H2} = 2.56$; meaning that less than the maximum volume is taken up by the information. Here $X2$ consumes about 64% of the volume of system $X1$. Thus, the entropic volume is a measure of the total complexity or the scales of the two systems.

But what happens if we consider the geometric mean instead of the entropy? In the example above, $l2G(X1) = (4 * log2(0.25))/4$ = -2, and $l2G(X2) = (log2(\frac{7}{10}) + 3 * log2(\frac{1}{10}))/4 = -2.62$. Exponentiating gives us values of 0.25 and 0.16 suggesting that the $l2G$ measure includes some estimate of size. Comparing the size of H vs $2^(l2G)$ suggests that both measures contain related information. Thus we can understand that scale is related to the information volume of a system, which in turn is related to the size of the system.

Empirically, we can see that $G(\mathbf{Y}_n^{\|})$ is strongly correlated with Shannon's Entropy $H(\mathbf{Y}_n^{\|})$ as expected from the discussion above, and that this difference converges to a constant as the number of entries in the probability vector increases regardless of the distribution, although different distributions converge at different rates. For example, if we plot the relationship between H and G as a function of the length of the probability vector we can see a direct inverse relationship.

```
## (Intercept)
```

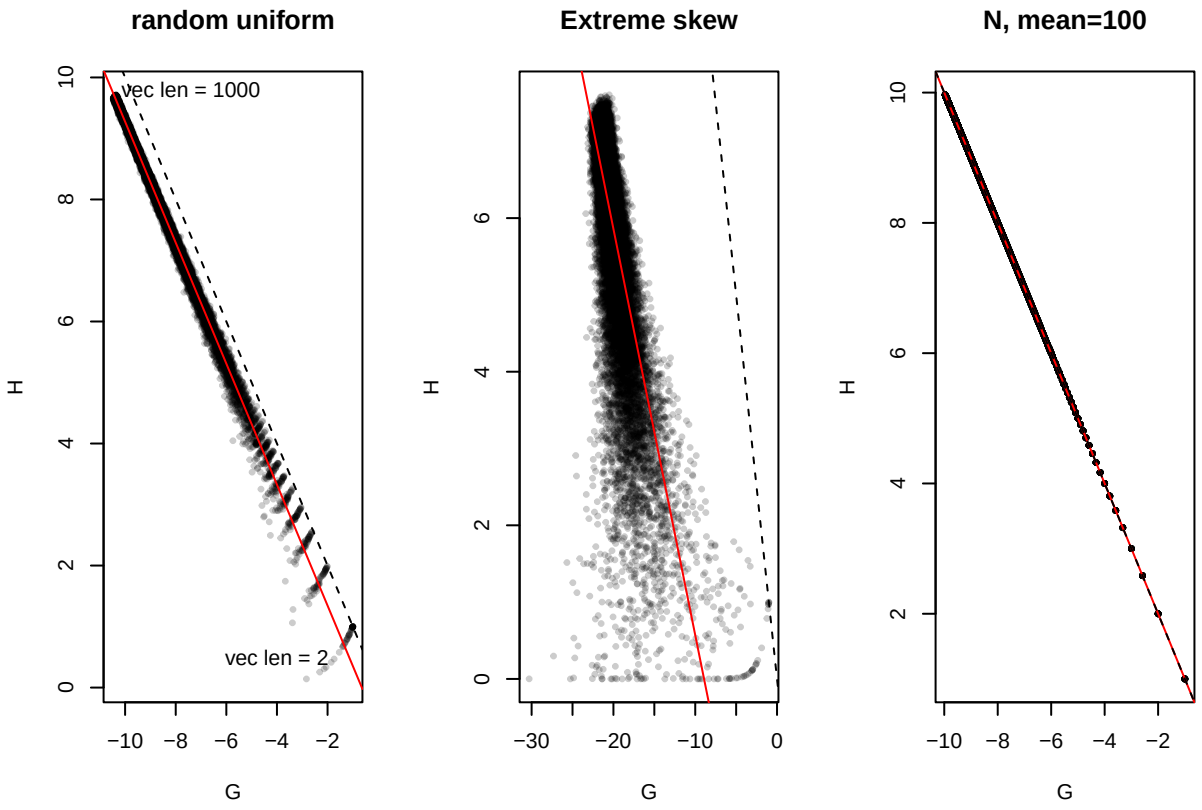


Figure 2: Association between entropy (H) and geometric mean (G) as a function of vector length. Twenty random vectors were constructed for each length between 2 and 1000 in increments of 2 for each of the random distributions in the legend; N = Normal, U = Uniform, B = Beta. The bottom right of each plot represents vectors of length 2, and the top left represents the vector of lenght 500. The maximum value of H increases as the vector length increases, and the maximum value of G decreases in lock-step. Each random distribution has an obviously distinct relationship between the two measures. For the purposes of high throughput sequencing the Beta distribution is most similar to that seen in the majority of instances.

```
##  -0.6384054
```

```
## (Intercept)
##   -4.718218
```

```
##   (Intercept)
## -0.0001096076
```

When we plot the relationship for any individual probability vector, we see that there is an direct relationship between the entropy and the log of the geometric mean, but that this relationship strongly depends on the underlying distribution of the probability distribution X.

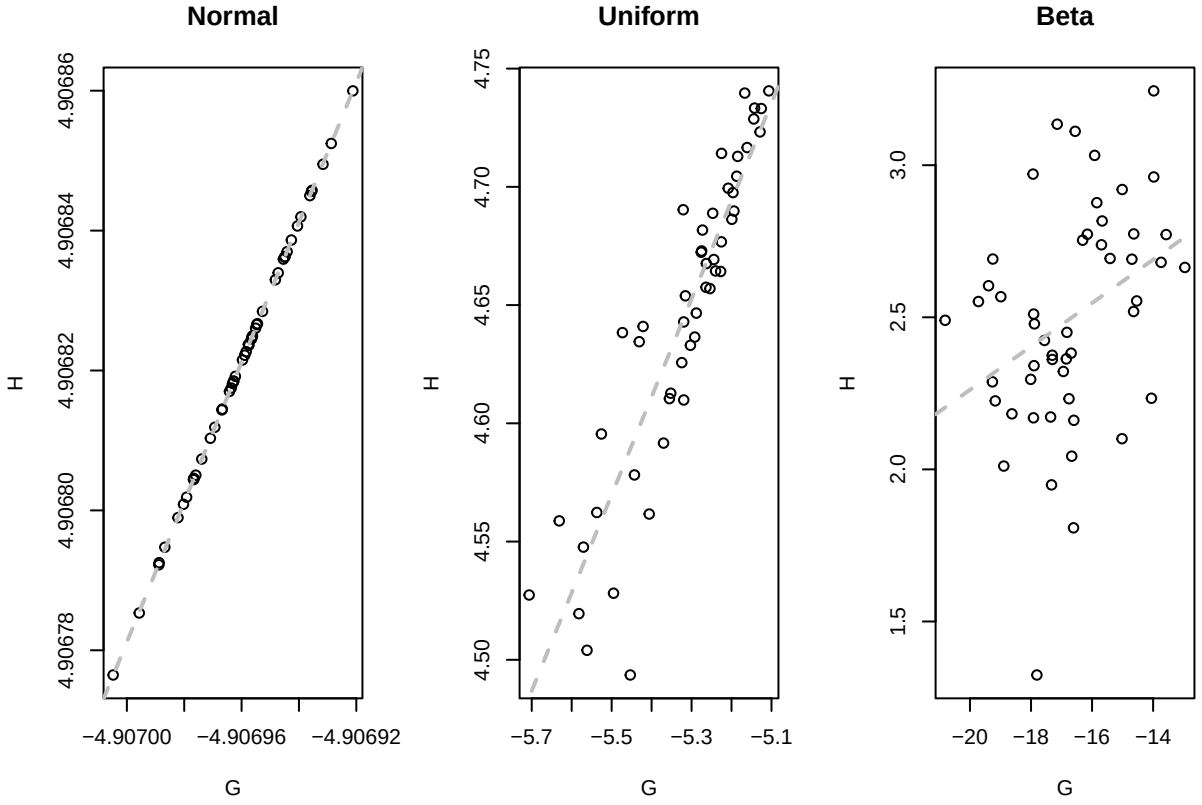


Figure 3: Plot of the association between H and G at a vector length of 30. The relationship between H and G is inverse, and the strength of that association depends on the distribution. The N distribution shows a very strong assocation, while the Beta distribution is less well defined. Associations are shown for a vector length of 30.

In real data, shown in Supplemental Figure 3 and Table 1 the correspondence is not as predictable, likely because the real data is a more complex distribution than any of the idealized distributions. Thus, these two measures have different behaviours with different distributions of p_i. In the case of a uniform distribution both $H(\mathbf{Y}_n^{\|})$ and $G(\mathbf{Y}_n^{\|})$ are maximal when $p(x)$ is equally and identically distributed. Thus, we expect that they are positively correlated here. In a Normal or a skewed distribution, we also observe a positive correlation because both are affected in the same direction by outlier values. In very sparse datasets, the two measures could become uncoupled because $H(\mathbf{Y}_n^{\|})$ could ascribe some uncertainty to the large number of low probability events, while $G(\mathbf{Y}_n^{\|})$ would tend to be very small. Here these two measures could be either uncorrelated or exhibit negative correlation. We can see this distributional behaviour in different datasets.

Intuitively, systems with different scales will contain different amounts of information and so we would expect $W_n^{\perp} \sim H_n$. As the scale of a system as defined by Nixon et al. (11) is inversely related to G, this means that scale is directly proportional to the information content and entropy of the data.

Below I show that we can replace G with H in the calculations performed by ALDEx2 without loss of utility.

Recall the underlying system is described by a $D \times N$ matrix of counts $\mathbf{W}$ decomposed into the proportions for the n^{th} sample $\mathbf{W}_n^{\parallel}$ (or the equivalent probability distribution $\mathbf{p}(w_n)$), and its scale $\mathbf{W}_n^{\perp}$, such that $\mathbf{W} = \mathbf{W}^{\parallel}\mathbf{W}^{\perp}$. Sequencing returns counts which are related to the underlying proportion; i.e., $\mathbf{Y}_n^{\parallel} \sim \mathbf{W}_n^{\parallel}$

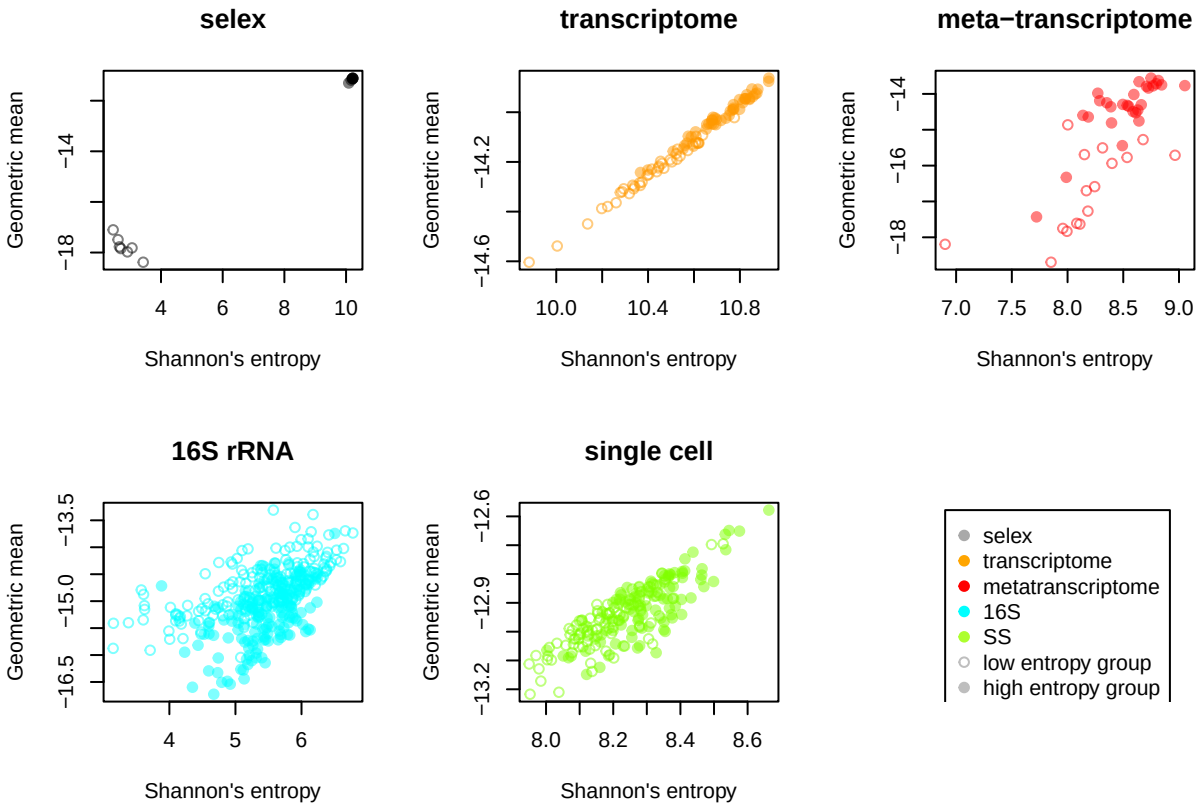


Figure 4: Plot of Shannon's entropy (H) vs geometric mean (G) for each sample in different datasets. The groups that each sample belong to are highlighted as filled or open circles. Each group in each dataset has different entropy with the groups in the selex and metatranscriptome datasets being highly distinct.

The table below summarizes the mean values for, and the correlation between, G and H (cor) and the sparsity defined as the proportion of features with less than 1 count per sample (spar) for each association in each group of samples:

Dataset	group	$\overline{G}$	$\overline{H}$	cor	spar
Selex	control	-11.2	10.2	0.99	0
"	selected	-17.8	2.8	-0.88	0.802
yeast	snf2 ko	-14.0	10.7	0.99	0.004
"	WT	-14.2	10.4	0.99	0.007
Meta	H	-18.8	8.6	0.78	0.451
"	BV	-18.2	8.9	0.79	0.238
16S	Pup	-14.7	5.4	0.68	0.079
"	Cent	-15.2	5.4	0.53	0.251
SS	A	-13.0	8.2	0.83	0.978
"	B	-12.9	8.3	0.80	0.977

While not necessary in all datasets (e.g., the transcriptome example discussed in the previous section and see

the Supplementary material), it can greatly improve modeling in certain cases, especially when the scale is highly asymmetric between conditions. In order to specify a scale model from scratch, we need to revisit the concept that all normalizations in widespread use are actually ratios with the denominator implied by the normalization. Therefore, we can easily deviate from a certain normalization (e.g., the geometric mean assumption implied by the CLR) by specifying a total model based on the mean difference between conditions. While knowing the mean difference between conditions may seem cumbersome in practice, it is the *relationship* between the group scale values that is important, not their raw values (Nixon et al. 2023). This can be illustrated quite simply by starting with the mean ratio in G between groups for the yeast transcriptome dataset which are 6.05×10^{-5} for snf2 and 5.08×10^{-5} for WT; their ratio being 1.17, or a 0.17-fold difference. Using this information, we can recapitulate the differential abundance analysis in Figure 2B and 2C exactly by using setting the mean denominator of group 1 to 1, and group 2 to 1.17 with a gamma of 0.5 as shown in Supplemental Figure 2. This ratio can be adjusted to alter the mean assumption placed on the group scale values.

We can see that for most datasets the difference between the $\overline{G}$ in each group is relatively small. Most significantly, the selex dataset has a very large difference of about 100-fold, and both the 16S and the metatranscriptome dataset have about a 1.5 fold difference. These three datasets are candidates for a full scale model correction.

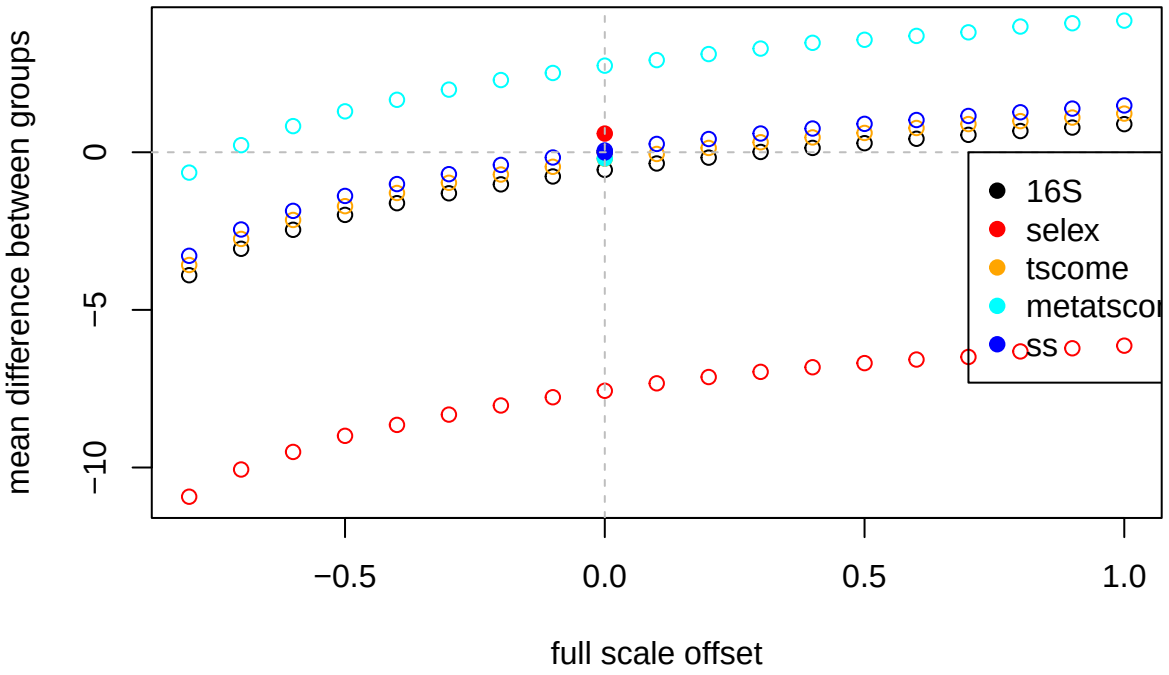


Figure 5: Plot of the offset of the mean differnece between groups as a function of scale ratio. For this, the default scale of 1:1 was altered in increments of 0.1 keeping the gamma parameter (dispersion) at 0.5. The filled circle shows the outcome when the calculation is done using the geometric mean and the same gamma parameter.

Nixon et al. (11) showed that many operations on HTS datasets relied on both the proportion and scale components of the data. Moreover, all normalizations impose a scale model on the data, but the appropriateness of these models has never been explicitly acknowledged or tested. Thus, the full scale model option allows the investigator to set both the offset between the geometric means of the features and their dispersion and observe how this affects the analysis outcome.

In the offset plot above we can see the cause and the effect of the full model with a fixed gamma of 0.5 and a base scale of 1 in each group. We see that the mean location of well centred datasets (yeast transcriptome, single-cell transcriptome) are close to 0, but could be centred better with small changes in scale ranging from

0 (single cell) to 1:1.1 for the yeast transcriptome dataset. In contrast, centring the 16S dataset requires about a 1:1.3 fold change. The metatranscriptome dataset would require about a 0.5:1 change in relative scale between groups, but as shown in the main text, centring the housekeeping genes is more apt.

The in vitro selection dataset is clearly an outlier in both the difference between the average group geometric mean, and in the offset plot. However, this dataset can be used to illustrate the power of the full scale model and the relationship between G_n and scale. In Figure 3 we can see that the default output of ALDEx2 has a centered output. This occurs largely by chance, as the high sparsity of the selected (S) group is balanced almost exactly by the arbitrarily chosen sequencing depth so the non-selected group (NS) appears to have a similar location as the S group. The difference in entropy between the two groups and the differences in geometric mean are very large, with the difference in $\log_2 \overline{G}(S)$ and $log_2\overline{G}(NS)$ being about $2^{6.6}$. Setting the scale of both the S and NS groups to 1 we find that the difference in location is approximately $2^{7.7}$ in close agreement with the difference the geometric means. For this dataset to be centered we need to have a scale ratio $\approx$ 1:50 or more. Note that the scale ratio is inverse to the ratio of geometric means as described above. In fact, in this dataset the relative abundances of the majority of features are nearly invariant, but this is masked by the large absolute changes in a small number of features (12), thus changing the scale of the data. Neither DESeq nor edgeR are able to provide a reasonable analysis of this dataset because the normalizations used assume equivalent scales (13).

Figure 3 shows an effect plot of various scale models with this dataset. The full scale model, where the strong assumption that the mean G is assumed to be 1:1 between the two groups, dramatically skews the output and the large number of relatively invariant features are now identified as significantly different. While not wrong as long as the assumption that the scales are identical is stated, this is not a useful analysis outcome. Modifying the mean scale difference between the NS:S groups to be $\approx$ 1:50 different moves the centre of the large number of relatively invariant features to the centreline of no difference, and recapitulates the default result obtained using G_n as the scale estimate where the ratio is $\sim 100:1$. Note that we get exactly the same answer (within random sampling error) with a scale of .02 for group NS and a scale of 1 for group S, or using a scale of 1 for group NS and a scale of 50 for group S. This shows that it is the relative difference between scales that is important in this dataset, not the absolute values. From this result we can conclude that, on average, the difference in underlying scale in the system is about $\approx$ 50-fold, and this is congruent with the circa 100-fold difference in $\overline{G}$ between groups; the discrepancy being explained because the default scale model is applied uniformly to all samples, whereas the different values of the within-group geometric means ranging over a $\{ > 2.5)$ fold range. Thus, an advantage of a full scale model is that we have gained both information and understanding about the drivers of asymmetry underlying system.

How scaling affects dispersion

Issues with DESeq2 and edgeR

DESeq2 and edgeR are two of the most commonly used tools for differential abundance analysis of bule RNA sequencing datasets. They both operate by finding a scaling factor that makes all the samples commesurate. DESeq2 does this by finding a midpoint feature that can be used as a reference in each sample; this can be different for different samples. The edgeR reference finds the midpoint of the 'typical' sample instead. In both cases the data are then scaled by dividing by a small factor that makes the read counts commesurate. Differential abundance analysis is then performed on the scaled values after taking their logarithm to base 2. In some ways this is similar to the log-ratio approach used by ALDEx2, but is more prone to dataset and sample effects than is the log-ratio method (14)

Examining the plots, edge R clearly not centred with the median housekeeping functions offset by 0.0515571 and minimum FDR value is 0.3382316. Additionally, as shown in Figure 4there is little range in the p-values relative to those seen for DESeq2, and the values are more in line with the range seen with ALDEx2.

DESeq2 is better centred with median housekeeping functions offset by -0.6359311 but the minimum FDR value is $6.9497481 \times 10^{-15}$. In addition there are a large number of functions with 0 variance, these are very low count functions with very high sparsity in the dataset. These are not differential in the DESeq2 analysis.

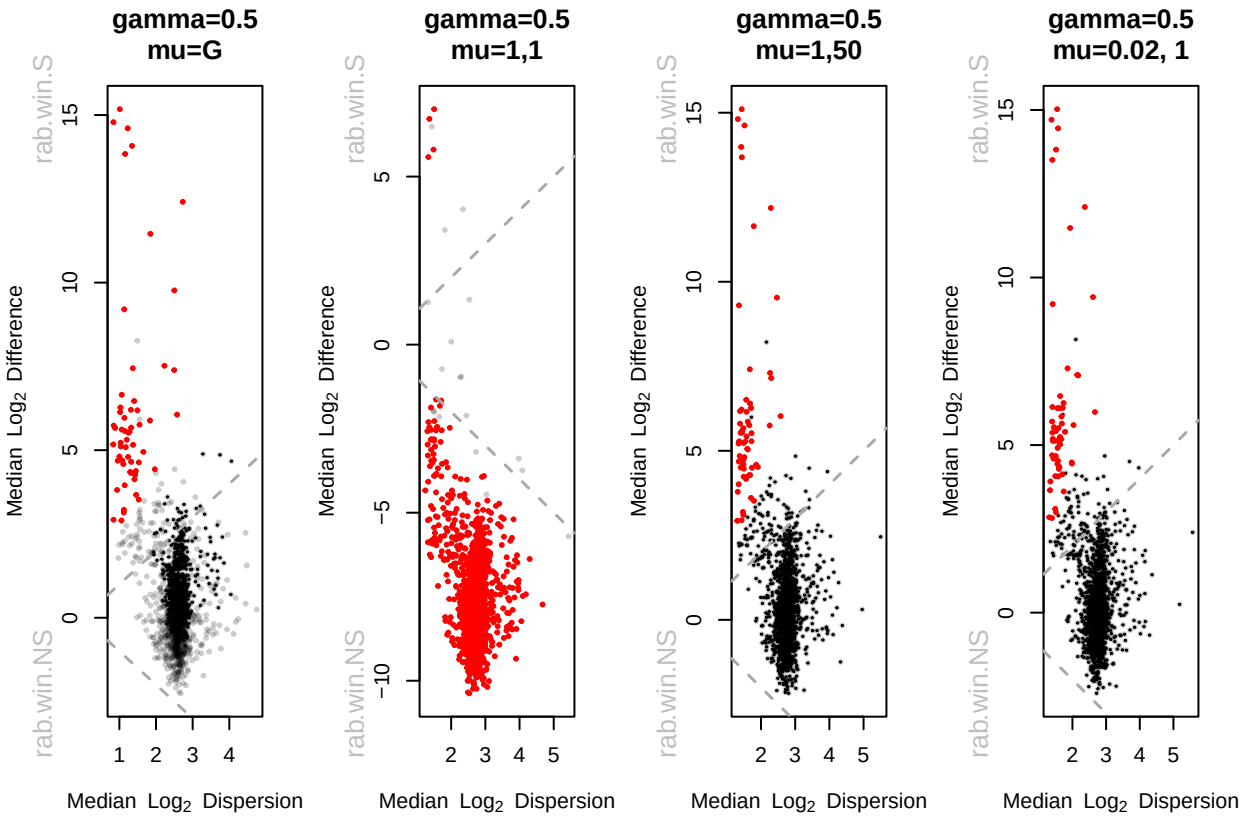


Figure 6: Effect plots of the selex dataset with various gamma and scale parameters. All scales are calculated with a logNormal distribution to ensure symmetry for the user.

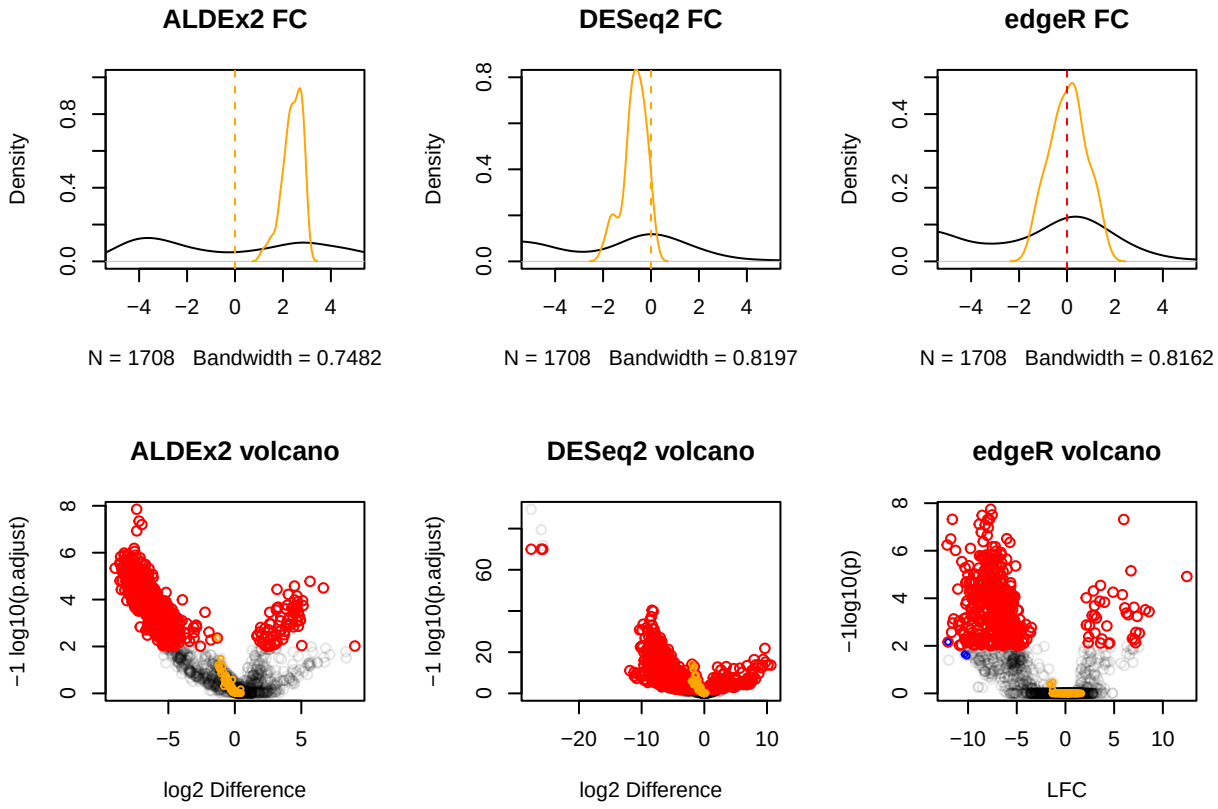


Figure 7: Shown here are the mean log2 fold change as a density plot, and a Volcano plot showing the location and adjusted p-value for each feature in the metatranscriptomic dataset. The DESeq2 approach does a good job of centring this data, while edgeR is less suitable. The volcano plots show dramatically different outcomes. The DESeq2 algorithm assigns very large fold changes to features that have only moderate change, and further identifies a very large proportion of features as significantly different. In contrast, edgeR exhibits a much smaller number of differentially abundant features. In both volcano plots, the housekeeping genes in the main Figure 3 are shown in orange. We can see that these are asymmetrically distributed in both plots. Additionally the location of the features that DESeq2 identified as having a very large difference are shown in the edgeR volcano plot as blue circles.

However, there are a number of functions in the DESeq2 analysis that are very differentially abundant, with a log2 fold change of < -20. Examination of the raw counts shows that these are uniformly 0 or 1 in the H dataset, and present at high counts in some, but not all of the BV dataset. That these stand out is odd, since inspection of the count table shows that there are many other functions that are missing from the H dataset, and have higher and more uniform counts in the BV dataset. Thus, the importance of the outlier functions seen in the DESeq2 volcano plot should be viewed with suspicion and may be an artefact of the normalization used. When overplotted on the edgeR volcano plot (blue), it is clear that these functions have non-signficant p-values.

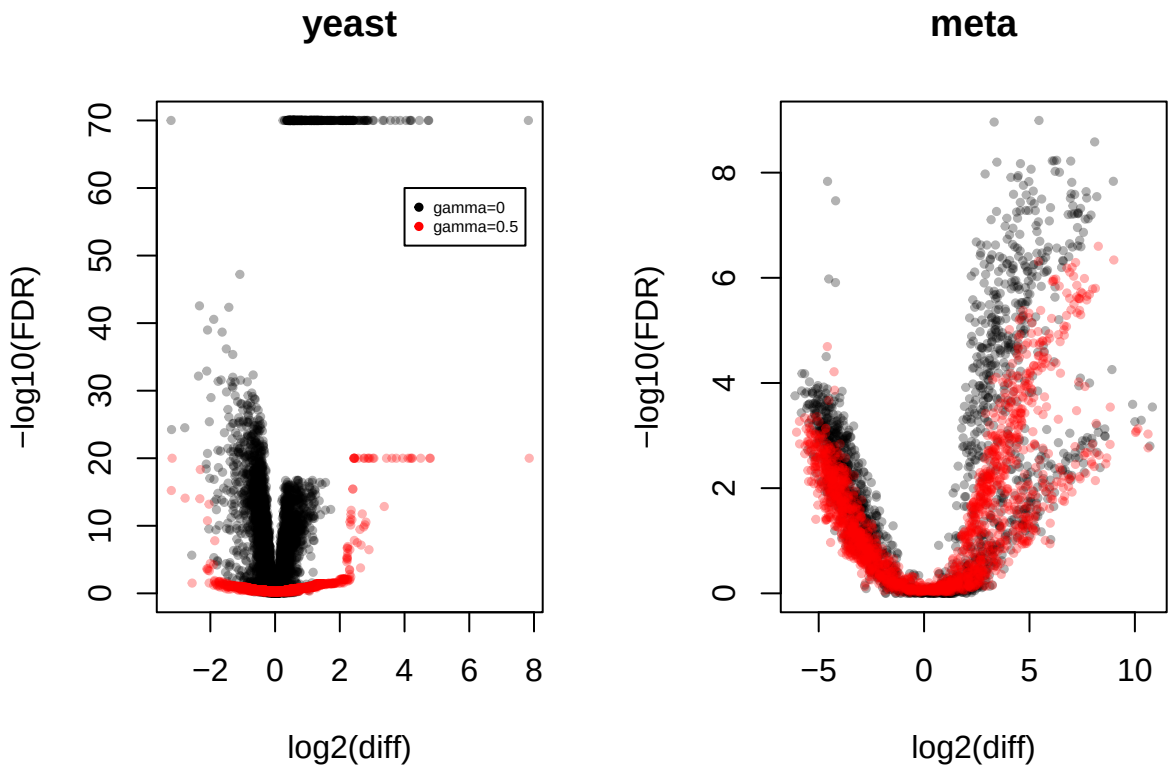


Figure 8: Shown here are two volcano plots that plot the mean log2 fold change plotted vs the log of the FDR for the yeast transcriptome dataset and for the metatranscriptome dataset. Data were generated in ALDEx2 with $\gamma = [0, 0.5]$ in black or in red.

Adding a small amount of scale, $\gamma = 0.5$, has a major effect on the volcano plot for the yeast transcriptome dataset, but mimimal effect on the vaginal metatranscriptome dataset. In both datasets the FDR values are raised, but no effect is observed on the difference between values. However, the concordance between the FDR values and the difference between become very tight for the yeast transcriptome dataset, but the concordance is not visibly unchanged for the other dataset. This is likely because both the difference between and the dispersion were both very small in the former and substantially larger in the latter.

Checking the scale assumptions of the RLE and TMM normalizations

We can show that the normalizations built into the edgeR Bioconductor package (RLE, TMM, TMMwsp, upperquantile) are scale assumptions by using the normalization factor as an input to `aldex.makeScaleMatrix()` and then measure the mean location of the data as above. The ideal behavior of a normalization is that the mean location of the data should be close to 0 or unchanged. This behavior will ensure that Type 1 and Type 2 errors due to scale assumptions are minimized.

The results, shown in the tables below compare these normalizations to the no scale assumption (iso) and the geometric mean normalization (GM) assuming that the normalizations are either on a log scale or a linear scale. That these affect scale can be observed by their effect on the mean location of the data. For the most part assuming no scale differences between groups centres the date less well than does the geometric mean. In most cases the other normalizations perform poorer than assuming no scale at all for the rRNA dataset and about as well as the no scale assumption in the metatranscriptome datasets and the single-cell transcriptome dataset. All normalizations perform poorly in the selex dataset. Overall, these results may not be surprising because the TMM and RLE (and related normalizations) were developed specifically to scale and normalize transcriptome datasets (15, 16), but have become widely used in the analysis of other data modalities; e.g. (17).

Table 2: Log scale models

	RLE	TMM	TMMwsp	upperquartile	iso	GM
rRNA	-2.1529252	-0.1763102	-0.3699510	0.0865392	-0.5599236	0.0093805
selex	-1.9343851	2.1047060	1.9617191	NaN	-7.5096353	-0.0062949
yst	0.0746243	0.0297339	0.0639645	-0.0694199	-0.2160747	-0.0129932
meta	2.7626908	2.5125909	2.8658800	2.5469506	2.7145893	-0.0381394
ss	0.1384218	0.0843315	0.0010290	0.1008664	0.0578294	-0.0276270

Table 3: Linear scale models

	RLE	TMM	TMMwsp	upperquartile	iso	GM
rRNA	-2.2112427	0.0072079	-0.1431558	0.2898137	-0.5577481	0.0093805
selex	-1.6912748	0.5256675	0.5113871	NaN	-7.5296170	-0.0062949
yst	0.1448714	0.1524261	0.1634843	0.0022118	-0.2288918	-0.0129932
meta	3.2141857	2.6937636	3.1516101	2.5240846	2.7581217	-0.0381394
ss	0.1735196	0.0961419	-0.0140678	0.1074003	0.0524419	-0.0276270

Thus far we have observed that adding scale uncertainty has a negligible effect on either the differences between groups or the relative abundance measure with a correlation of 0.999 and 0.998. The effect is entirely restricted to the dispersion estimats as shown below. Panel A shows the change in dispersion relative to the rAbundance of the features. Here we can see that the minimum dispersion is increased as gamma increases. The horizontal lines show the median value for the features with a rAbundance between -0.5 and 0.5. Panel B shows that this increase is non-linear, being more pronounced among those features that had minimal dispersion when gamma=0. Panel C shows that increasing gamma rotates the dispersion estimates at high relative abundances, such that housekeeping genes no longer have larger mean dispersions (rAB > 5) than do regulated genes (rAB >0, < 5). Overplotted are the lowess lines of fit through the densities of transcripts that are judged as significantly different using either scaled or unscaled data, and with or without thresholding. Statistical significance was determined with a FDR < 0.05.

Thus, adding scale uncertainty has two effects both on dispersion. The first is to increase the dispersion estimate for each part and this has its primary effect to reduce the p-values and standardized effect sizes returned by ALDEx2. The second is to reduce the range of dispersions, and rotate them such that the parts with the lowest dispersions have the greatest increase. This increase in dispersion comes about because the underlying distributions are widened with the inclusion of scale uncertainty.

1. Reza,F.M. (1994) An introduction to information theory Courier Corporation.

2. MacKay,D.J. (2003) Information theory, inference and learning algorithms Cambridge university press.

3. Wilde,M.M. (2017) Quantum information theory 2nd ed. Cambridge University Press.

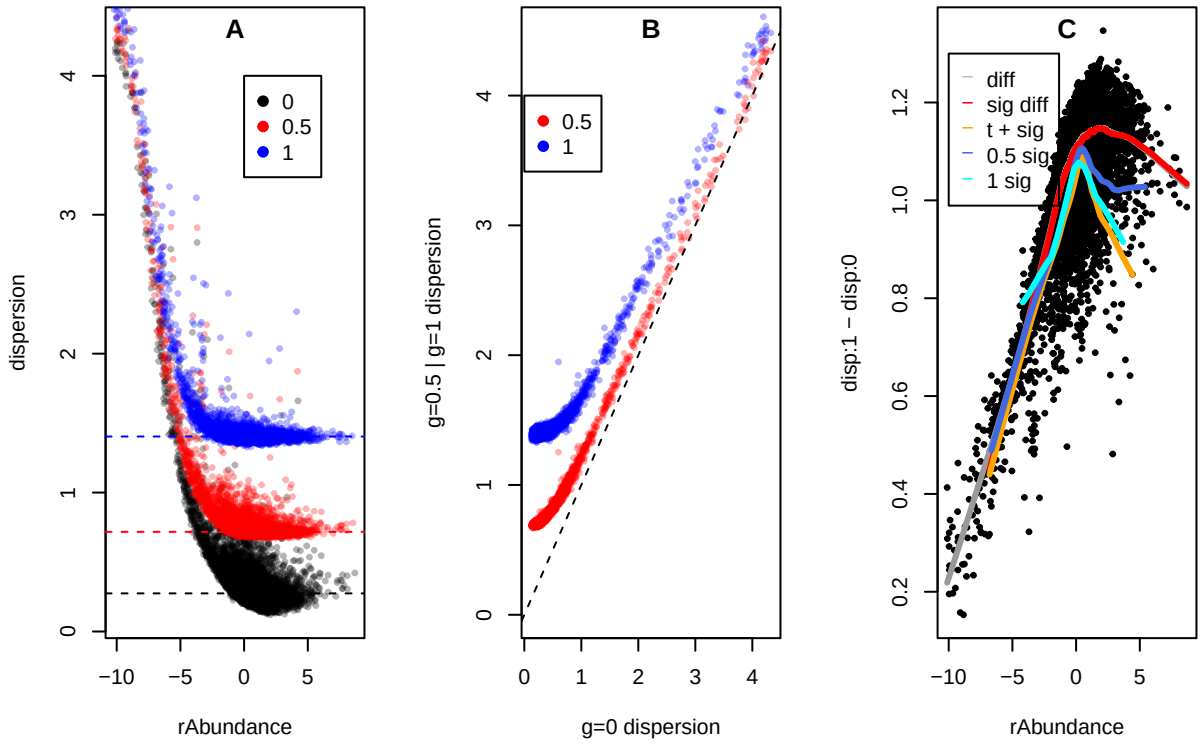


Figure 9: Adding scale uncertainty changes the dispersion distribution. Panel A shows a plot of the expected value for relative abundance vs the expected value for the pooled dispersion as output by aldex.effect. The dashed horizontal lines show the median value for the features with a rAbundance between -0.5 and 0.5. Panel B plots the unscaled vs scaled dispersion, note the non-linear relationship. Panel C plots relative abundance vs the difference between dispersion with gamma set to 0 and to 1 to highlight the rotation that is evident in Panel A. The colored lines indicate the lowess line of fit through the centre of mass of the plot. Line colors represent different populations of points. The grey line is the total population, the red line is the population of significant transcripts with no scale, the orange line is the population of significant transcripts with a difference threshold of about 2.5-fold change, the blue line is the population of significant transcripts with gamma =0.5, and the cyan line is the significant population with gamma = 1.

4. Aitchison,J. (1986) The statistical analysis of compositional data Chapman & Hall, London, England.

5. Aitchison,J. (1982) The statistical analysis of compositional data. *Journal of the Royal Statistical Society: Series B (Methodological)*, **44**, 139–160.

6. Greenacre,M., Martínez-Álvaro,M. and Blasco,A. (2021) Compositional data analysis of microbiome and any-omics datasets: A validation of the additive logratio transformation. *Front Microbiol*, **12**, 727398.

7. Shannon,C.E. (1948) A mathematical theory of communication. *The Bell System Technical Journal*, **27**, 379–423.

8. Jaynes,E.T. and Bretthorst,G.L. (2003) Probability theory: The logic of science Cambridge University Press, Cambridge, UK.

9. Cover,T.M. and Thomas,J.A. (1991) Elements of information theory Wiley, New York.

10. Lecamwasam,R. (2021) Investigations of metrology in optomechanics and quantum information theory.

11. Nixon,M.P., Letourneau,J., David,L.A., Lazar,N.A., Mukherjee,S. and Silverman,J.D. (2023) Scale reliant inference. *https://arxiv.org/abs/2201.03616*.

12. McMurrough,T.A., Dickson,R.J., Thibert,S.M.F., Gloor,G.B. and Edgell,D.R. (2014) Control of catalytic efficiency by a coevolving network of catalytic and noncatalytic residues. *Proc Natl Acad Sci U S A*, **111**, E2376–83.

13. Gloor,G., Macklaim,J., Vu,M. and Fernandes,A. (2016) Compositional uncertainty should not be ignored in high-throughput sequencing data analysis. *Austrian Journal of Statistics*, **45**, 73–87.

14. Gloor,G.B. (2023) AmIcompositional: Simple tests for compositional behaviour of high throughput data with common transformations. *Austrian Journal of Statistics*, **52**, 180–197.

15. Robinson,M.D. and Oshlack,A. (2010) A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biol*, **11**, R25.1–R25.9.

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

17. McMurdie,P.J. and Holmes,S. (2014) Waste not, want not: Why rarefying microbiome data is inadmissible. *PLoS Comput Biol*, **10**, e1003531.