HyperLog™ – A Flexible Log-like Transform for Negative, Zero, and Positive Valued Data

This is a preprint of an article published in Cytometry Part A Volume 64A, Issue 1, 2005. Pages: 34-42, Copyright Wiley-Liss, Inc.
Visit http://www3.interscience.wiley.com for the actual publication.
It is available here for professional use and may not be redistributed.

Author:
C. Bruce Bagwell MD, Ph.D.
Verity Software House, Inc.
PO Box 247
Topsham, ME 04086
U.S.A.

Tel: (207) 729-6767 x102
FAX: (207) 729-5443
EMAIL: cbb@vsh.com

Keywords: Flow Cytometry, Display Data Transform, Graphics, Log/Linear Transform, Log-Linear Hybrid, Compensation, Log-Like Transform

Abstract:

The remarkable success of cytometry over the last thirty years is largely due to its uncanny ability to display populations that vastly differ in numbers and fluorescence intensity on one scale. The log (L) transform, either implemented in hardware as a log amplifier, or in software, normalizes signals or channels such that these populations appear as clearly discernable peaks. With the advent of multiple fluorescence cytometry, spectral-crossover compensation of these signals has been necessary to properly interpret the data. Unfortunately, since compensation is a subtractive process, it can produce negative and zero valued data. The log transform is undefined for these values and as a result, forces computer algorithms to truncate these values, creating a few problems for Cytometrists.

Data truncation biases displays making properly compensated data appear undercompensated; thus, enticing many operators to over-compensate their data. Also, events truncated into the first histogram channel are not normally visible with typical two dimensional graphic displays, hiding a large number of events and obscuring the true proportionality of negative distributions. In addition, the L transform creates unequal binning which can dramatically distort negative population distributions.

The HyperLog™ (HL) transform is a log-like transform that admits negative, zero, and positive values. The transform is a hybrid type of transform specifically designed for compensated data. One of its parameters allows it to smoothly transition from a logarithmic to linear type of transform which is ideal for compensated data. The HL transform is easily implemented in computer systems and results in display systems that present compensated data in an unbiased manner.

Introduction

Since Cytometry's infancy, scientists and engineers have known about the large dynamic range of many molecules on and within cells. If typical immunofluorescence data were presented on a linear scale, it would be difficult, if not impossible, to visually appreciate cell populations separated by large intensity differences for one or more measured parameters.

Solving this display problem required a display transform that not only compressed the absolute dynamic range of fluorescence signals from these molecules, but also normalized their relative differences. The logarithmic (L) transform was almost the perfect mathematical solution to this display problem and cytometers were engineered such that some signals could be logarithmically amplified before digitization, storage, and display (1). Cytometry's success has in large part been due to this ability to visualize immunologically defined populations on a single log scale. The L transform and its inverse are typically defined as,

$$L(x) = \begin{cases} \log(x) \cdot \frac{r}{d} & \text{if } x \geq 1 \\ 0 & \text{otherwise} \end{cases}$$

$$E(y) = 10^{y \cdot \frac{d}{r}}$$

where x, the relative linear channels, is in the set of all real numbers; L(x) is restricted to the interval [0,r), and E(y), [1,10^d); r is the analog to digital (ADC) resolution; and d is the number of decades for x's dynamic range.

The independent variable, x, of the L transform can admit numbers in the real domain, but only those numbers equal or greater to some threshold, normally set to one, are evaluated by the log function. All other values of x cause the L transform to return a zero. The L transform can be implemented in hardware as a logarithmic amplifier or in software as shown above. Either way, the L transform must protect against taking the log of a value less than or equal to zero. The E transform is the inverse of L such that E(L(x))=x for x>=1. The restricted range of x in this equivalency creates a number of problematic issues for both graphics and analysis of L transformed data.

Figure 1A and B illustrates the positive attributes of the L transform (see M&M, Example 1 for details). Figure 1A shows two hypothetical populations on a single parameter linear scale. The events that make up the right-most population, H,

are normally distributed with a relatively arbitrary mean and standard deviation (SD). The events that comprise the left-most population, SL, are multiplicatively scaled to 1% of the H population, maintaining a constant coefficient of variation (CV). The large reduction in intensity and variance compresses the SL population into a few channels making it difficult to appreciate as a separate population.

If the same data are L transformed (see Fig. 1B), the SL population is visually distinct with the same apparent SD as H. The positive characteristics of the L transform are that it minimizes the mean and variance differences for populations with similar CV's.

Unfortunately, the L transform also has some well-known disadvantages. Zero and negative valued data are undefined and must be truncated to zero. In addition, there are pronounced binning effects that dramatically affect the distribution of populations near the origin.

To better illustrate these problems, Fig. 1C again shows a parent population, H. The H population events are translocated to 1% of their original intensity, forming a TL population that has both negative and positive values (see M&M, Example 2 for details). When these data are L transformed, TL is distributed over much of the axis. A peak is observed in the middle portion of the axis that continually decreases in frequency until the origin, where about half the TL population is accumulated into the first histogram channel. For data that are translocated to or near the origin, the L transform is not a suitable because low intensity population SD's or variances are not preserved and their distributions are highly distorted.

The ability to analyze and graph data defined over the real domain of numbers with a log-like transform has wide ranging applications in many scientific disciplines. A solution to the log problem was first published by Johnson in 1949 (2) where he proposed using a generalized inverse hyperbolic sine (IHS) function, Su, to define a log-like transform that spanned the real number domain from negative to positive infinity.

In 1981 Bickel and Doksum (3) proposed a modification to the well-known Box-Cox (BC) transform (4) that was also log-like and would admit negative numbers. Later in 1989, Burbige (5) compared the IHS and BC transforms and came to the conclusion that the IHS had more merits. The IHS transform has since been selected as a surrogate log transform for signed data in numerous publications from various scientific disciplines ranging from agriculture to astrophysics (6, 7).

Many traditional statistical methodologies (e.g. regression, ANOVA) require that data be normally distributed and have constant variance. As seen in Fig. 1B, the L transform does have a "stabilizing" influence on SD or variance with multiplicatively scaled data and thus has been extensively used to transform this type of data into a form amenable to statistical analyses with the constant

variance requirement. In some cases, however, data will have negative, zero, and positive values which complicates the use of the log transform. In these cases, other transformations such as the generalized logarithm (GLOG, 8), Started Logarithm (SL, 9-10) and the log-linear hybrid (LL, 11) have been shown to be log-like, preserving the variance with multiplicatively scaled data, that also can accept extreme positive and negative values. In 2003 the performance of these three different transforms was evaluated by Rocke and Durbin (12) where they found that the GLOG transform was the best choice for gene-expression microarray data.

In the field of Cytometry, this issue is important since compensation is a process that does involve subtraction or translocation (13), resulting in negative, zero, and positive value data. Compensated data have multiplicative and translocated characteristics. The multiplicative attribute is fundamental to measurement processes such as found in Cytometry. DNA histograms are an excellent example, the SD of the G2M population is very close to twice G0G1. The CV of a population is the ratio between its mean intensity and variability and therefore, for the most part, CV is preserved in measured data. As was shown in Fig. 1B, this type of data is particularly amenable to the L transform since constant CV translates to constant SD's or variances, making the populations clearly distinguishable.

The other characteristic of compensated data is translocation. In the compensation process, a population is moved or translocated via subtraction along one or more of its parameters to or near the origin of an axis without significantly changing its variance. As shown in Fig. 1D, the L transform is particularly unsuited for this type of data. An optimal transform for compensated data must be able to handle both of these characteristics.

In 2002 Parks and Moore (14) proposed using a modified IHS transform as a replacement for the L transform for compensated data called the Logicle/Biexponential (LB) transform. Since it is not necessary to protect the family of IHS transforms from negative or zero data, there is no inherent bias in the transformed data. Thus, properly compensated data does not appear over-compensated with the LB transform, which eliminates a major source of error in Cytometry data analysis. Parks and Moore generalized the IHS transform to better meet the needs of displaying compensated data. Parameters were added to the hyperbolic sine exponentials to better control the size of a linear window through the origin, which helped eliminate low intensity binning artifacts and controlled the variance of negative populations.

In 2003 the HyperLog (HL) was released in software (15) and later presented (16) as a log-like transform optimized for compensated data. The HL transform was engineered as a hybrid function. One component of the function is ideally suited to the multiplicative nature of compensated data, and the other, to its translocated nature. The transform smoothly transitions from one type of

transform to the other, depending on the analysis or graphics needs. The HL transform is similar but not identical to (LL) and fundamentally different from the IHS types of transforms. The HL transform is an inverse hybrid linear/exponential function that is defined over the real-numbered domain. The HL transform is flexible enough to represent unbiased compensated data with visually pleasing axes (see Fig. 6A, 6B, 7).

The purpose of this paper is to mathematically describe the HL transform, investigate its properties, especially in eliminating binning artifacts, and demonstrate its usefulness in representing axes for Cytometry data. Although the HL transform was designed specifically for the unique characteristics of compensated data, it is flexible enough to be used in a more general manner.

Materials and Methods

Simulations

Simulations were performed by MathCad 2001 Professional (MathSoft Engineering & Education, Inc,

The following examples have only been designed to demonstrate the problems associated with the logarithmic transform for translocated data. These examples do not incorporate data measurement errors due to photon-counting or digitization.

Example 1: Multiplicatively Scaled Data (see Fig. 1A, B)
Populations are normally distributed using the Box-Muller equation (17) from 10,000 synthesized events. The mean and standard deviation (SD) of the unscaled population, H, are 600 and 50 respectively. The linear scale and L transformed scales range between 0 and 1000 (r=1000, d=3). As described for the L transform, r is the ADC resolution and d is the number of decade dynamic range. The low-intensity population (SL) is formed by assuming constant CV and scaling the mean and SD to 1% of their original values, 6 and 0.05 respectively.

Example 2: Translocated Data (see Fig. 1C, D)
The same method as described for Example 1 was used to create the high-intensity population, H. The low-intensity population, TL, is formed by translocating H to 1% its original value, 6, while preserving the original SD, 50.

Example 3: Translocated Data (see Fig. 5)
The same method was used as described for Example 2, except that the TL population was translocated to the origin.

Results

The inverse of the basic HyperLog transform is given by,

$$\begin{cases} e^{a\cdot y} + b\cdot y - 1 & \text{if } y \geq 0 \\ -e^{-a\cdot y} + b\cdot y + 1 & \text{otherwise} \end{cases}$$

where a and b are constants and y is defined over the

real-numbered domain.

A more practical base-ten form of this transform is,

$$EH(y,b) = \begin{cases} 10^{\frac{d}{r}\cdot y} + b\cdot\frac{d}{r}\cdot y - 1 & \text{if } y \geq 0 \\ -10^{\frac{-d}{r}\cdot y} + b\cdot\frac{d}{r}\cdot y + 1 & \text{otherwise} \end{cases}$$

The constant d, decades, and has a similar

definition as described for the L transform; at

y=r, EH(y,b) is at its maximum.

The EH transform is continuous and symmetric about the origin as shown in Fig. 2 and its zoomed inset. Three different b coefficients, 0, 35, and 100, are shown to demonstrate how b affects the transform through the origin (see Fig. 2 inset). The H transform is also shown to illustrate how EH(y,b) approaches E(y) for y>>0. Note that the number of decades for these transforms is three in order to correspond to the example data shown in Fig. 1. With four or higher number of decades, the relative effect of the linear term in the EH transform in the high intensity region will be a lot smaller than that shown in Fig.'s 2 and 3.

The HyperLog (HL) transform is the inverse of EH which is found by using a suitable root finding algorithm (18) and restricting the roots to non-imaginary values.

$$HL(x,b) = \text{root}(EH(y,b) - x)$$

where root(…) is a standard root finding

algorithm (18) that finds y such that EH(y)=x.

Fig. 3 shows the HL transform with b coefficients 0, 35, and 100 as well as the corresponding L transform. For extreme values of x, the HL and L transform have very similar characteristics.

Binning Effects and Population Splitting

Axes transformations can radically change the shape of histograms due to unequal bin sizes as was shown for the L transform in Figure 1D. The reason unequal bin sizes cause distortion is depicted in Figure 4. Panel 4A shows a histogram with four equal sized bins containing the same number of events in each bin. When a transform is applied to the x bin boundaries, the sizes for each histogram bin can change as shown in Panel 4B. Since the frequency in each bin must be preserved, the height of the histogram bins must change inversely as their widths change. In the case of the L transform, this binning effect creates a peak in the middle of the axis with a decreasing continuum approaching the origin as shown in Figure 1D.

The HL transform with b=0 also has binning effects near the origin, often splitting negative populations into two peaks (see Fig. 5 b=0 distribution). This artificial population splitting is a graphical distortion that should be minimized in order to unambiguously appreciate separate clusters.

By increasing the HL transform's b-coefficient, this peak splitting can be eliminated as shown in Fig. 5 for b=35, 100, 277, and 333. A simple approach to finding an appropriate b-coefficient would be to manually set it to a value large enough to eliminate peak splitting for a number of data sets. For four decade Cytometry data, a single b-coefficient of 100 can be used on a wide variety of data sets with varying amounts of applied compensation (see Fig 6A, 6B).

The critical b-coefficient, b_c, that just eliminates peak-splitting for an arbitrary population at the origin is calculated as follows:

Let F(i) return the number of events for the untransformed population's channel i. By using the chain rule for inverse functions and taking the derivative of the i>=0 part of EH(I,b), the binning effect on F(i) is given by,

$$G(i) = F(i)\cdot\left[\frac{d}{r}\cdot\left(10^{\frac{d}{r}\cdot HL(i,\,b)}\cdot\ln(10) + b\right)\right] \qquad \text{Eq. 1}$$

for i>=0

If we assume that the SD of the negative population is sd; we can write a characteristic function, C, as

$$C = \sum_{i=0}^{int(sd)} \Delta G(i)^2 \qquad \text{Eq. 2}$$

$$\Delta G(i) = G(i+1) - G(i)$$

By using a suitable minimization method on C varying b (19), we can find b_c. A negative distribution will be very flat between zero and sd when HL transformed with $b=b_c$. If b is less than b_c, the data will appear split. If $b>>b_c$ it will appear as a single population. For our example data, b_c is calculated as 35.332. See Fig. 5, b=35 to appreciate the flat distribution that is associated with b_c for Example 3 data.

A reasonable approach to finding an "appropriate" b-coefficient might be to maximize the Fisher distance function for a negative (mean2=0, sd2) and positive (mean1, sd1) population in HL transformed space. The Fisher distance function is given by,

$$FD(mean1, mean2, sd1, sd2) := \left| \frac{|mean1 - mean2|}{\sqrt{sd1^2 + sd2^2}} \right|$$

The distance is maximum when the means between the two populations are far apart and their respective variances are relatively small. The HL transform characteristic function that when maximized (19) yields the "optimal" Fisher distance between negative and positive distributions is given by,

$$\frac{HL(mean1, b)}{\sqrt{(HL(mean1 + sd1, b) - HL(mean1, b))^2 - HL(sd2, b)^2}}$$

For Example 3 H and TL populations, the b-coefficient that maximizes the Fisher distance is 276.67 (see Fig. 5, b=277).

The log transform normalizes extreme positive valued data and stabilizes the variance for multiplicatively scaled data. An interesting question to ask is whether the HL transform with a suitable b-coefficient, b_t, can stabilize the variance of translocated data to the origin (see Fig. 1C and D). When the b-coefficient is very high, the HL transform will be close to linear and would conserve the variance in a way that is similar to Fig. 1C. Unfortunately, the transform would have very little log-like characteristics which would be undesirable.

One can, however, find a b-coefficient, b_t, where the linear SD or variance of a positive population is approximately the same as the SD of the translocated HL transformed population. For this equality to hold, then,

$$10^{\frac{d}{r}\cdot sd} + b_t\cdot\frac{d}{r}\cdot sd - 1 = sd$$

Solving for b_t,

$$b_t = \frac{r}{d}\cdot\left(\frac{1}{sd} - 10^{\frac{d}{r}\cdot sd - \log(sd)} + 1\right) \qquad \text{Eq. 3}$$

If the term d*sd/r << log(sd), which it normally is for typical values of d, r, and sd; the above equation simplifies to,

$$b_t = \frac{r}{d} \qquad \text{Eq. 4}$$

Since b_t is relatively independent of sd and the positive population mean, the above equation is quite general for most cases. For the Example 3 data, b_t is calculated as 333.33 (see Fig. 5, b=333). Over the range sd=1..100, the sd of the population translocated to the origin varied by less than 1% from its original sd with b_t=r/d for the Example 3 data.

The ability of the HL transform to accept negative numbers usually requires that a lowest negative HL transformed value be determined. This boundary is either set manually or calculated as the value that excludes some percentage (e.g. 5%) of negative events.

Multi-parameter data generated from four different fluorescent proteins are traditionally difficult to compensate because of their relatively weak fluorescence and significant signal-crossover. Therefore, this type of data is a good test case for comparing HL and L transforms (see Fig. 6). As shown in Fig. 6 Panels A, B, and C; HyperLog axes can represent compensated data in an unbiased display format. Compare the same data with traditional log axes (see Panels D, E, and F). The encircled population in Panel D clearly shows compensation bias for the L transform, making the data appear to be under-compensated. The corresponding population in Panel A does not show this bias. Panel F uses an isometric display of Panel D data to better demonstrate the truncation and binning effects of the log transform, resulting in high number of events directly on

the axes. The HL transformed data shown in Panel C does not show these undesirable effects.

HyperLog axes also have a well-balanced appearance (see Figs 6A, 6B, 7). Part of the reason is that the linear axis distance is approximately the same as the adjacent log axis distance (see Fig. 7 Linear and Log distance arrows). Stating this relationship mathematically,

$$R = \frac{HL(10 \cdot b, b) - HL(b, b)}{HL(b, b) \cdot 2}$$

where R~1 for b=0..100.

The right-most inset in Fig. 7 shows that R is approximately one for a wide range of b-coefficients, 0..100, giving the axis its well-balanced and aesthetically pleasing appearance.

Discussion

The HyperLog (HL) transform is a log-like transform for display or analysis systems that need to span the real numbered domain. The inverse of the HL transform, EH, is an exponential/linear hybrid function designed specifically for compensated data. Each value that comprises the HL transform is calculated by a suitable root finding routine. If the transform is to be used in software, these transformed values are normally entered into a lookup table with linear interpolation. This requirement is not really a problem since transforms like log and exp are normally implemented with lookup tables anyway since transcendental functions are inherently computationally expensive.

Log and Log-like transforms can create binning artifacts that can be confusing if not minimized or eliminated. For example, the HL transform with b=0 can cause a population located at the origin to appear as two separate populations defined in the negative and positive numbered domains (see Fig. 5, b=0). Negative population splitting is easily eliminated by increasing the b-coefficient to some positive value (see Fig 5, $b>b_c$).

The b-coefficient smoothly transitions the HL transform between two different types of transforms. With a low b-coefficient, the transform is more log-like and thus stabilizes population variances that are multiplicative scaled (e.g. constant CV). As the b-coefficient approaches b_t, the transform becomes more optimal for stabilizing the variance for translocation or subtractive scaling. The “appropriate” b-coefficient is ultimately determined by which of these two types of scaling is most important for the analysis or graphics at hand.

For graphic representation of compensated data, the lower limit of this appropriate b-coefficient is b_c, since negative population splitting is not a desired characteristic. The upper limit should be close to b_t since optimizing the transform for translocated data represents an extreme for compensated data and results in a significant reduction of the separation between negative and positive populations. One possible approach to finding an optimal b-coefficient that hopefully would fall between the b_c and b_t extremes would be to maximize or minimize some characteristic function that has some desired functionality.

One obvious characteristic function would be the Fisher distance between a negative and positive population. It makes intuitive sense that a good transform would attempt to separate these two populations as best as possible. At first this approach appears to be an ideal solution, but finding this optimal HL transform is relatively easy for the simple example as described in this paper, but becomes complicated with real data. With real data there may be many positive populations with ill-defined boundaries and it may not be clear which one to use for the optimization. It also may be that it is more important to the investigator to separate two positive populations than a positive and negative population.

Another possible approach would be to find the HL transform that minimizes the difference between all the observed population variances. This approach also has problems for real data. It again requires some kind of method for identifying these populations, which may be very difficult to do automatically. For some, this approach might create an aesthetically pleasing graph; but for others it may distort the log nature of the transform and create strange looking axes. Another approach might be to find the lowest b-coefficient that makes the negative population similar to a Gaussian (16). This approach would appeal to those that wish to minimally perturb the log-like nature of the transform.

Transforms defined over the entire real domain like HL neither create nor destroy information; they only rearrange information. When used as a graphics transform, a family of HL transforms formed by different b-coefficients are essentially equivalent from the point-of-view of information content and the decision for finding the “best” b-coefficient is somewhat arbitrary and largely influenced by a user’s or designer’s sense of aesthetics.

Complicating this issue is that any method that optimizes the HL transform or other similar log-like transforms based on data will ultimately cause the axis to change as the data changes. This variability can create problems for investigators who wish to compare data. Comparing data is so common in the scientific process, that the “changing axis” problem in many ways outweighs any benefits in optimizing the transform.

Another aspect to this decision is the complexity of the transform axis. If the axis appears strange in any way, it tends to distract the viewer from appreciating the information contained in the data, a well-known problem in graphic design. If the b-coefficient is a power of ten, the HL transformed axis smoothly transitions from a complete linear scale to a log scale yielding an axis that is well-balanced and simple to interpret (see Fig. 6A, 6B,7).

When HL is used in graphic displays, the “best” b-coefficient is normally the simplest. For four decade log data, a value of 100 works for just about all cases. The axes are consistent from one data set to another and thus have the important advantage of being comparable. The axes are interpretable by any scientist that is familiar with linear and logarithmic axes and don’t appear needlessly complicated. Thus, even though the HL transform is amenable to optimization techniques, software engineers should be discouraged from forcing the user to use a data-dependent optimized result.

Other types of transforms (e.g. GLOG and LB) can also be used to display compensated data in an unbiased manner. The advantages and disadvantages of each of these transforms are relatively minor, especially for graphic displays. The ultimate success or failure of these approaches will largely be determined by the aesthetics of the axes they produce.

In summary, the HL transform is a log-like transform that was originally designed specifically for the display of compensated data. Its ability to smoothly transition between exponential and linear scales gives it desirable features that may have general application beyond the visualization of unbiased compensated data.

Acknowledgements

Mark Munson, Don Herbert, Chris Bray, and Ben Hunsberger all played important roles in creating this manuscript.

Literature Cited

1. Loken MR, Parks DR, and Herzenberg LA. Two-color immunofluorescence using a fluorescence-activated cell sorter. J Hist Cyto 1977;25;899-907.

2. Johnson NL. Systems of frequency curves generated by methods of translation. Biometrika 1949;36;149-176.

3. Bickle PJ and Doksum KA. An analysis of transformations revisited. J Amer Statist Assoc 1981;76;296-311.

4. Box GEP and Cox DR. An analysis of transformations. J Royal Statist Soc., Ser B 1964;26;211-243.

5. Burbidge JB, Magree L, and Robb AL. Alternative transformations to handle extreme values of the dependent variable. J Amer Statist Assoc 1988(March);Vol 83;No. 401;123-27.

6. Layton DF. Alternative approaches for modeling concave willingness to pay functions in conjoint valuation. Amer. J. Agr. Econ 2001;83 (No 5);1314-1320.

7. Guarnieri MD, Ortolani S, Montegriffo P, Renzini A, Barbuy B, Bica E and Moneti A. Infrared array photometry of bulge globular clusters. Astron Astrophys 1998;331;70-80.

8. Munson P. A 'Consistency' test for determining the significance of gene expression changes on replicate samples and two convenient variance-stabilizing transformations. GeneLogic Workshop on Low Level Analysis of Affymetrix GeneChip Data 2001.

9. Tukey JW. On the compariative anatomy of transformations. Ann of Math Stati 1964;28;602-632.

10. Tukey JW. Exploratory data analysis. Addison-Wesley, Reading, MA 1977.

11. Holder D, Raubertas RF, Pikounis VB, Svetnik V, and Soper K. Statistical analysis of high density oligonucleotide arrays: a safer approach. GeneLogic Workshop on Low Level Analysis of Affymetrix GeneChip Data 2001.

12. Rocke DM, Durbin B. Approximate variance-stabilizing transformations for gene-expression microarray data. Bioinformatics 2003;Vol 19;no. 8;966-972.

13. Bagwell CB and Adams EG. Software spectral overlap compensation for any number of flow cytometry parameters. NYAS 1993;20;677;167-84.

14. Parks DR & More W. Presentation, Cytometry Development Workshop, Asilomar Conference Center in Pacific Grove, California Oct 18-21 2002.

15. HyperLog released commercially August 2003, WinList 5.0 SP4, Verity Software House, Inc.

16. Bagwell CB. An alternative display transform for cytometry data, ISAC Montpellier France, 2003 May 22-28.

17. Box GEP and Muller ME. A note on the generation of random normal deviates. Ann Math Stat 1958;29;610-611.

18. Ridders CJF. IEEE Transactions on circuits and systems 1979;Vol CAS-26;979-980.

19. Numerical Recipes in C. Second Edition, Cambridge University Press, 1992;425.

Figures

Figure 1

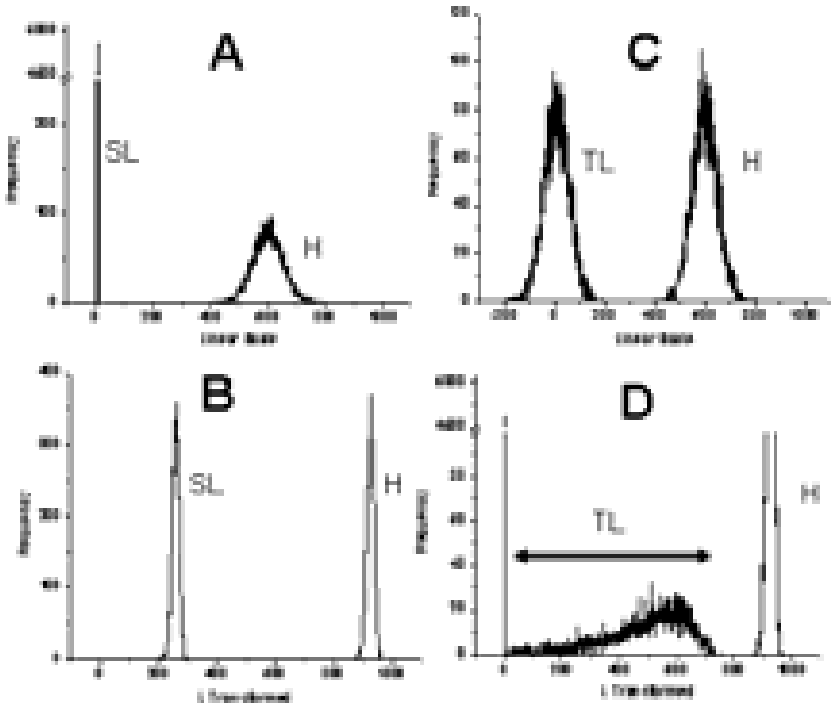

Figure 2

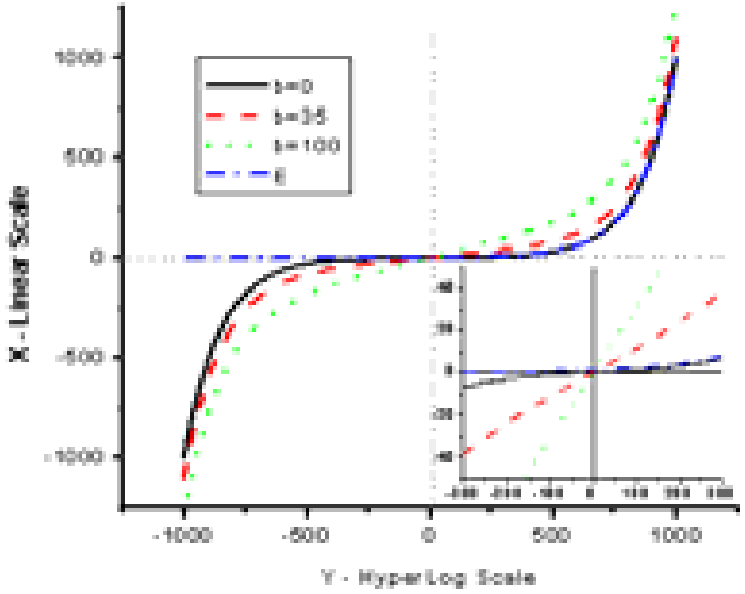

Fig 2.

Figure 3

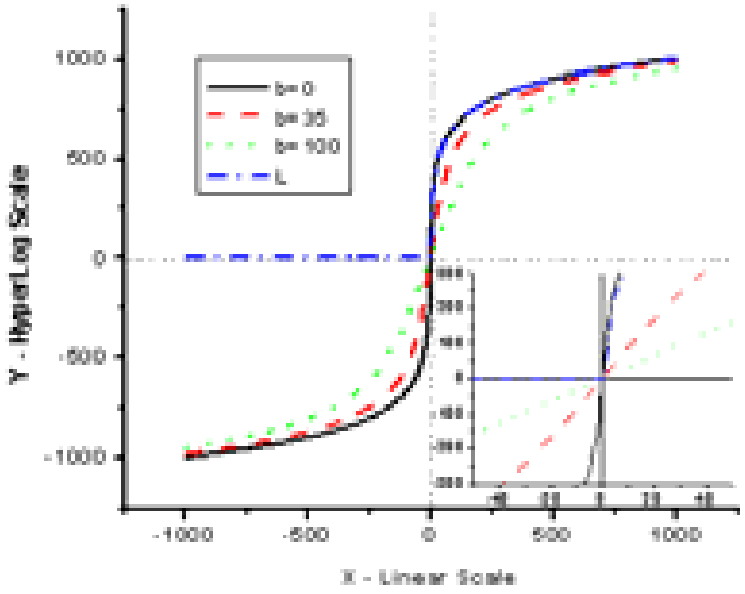

Fig 3.

Figure 4

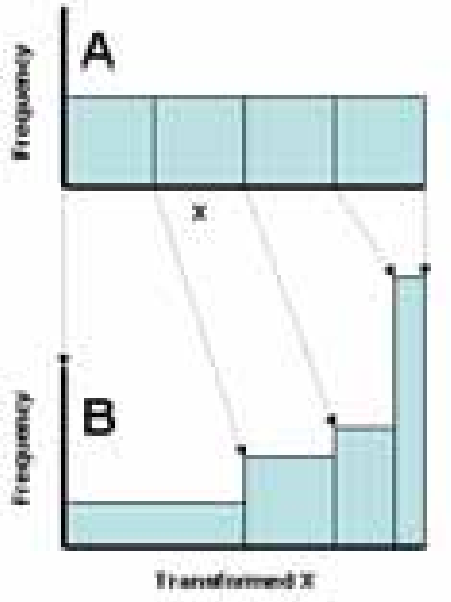

Fig 4.

Figure 5

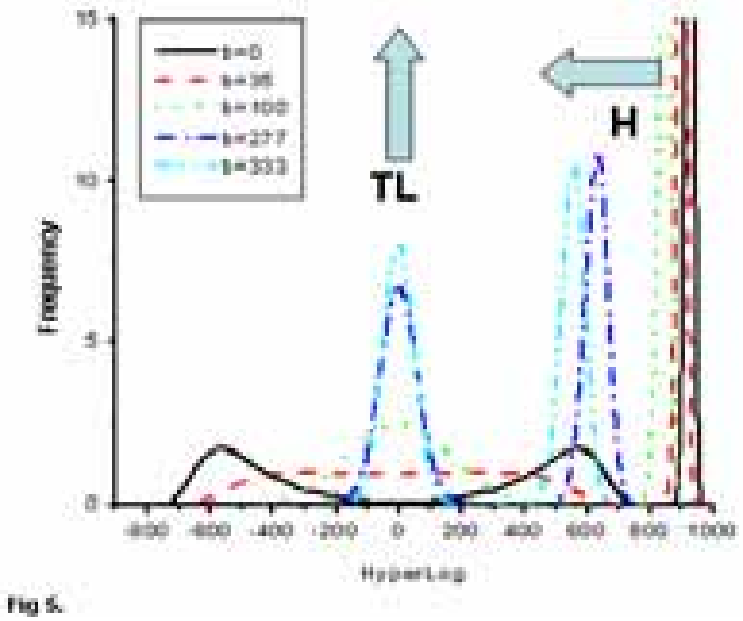

Figure 6

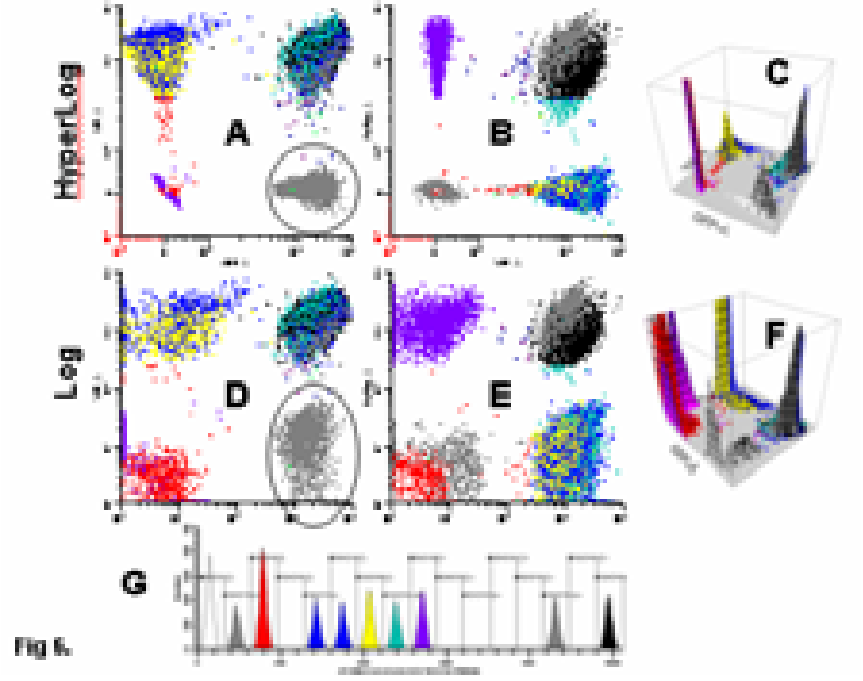

Figure 7

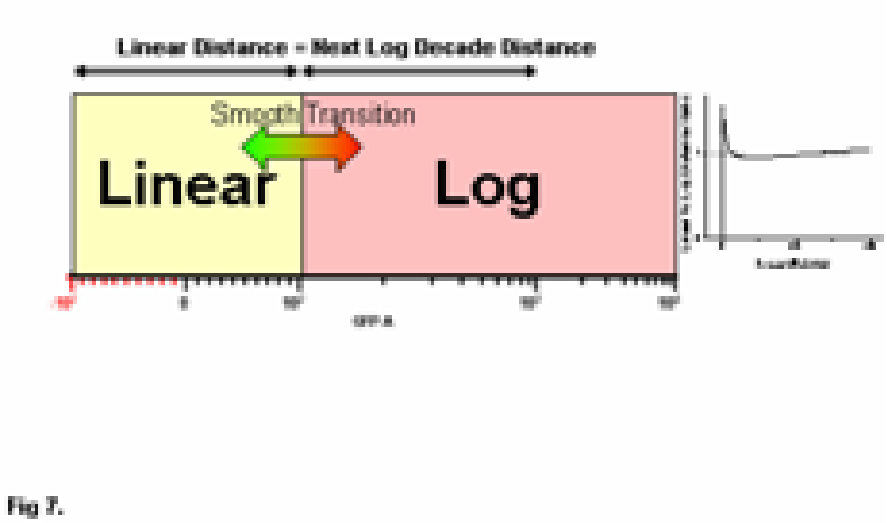

Fig 7.

Figure Legends

Fig. 1: Effects of log transformation for multiplicatively scaled and translocated data.

Panels A and B show low (SL) and high (H) intensity populations (see M&M, Example 1 for details) for linear and L transformed scales. Each of these populations was synthesized to have the same coefficient of variation (CV). In the linear domain (Panel A), SL is not readily apparent as a separate population since it is compressed into a few channels. The L transformed scale (Panel B) reduces mean and variance differences making the populations clearly discernable.

Panels C and D also show low (TL) and high (H) intensity populations (see M&M, Example 2 for details). The TL population was translocated to 1% of the H mean value without changing its standard deviation (SD). When the TL population is presented on a log scale, the population is highly distorted and distributed over much of the axis, demonstrating the unsuitability of log transforms for translocated data.

Fig 2: Inverse HyperLog (EH) transform

A family of EH transforms is displayed with the exponential (E) transform. The EH transform (r=1000, d=3) is symmetric about the origin and predominately linear through the origin. The EH transform approaches the E transform (blue, dash dot) with the identical resolution (r) and decades (d) values for y>>0. The inset shows a zoomed region of the EH transforms demonstrating how they change with various values of the b-coefficient, b=0 (black, solid), b=35 (red, dash), and b=100(green, dot).

Fig 3: HyperLog (HL) transform

The HL transform is symmetric about the origin and predominately linear through the origin. The HL transform approaches the L transform (blue, dash-dot) for x>>0. The inset shows a zoomed region of the HL transforms demonstrating how they change with various values of the b-coefficient, b=0 (black, solid), b=35 (red, dash), and b=100 (green, dot).

Fig 4: Transformation and binning effects.

Panel A shows a histogram with four equal sized bins containing the same number of events in each bin. When a transform is applied to the x bin boundaries, the sizes for each histogram bin can change as shown in Panel B. Since the frequency in each bin must be preserved, the height of the histogram bins must change inversely as their widths change.

Fig 5: HyperLog transforms and translocated Data

The H and TL populations (see Example 3 for details) are HL transformed with various b-coefficients (0 (black solid), 35 (red dash), 100 (green dot), 277 (blue dash-dot), and 333 (lavendar dash dot dot). As the b-coefficient is increased the negative population splitting is eliminated and the variance becomes smaller signified by the upward arrow above the TL population. With increasing b-coefficients the H population is moved to the left, reducing the separation between the H and TL populations.

Fig 6: HyperLog and Log transforms with compensated cytometry data.

The first two Panels, A and B, show three correlated fluorescent protein distributions with HyperLog (HL) axes demonstrating the ability of the HL transform to represent compensated data in an unbiased display format. All the axes presented in Panels A, B, and C have b-coefficients set to 100. Panel C shows an isometric display of Panel A demonstrating that the clusters are well delineated and not distributed along the axes.

Panels D, E, and F show the same data with traditional log axes. The encircled population in Panel D shows compensation display bias; whereas, Panel A's corresponding population does not. The log transformation truncation and binning effects, resulting in large number of events on the axes, are demonstrated in Panel F. Panel G presents all the color-coded combinations of the four correlated fluorescent protein expressions in the FCS file.

The data are kindly provided by Teresa and Robert Hawley, George Washington University Medical Center.

Fig 7: HyperLog axis aesthetics.

HyperLog axes have a ratio between the linear and the adjacent log regions of approximately one (see right-most graph inset) giving the axis a well-balanced and aesthetically pleasing appearance. The right-most inset shows that the ratio is approximately one for a wide range of b-coefficients, 0..100.