5-1-2002

Quantifying Bimodality Part I: An Easily Implemented Method Using *SPSS*

B. W. Frankland
Dalhousie University, Brad.Frankland@dal.ca

Bruno D. Zumbo
University of British Columbia, bruno.zumbo@ubc.ca

Quantifying Bimodality Part I: An Easily Implemented Method Using *SPSS*

B. W. Frankland
Department of Psychology
Dalhousie University

Bruno D. Zumbo
Measurement, Evaluation, & Research Methodology
University of British Columbia

Scientists in a variety of fields are faced with the question of whether or not a particular sample of data are best described as unimodal or bimodal. We provide a simple and convenient method for assessing bimodality. The use of the non-linear algorithms in *SPSS* for modeling complex mixture distributions is demonstrated on a unimodal normal distribution (with 2 free parameters) and on bimodal mixture of two normal distributions (with 5 free parameters).

Keywords: Mixture modeling, Bimodality

Introduction

Research in the social, natural and health sciences (e.g., epidemiology, health care, education, psychology, sociology) are often faced with the question of whether or not a particular sample is best described as unimodal or bimodal (e.g., Do & McLachlan, 1984; Hoffmann & Miller, 1998; Knoll, Garver, Ramberg, Kingsbury, Croissant & McDermott, B., 1998; Johnson & Yantis, 1995; Reischies, Schaub & Schlattmann, 1996; Roeder, 1994; Sussman, 1999; Volbrecht, Nerger & Harlow, 1997). Issues of bimodality surface in fields as diverse as astrophysics (Roeder, 1990) and medicine (Ottong & Garver, 1997). Fundamentally, this is the question of whether the set of data was extracted from one population or two populations (note that one can ask this question for data that is between subjects or within a single subject). The generic terms unimodal and bimodal are too vague for analysis, so to make the question more specific (and therefore testable), one can ask, do the data represent a single unimodal normal population? or do the data better represent a bimodal mixture of two normal distributions?

The current work is focused on the development of a fairly simple, but general, procedure for testing such alternatives. It should be noted that, although the techniques developed herein are focused on putative mixtures of normal distributions, they can be applied, in principle, to the

Bradley Frankland is Assistant Professor, Psychology Department, Life Sciences Centre, Dalhousie University, Hailifax, Nova Scotia, Canada, B3H 4J1. E-mail him at franklan@is.dal.ca. His areas of expertise include computational modeling, statistical methods, and cognition. Bruno D. Zumbo. is Professor, Measurement, Evaluation, and Research Methodology, and Associate Member of the Department of Statistics, University of British Columbia. His areas of statistical research include performance characteristics of standard procedures (both parametric and nonparametric) under non-standard conditions, multivariate analysis, and statistical theories of measurement.

comparison of any set of theoretical distributions. The choice of the normal distribution as the starting point was not arbitrary: If the data is unimodal, then it is likely that the normal distribution is at least a good approximation[1] and if the data is bimodal representing two populations, then it is likely that the normal distribution is a good approximation for each population. Regardless of the choice of distribution, the main point is to present a methodology that permits any researcher to quickly and easily discover the most appropriate parent population for a given data set.

In broad strokes, mixture modeling concerns modeling a statistical distribution by a mixture (or weighted sum) of other distributions. This modeling strategy comes under a variety of names in various disciplines: unsupervised concept learning (in artificial intelligence), intrinsic classification (in philosophy), or, classification, clustering, and numerical taxonomy.

It is admitted, a priori, that the solution offered herein is not an analytical solution to the question of bimodality. The point was to develop an accessible, flexible and, most importantly, accurate method that could be used to test any number of hypotheses. To achieve accessibility, the commercially available statistical package *SPSS* was used: It could be added that any statistical package should be capable of a comparable level of analysis. Flexibility demanded that the procedure place few restrictions on the nature of the hypothetical parent population. Too often, methods for exploring multimodal population distributions require constraints that are outside the interests of the research at hand.

For example, it is often assumed that the parent population *must* be normal distributions or mixtures of such (e.g., Yellott, 1971); such an assumption may not be reasonable in all cases. Although the current work is focused on mixtures of normal distributions, the techniques can be applied to any distribution. At the other end of the spectrum, to avoid any assumptions of nature of the parent distributions, some researchers (e.g., Yantis, Meyer & Smith; 1991) assumed that it possible to obtain, by empirical

means, pure unimodal distributions (separate experiments are run to obtain data that is "purely" unimodal) from which the mixture distribution can be constructed. Even if/when possible, the requirement is unwieldy for many experimental designs, particularly for quasi-experimental designs. Flexibility has the added benefit of permitting the analysis to be tailored to the experimental design, rather than vice versa[2].

Although an attempt has been made to make these routines accessible, the desire for accuracy precluded the simpler solutions such as analyses based on increased kurtosis (e.g., Hoffman & Miller, 1998), increased variance (Eriksen & Eriksen, 1972) or a shift in the mean (e.g., Eriksen & Yeh, 1985). Such solutions tend to be inadequate because they cannot distinguish a mixture from a non-normal, but unimodal and symmetric, distribution (e.g., the t-distribution or the Cauchy). To complicate the issue, a mixture distribution may appear to be unimodal and symmetric (see Yantis et al. for more extensive discussion of alternative approaches). Interestingly, an advantage of the SPSS routines is that they provide a wealth of statistics pertaining to the fit of the solution.

Method

A set of empirically determined data is compared to two hypothetical population distributions: the first population is a unimodal normal distribution and the second population is a bimodal mixture of two normal distributions. Several steps are required to fit each population to the empirical data. First, the empirical data must be binned to create a histogram. Second, this binned data must be compared, using the non-linear regression algorithm of SPSS, to each hypothetical population. Finally, the fits for the two populations must be compared. For reasons of space and clarity, the development of a rigorous likelihood-ratio test of the best choice will be presented in a forthcoming paper. In the usual case, a researcher will have a single, or limited number of data sets. Hence, procedures are demonstrated within that context.

The crucial, though often ignored, step involves binning the data to create a probability histogram that accurately represents the data distribution. Critically, the most appropriate bin size (aka: bin widths) must be chosen, because an inappropriate choice will actually result in the loss of information. When working with a single or limited number of data sets, this can most easily be done "by eye/hand". Various statistical or spreadsheet packages will also provide one with binned data but it must be remembered that these packages are generally tailored more to the goal of a pleasing presentation (e.g., limiting the number of bins to 20) than to accurate representation. A commonly accepted guideline is that no bin should have less than 5 counts[3]. In addition, when choosing bin sizes by hand, one can vary the bin widths, using narrow bins in the centre of the distribution and wide bins at the edges of the

distribution which might result in a more accurate representation while maintaining a reasonable minimum bin count. Regardless of how one achieves it, for subsequent analyses, the SPSS data file is expected to contain the following variables:

binnum	bin number (not actually used, but useful for humans)
bincnt	count per bin
xl	bin lower limit in the original scores
xc	bin centre in the original scores
xu	bin upper limit in the original scores
totalcnt	total counts (total number of data points)

It must be noted that before fitting, the count within each bin will be converted to a probability because fitting is based upon the numerical integration of the normal distribution. Hence, one needs the total number of counts (number of data points) as well as the counts per bin. The variable total counts (totalcnt) will be the same for all bins, and could be entered using a compute statement. One could simply enter probabilities per bin in place of count per bin and total count. Pragmatically, however, since one must maintain many significant digits, it is often easier to enter two integers (bincnt and totalcnt) than one long real number (e.g., prob) — and one is less likely commit a data entry error with simple integers.

To fit the data to the population, the fitting algorithm converts the theoretical population distribution into a histogram with bin sizes that are matched to the bin sizes of the data (i.e., the real data determines the bins sizes for the theoretical population). Then, the procedure adjusts the population parameters so that the counts per bin in the binned theoretical histogram matches (as best as possible) the counts per bin in the binned data histogram. Fitting is accomplished by the non-linear regression routines provided by SPSS.

Generally, a non-linear regression algorithm has four basic components: (1) the data to be fitted, (2) the function to be fitted, (3) the free parameters of the function to be fitted, and (4) the error or loss function of the function to be fitted. In this work, the data to be fitted is the proportion per bin in the data histogram (Y_i, where i is the bin number). The function to be fitted is the proportion of the hypothetical population distribution that should fall within each bin, i (recall that bin sizes are determined from the data histogram). These proportions were determined from the theoretical unimodal normal distribution or from the theoretical bimodal mixture of two normal distributions. In the case of the unimodal normal distribution, the free parameters to be fitted are, the mean (μ) and standard

deviation (F, or variance, F^2):

$$N(\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}}e^{\frac{-(Y_i-\mu)^2}{2\sigma^2}} \tag{1}$$

In the case of the bimodal mixture of two normal distributions, there is a mean $(:_1,:_2)$ and a standard deviation (F_1,F_2) for each normal distribution, as well as, the mixture proportion (8; note many authors use B to denote this parameter, which can be confused with the constant B=3.14, while others use α, which can be confused with Type 1 Error Rate):

$$B(\mu_1,\sigma_1,\mu_2,\sigma_2,\pi) = \lambda*\frac{1}{\sqrt{2\pi\sigma_1^2}}e^{\frac{-(Y_i-\mu_1)^2}{2\sigma_1^2}} + (1-\lambda)*\frac{1}{\sqrt{2\pi\sigma_2^2}}e^{\frac{-(Y_i-\mu_2)^2}{2\sigma_2^2}} \tag{2}$$

$$= \lambda*N_1(\mu_1,\sigma_1) + (1-\lambda)*N_2(\mu_2,\sigma_2)$$

The loss function assesses the deviation between the data and the population. Although the *cnlr* routine allows one to define the error (or loss) function, the default of least-squares error was used, which is the same as that which is used in ordinary least-squares regression (OLS).

The fitting algorithm returns the best-fit parameters, given the data and the error function. In *SPSS*, there are two possible non-linear regression routines. The *nlr* (non-linear regression) procedure uses the Levenberg-Marguart algorithm for fitting the data to the function, while the *cnlr* (constrained non-linear regression) uses a sequential quadratic search algorithm that permits constraints to be placed on the parameters. The basic Marquart algorithm is arguably the most general fitting routine (Bevington & Robinson, 1992 p. 164), but the ability of *cnlr* to place bounds on the values provides some additional stability (this is more important within Monte Carlo simulations), so the results presented here cite only the *cnlr* approach.

Application to a Single Set of Data
The fitting of two different data sets — one

unimodal and one bimodal — is demonstrated. Both data sets consisted of data generated by *SPSS*. Hereafter, the data from the unimodal normal distribution will be labeled as unimodal data, while data generated from a mixture of two normal distributions will be labeled as bimodal data. Similarly, the function defining the unimodal normal distribution will be labeled as the unimodal function while the function defining the mixture or two normal distributions will be labeled as the bimodal function.

Unimodal Data
A unimodal data set of 500 data points were obtained from a unimodal function (a normal distribution) defined as $N(:,F) = N(0,1)$ using the *SPSS* command *NOR-MAL*, which generates standard Normal pseudo-random variates. The resulting distribution is shown in Figure 1, and with empirical mean 0.001 and standard deviation 0.967 (median: 0.003). The process of binning produced a range of 79 bins for 500 data points with 12 bins per standard deviation, but in fact, only 64 bins contained non-zero counts (see Figure 1): With 500 data points, one can

---

Listing 1
Fitting Algorithm for the Bimodal Function

```
compute          prop = observed/total.

model program    mean=0.0 sd=1.0 c=0.0.
compute          xa = abs(xl-xc).
compute          xb = abs(xu-xc).
compute          h1 = (.398942/sd ) * exp( -(((xl-mean)**2) / (2*sd**2)) ).
compute          h2 = (.398942/sd ) * exp( -(((xc-mean)**2) / (2*sd**2)) ).
compute          h3 = (.398942/sd ) * exp( -(((xu-mean)**2) / (2*sd**2)) ).
compute          preduni =  (.5*(h1+h2)*xa + .5*(h2+h3)*xb) + c.
cnlr             prop
                 /bounds  sd gt 0.0001
                 /pred = preduni
                 /save = preduni residuni.
```

expect to obtain a few high z-scores, and consequently, bins in the tails that contain no counts. The algorithm only considers bins with non-zero counts: More will be said on this point at the end of this paper. The binned mean, 0.001, and standard deviation, 0.967 (median: 0.001) were not different, providing a rudimentary check on the binning process. Such a check is more important when there are fewer data points.
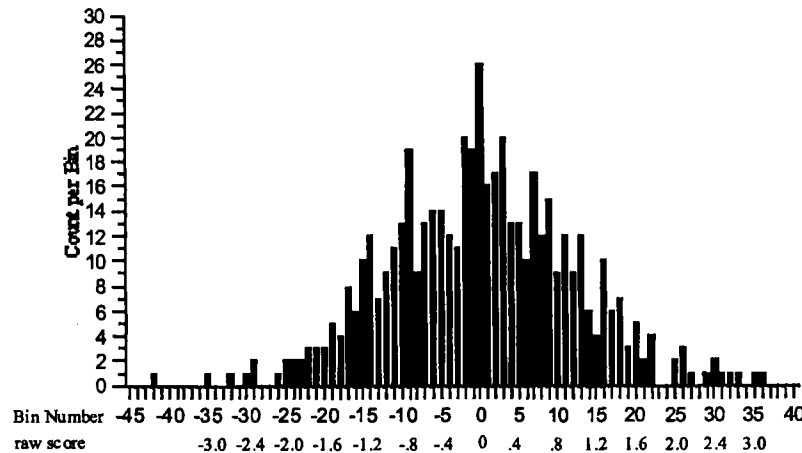

Figure 1 The unimodal distribution.

The first compute command simple converts the observed counts per bin into a proportion within each bin (percentages could also be used and would provide more significant digits in the output on some statistics, but percentages would require adjustments to the formula for the normal distribution).

All the compute statements between the model program and cnlr commands are setting up the fitting algorithm: All of these could be collapsed into one line, but it would not be as readable. Essentially, as shown in Figure 2, the area defined by each bin (i.e., the integration) under the normal curve is computed: This area is the probability or proportion of scores that should fall into that bin. There are numerous ways to obtain this area, but in this case, it is computed using the trapezoidal rule for numerical integration: In this case, the proportion is computed from the sum of two trapezoid rule integrations, one for the area from the bin lower limit to the bin centre and one for the bin centre to the bin upper limit (hence, the $h1$, $h2$, and $h3$ of the compute statements refer to the heights of the curve at the lower limit, centre and upper limit of the bin). The use of two trapezoids should simply provide greater resolution, particularly if a bin should happen to cover a large range (in units of standard deviations). It should be noted that, in the end, what really matters is the prediction per bin (the variable preduni) — how one generates the prediction is flexible. In fact, the use of a single trapezoid did not alter the results dramatically.

A constant term, c, is included in the model to insure that the residuals sum to zero: It has minimal effect on the parameters returned or the fit. More is said on the rationale for its inclusion in a forthcoming paper.

It is the cnlr command that actually runs the analysis, using the previous information, subject to the constraints that follow (/bounds, /pred, and /save). Basically, cnlr tries to match the actual data (the variable prop) to the predictions of the model (the variable preduni) by adjusting the values of mean and sd (and c). It is the subcommand /pred that creates and names the predicted variable. The adjustment is performed iteratively — small changes are made to all variables, the fit is computed, and if the fit does not improve "substantially" with the new values, the processes stops.

Because the process is iterative, the program needs a starting value for all of the parameters that are to be adjusted. The model program command identifies the parameters to be adjusted and set their initial (starting) values. It is important to have good starting values: Poor starting values may result in no solution, or worse, in a "second best" solution that can masquerade as the correct solution. In a similar fashion, the subcommand /bounds allows one to set limits on the values of the fitted parameters. This can prevent the algorithm from drifting into a local solution, producing ridiculous values. In this case, since variances can never be less than zero, zero is used as a lower bound.

Finally, the subcommand /save preduni residuni saves and names the predicted scores and the residual scores onto the original data file. It is the residuals that are the key to the algorithm. The algorithm works by trying to minimize the residual. In fact, the algorithm minimizes:
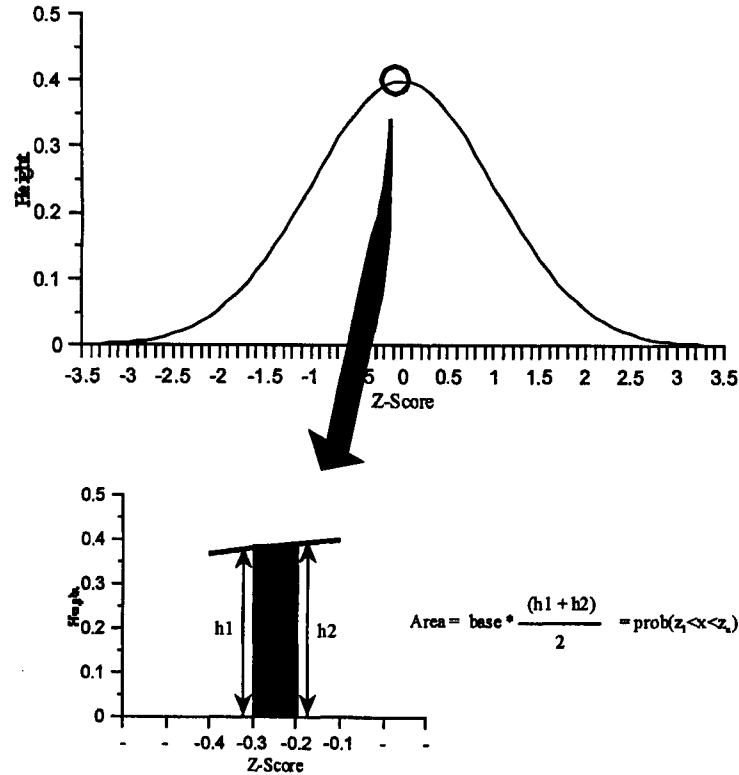
Figure 2 The trapezoid rule for obtaining the area under the curve (numerical integration). The accuracy of the integration depends on the magnitude of the base and the shape of the curve.

$$error = \Sigma\left(Y_{data} - Y_{predicted}\right)^2 = \Sigma\left(Y_i - \hat{Y}\right)^2 = \Sigma residual^2$$

which is essentially the same error term as is used in any OLS regression. The only real difference (in this context) between the more common linear regression (the regression command in SPSS), and non-linear regression ($cnlr$) is that non-linear regression uses an iterative (trial and error) process.

In the case of the current unimodal data set, the algorithm returned a mean of -0.008 ± 0.044, a standard deviation of 0.956 ± 0.037 and a constant of 0.000 ± 0.001 (cited errors are the standard errors of fit, analogous to the standard error on a slope in linear regression) These values, within the defined confidence intervals, are the same as the original measures.

In linear regression, one maximizes the fit between the predicted and actual score by minimizing the sum of the squared residuals. The measure of fit can be defined by numerous statistics, with s^2_{error}, $R^2_{Y.1}$ and P^2_{df} being the most common. Similar measures can be used here and, in fact, the $cnlr$ and nlr algorithms actually provide SS_{error}

(hence, s^2_{error}) and $R^2_{Y.1}$ directly.

SS_{error} can be used on its own as an estimate of the degree of fit (Equation 3). Ideally, SS_{error} should be small, but beyond that, one cannot say much about the predicted value of SS_{error}: It depends on the underlying distribution, and scale of the data.

Another measure of fit is $R^2_{Y.1}$, which is the correlation between the predicted proportion, or count, per bin ($\hat{1}_i$) and actual proportion, or count, per bin (Y_i). It is completely analogous to $R^2_{Y.1}$ in the more common linear regression, and it is essentially:

$$R^2_{Y.\hat{Y}} = 1 - \frac{SS_{error}}{SS_{total}}$$

$$where \quad SS_{error} = \Sigma\left(Y_i - \hat{Y}_i\right)^2$$

$$SS_{total} = \Sigma\left(Y_i - \bar{Y}\right)^2 \tag{4}$$

$\bar{Y}_i$ = the mean proportion

Because R^2 is just a correlation, it can be tested for significance, using the traditional Type I error rate of $\alpha = 0.05$. However, little faith should be placed in "significance" since, in principle, one would not be testing a model that did not have some hope of fitting the data: It is the magnitude of R^2 that matters, particularly, the relative magnitude when comparing models. As shown in Figure 3, what is actually tested by R^2 is the nature of the relationship between the predicted bin proportion (or count) and the actual bin proportion (or count). For R^2 to apply, this relationship should be linear. Given that R^2 is affected by

outliers, one should carefully check the tails of the distribution (theoretical and/or actual) for such points. In addition, non-linearities in the plot imply that the model is not a good match to the data, regardless of the magnitude of the R^2. In this case, the plot seems reasonable. Note that the standard residuals plots will provide the same information.

A third traditional test of the fit of a sample to a theoretical function is that of the χ^2_{df} distribution, which uses actual counts (not proportions):
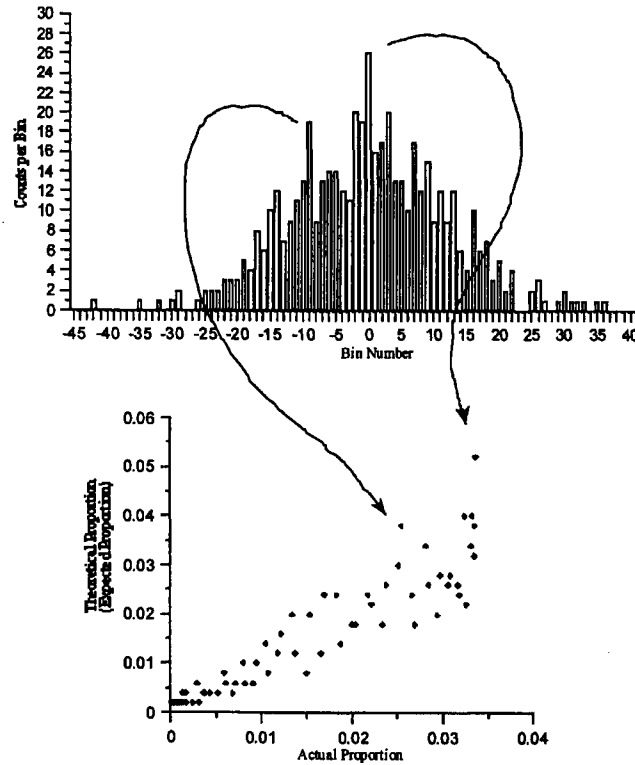

Figure 3 The test provided by R^2 compares theoretical bin counts (proportions) to actual bin counts (proportions). Note that the relationship is relatively linear, but that the variance is not perfectly constant.

$$\chi^2_{df} = \frac{\Sigma\left(C_{obs} - C_{pred}\right)^2}{C_{pred}}$$

(5)

where C_{obs} = the observed counts in each bin

C_{pred} = the predicted counts in each bin

df = #bins - 2 since 2 parameters are estimated

Essentially, if the sample is a good fit to the function, then the value of the χ^2_{df} should be small. Traditionally, one uses a type 1 error rate of $\alpha = 0.05$ that assumes that if the χ^2_{df} is one of those values that falls into the upper 5% of the distribution, then the theoretically-derived distribution is not a good match to the data.

For the current data, $SS_{error} = 0.0015$ and $R^2_{Y\hat{Y}} = 0.848$ with an associated $F(1,62) = 346.429$ ($p<.000$). Both SS_{error} and $R^2_{Y\hat{Y}}$ are directly available within the output of the *cnlr* (or *nlr*). The value of χ^2 must be computed separately using the actual bin score and the saved predicted values. Here, $\chi^2_{62} = 76.118$ ($p>.100$) meaning that the model does not deviate significantly from the data.

There is one crucial point that has been mentioned previously, is that in this fitting, bins with zero counts (see Figure 1) were not included in the analysis. The results might be different if the zero-count bins (particularly zero-count bins flanked by non-zero-count bins) were included. This is an issue that will be returned to in a forthcoming paper.

Bimodal Data

A bimodal data set consisting of 500 data points

was generated from a bimodal function (a mixture of two normal distributions) defined as $B(\mu_1,\sigma_1^2,\mu_2,\sigma_2^2,\lambda) = B(-1.0,1.0,1.0,1.0,0.5)$. Note that the population means were placed symmetrically around 0.0, that the variances were equal and that the mixture proportion was 0.50. Hence, by design, the mean of the distribution should be zero (a quick, but useful, check). The resulting distribution is shown in Figure 4, had a mean of 0.063 and standard deviation of 1.44 before binning (median: 0.091) and a mean of 0.064 and a standard deviation of 1.44 after binning (median: 0.063). The process of binning produced a range of 75 bins but only 58 contained non-zero counts (see Figure 4).

This data was then fit to the bimodal function (the mixture of two normal distributions). In this case, there are five parameters that are free to vary: two means (μ_1,μ_2), two variances (σ_1^2,σ_2^2), and the mixture proportion (λ) as well as a constant. The algorithm is more complicated, but it is essentially the same as before. Note that the fitted statistics should reproduce the original population, with sampling error, as well as, drift due to binning. The fitting algorithm is shown in Listing 2.
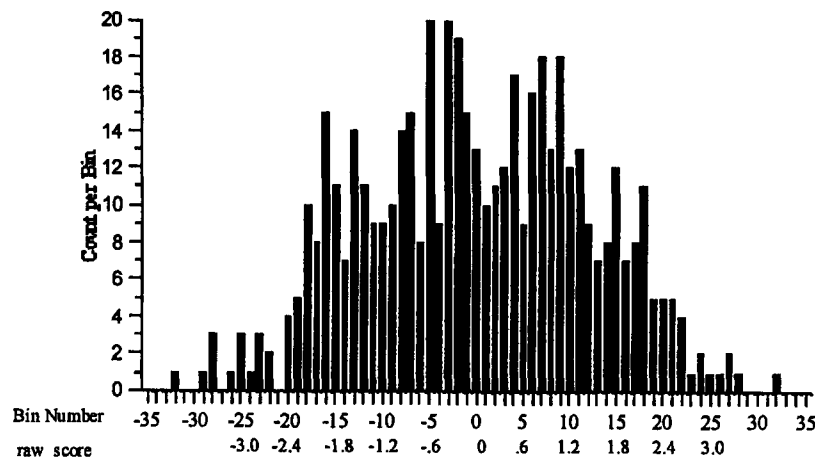

Figure 4 The bimodal distribution.

Listing 2
Fitting Algorithm for the Bimodal Function

```
compute          prop = observed/total * 100.
model program    mean1=-2.0 mean2=1.0 sd1=2.0 sd2=1.0 ratio=0.5 c=0.
compute          xa = abs(xl - xc).
compute          xb = abs(xu - xc).
compute          h1 = (.398942/sd1)*exp(_(((xl-mean1)**2) /(2*sd1**2)) ).
compute          h2 = (.398942/sd1)*exp(_(((xc-mean1)**2) /(2*sd1**2)) ).
compute          h3 = (.398942/sd1)*exp(_(((xu-mean1)**2) /(2*sd1**2)) ).
compute          h4 = (.398942/sd2)*exp(_(((xl-mean2)**2) /(2*sd2**2)) ).
```

compute   h5 = (.398942/sd2)*exp(-(((xc-mean2)**2) /(2*sd2**2))).
compute   h6 = (.398942/sd2)*exp(-(((xu-mean2)**2) /(2*sd2**2))).
compute   predbi = ratio *(.5*(h1+h2)*xa + .5*(h2+h3)*xb)
                       + (1-ratio)*(.5*(h4+h5)*xa + .5*(h5+h6)*xb) + c.
cnlr   prop /pred = predbi
               /bounds sd1 gt 0.0001;
                      sd2 gt 0.0001;
                      1.0 ge ratio ge 0.0
               /save = predbi residbi.

As before, the first *compute* command simply converts the observed counts per bin into a proportion. The *compute* commands between the *model program* and *cnlr* commands create the model. Again, in principle this could be done in one line. The *model program* command defines the parameters to be fitted and sets their initial values. The *cnlr* command defines the dependent variable, sets some bounds on the parameters and sets the routine to save the predicted and residual. The difference between this and the previous unimodal case is one of increasing complexity: There are no substantive changes.

In the case of the current bimodal data set, the values returned by the algorithm were -1.140 ± 0.828 for first mean, 1.047 ± 0.411 for the first standard deviation, 0.967 ± 0.746 for the second mean, 1.049 ± 0.366 for the second standard deviation, 0.473 ± 0.406 for the mixing proportion, and 0.000 ± 0.001 for the constant (cited errors are the standard errors of fit analogous to the standard error on a slope in linear regression). The algorithm returns the original population parameters (within the defined confidence limits).

For the bimodal data, SS_{error} = 0.0018 and $R^2_{Y \cdot \hat{Y}}$ = 0.751 with an associated significance of $F(1,53)$=168.928 (p<.000). χ^2_{53} = 46.277 (p>.950), which implies that the model does not differ significantly from the data. Again, it must be emphasized that bins with zero counts (see Figure 4) were not included in the computations and, as such, result might be different if they were.

Cross Fits

In practice, one would not know whether the unimodal or bimodal function provided the proper fit to the data. To address this question, it is necessary to fit the data with both functions so that the best choice may be obvious. In an ideal world, the unimodal data will fit a unimodal function better than a bimodal function, and the bimodal data will fit the bimodal function better than the unimodal function. Hence, the previous unimodal data was fitted to the bimodal function and the previous bimodal data was fitted to the unimodal function.

The fitting of the unimodal data to the bimodal function produced -0.016 ± 0.091 for first mean, 0.911 ± 0.149 for the first standard deviation, 4.912 ±22.797 for the second mean, 2.771 ± 54.287 for the second standard deviation, and 0.927 ± 0.362 for the mixing proportion. The constant was near zero, 0.001 ± 0.003. The bimodal fit suggests that the data can be modeled by the proper parent distribution centered at 0 and a wider distribution centered five standard deviations above the mean. The wider distribution provides only 7% of the data. Effectively, the bimodal function finds a unimodal distribution.

The fitting of the bimodal data to the unimodal function produced a mean of -0.009 ± 0.097 and a standard deviation of 1.620 ± 0.083, with a constant near zero, 0.001 ± 0.001. These values were within expectation given that $\mu_{bimodal} = \lambda * \mu_1 + (1 - \lambda) * \mu_2 = 0.00$ and $\sigma^2_{bimodal} = \lambda * \sigma_1^2 + (1 - \lambda) * \sigma_2^2 + \lambda * (1 - \lambda) * (\mu_1 - \mu_2)^2 = 2.00$. Hence, fitting the bimodal data with a unimodal function finds a single distribution centered at zero, which is

Table 1
Fitting the unimodal and bimodal data with the unimodal and bimodal functions.

Function	Unimodal Data Unimodal	Bimodal	Bimodal Data Unimodal	Bimodal
SS_{error}	0.0015	0.0015	0.002	0.0018
R^2	0.848	0.849	0.729	0.751
$F(R^2)$	346.429	348.975	151.615	168.928
χ^2	$76.118_{df=62}$	$51.777_{df=59}$	$52.315_{df=56}$	$46.277_{df=53}$

reasonable given that the data were symmetric about zero.

In many cases, the actual parameters produced are less important than the fits of the data to the alternative functions. Table 1 provides SS_{error}, $R^2_{Y.\hat{Y}}$ and χ^2_{df} for the unimodal data fitted to both functions and for the bimodal data fitted to both functions. In the case of the unimodal data, both functions seem to fit equally well. As such, there would be no grounds for arguing that the function is bimodal (parsimony). In the case of the bimodal data, the bimodal function provides the better fit.

Conclusion

This work has been a demonstration of the application of commonly available statistical software, in this case *SPSS*, to solving the problem of assessing putative mixture distributions: particularly, decisions concerning a unimodal normal distribution or a bimodal mixture of two normal distributions. Routines were developed to enable anyone to determine the best-fit statistics for fitting data to a unimodal normal distribution or a bimodal mixture of two normal distribution.

The method presented has several advantages. The most important advantage is that the method is not limited to normal distributions or mixtures of two normal distributions. There is unlimited flexibility in the choice of fitting functions. In fact, the fitting function does not have be continuous (however, non-linear regression is best done with continuous functions). As noted previously, one only needs to be able to create a predicted bin count that can be compared to the actual bin count. The predicted bin count can be based on any function. The likelihood test to be presented later also has this advantage.

The second advantage is that one can modify the algorithm to obtain greater accuracy. Non-linear regression using a least-squares error term assumes that the error is a constant for all values of the independent variable. If bins have variable counts (or proportions), the error per bin is not constant. This could be a factor in the fit if the range of counts is large. *SPSS* non-linear regression allows one to specify the error term. As such, one could switch to weighted least-squares (non-linear) regression if greater accuracy is demanded. This change would also help to alleviate the problem of bins with low counts. It has not been presented in this work because it adds an additional layer of complication thereby obscuring the main points.

The third advantage is related to the first two. In particular situations, some parameters can be fixed. For example, the two variances may set to be equal, or either variance could be set to a constant (e.g., 1.0) or the mixture proportion may be fixed (e.g., 0.5). This would effectively improve the fit for the remaining parameters (assuming that the alterations were appropriate to the theoretical distribution) because it would increase the df per

parameter. Such an approach might be more useful when there are only a few data points (or bins). In a similar fashion, the regression model has the flexibility to include a constant, or to not include a constant. If there is no constant, the model assumes that the regression line goes through the point $(X,\hat{Y}) = (0,0)$. If the model does not include a constant when it should, then the resulting fit will be biased (i.e., the sum of the residuals will not be zero). In this work, it is arguable that the model does not need a constant because the $(0,0)$ point can be assumed (i.e., the predicted value should be zero, when the bin is empty). The caveat is that this is not strictly true when dealing with binned data. That is, bins with counts of zero may correspond to a non-zero theoretical probability. In fact, given a normal parent population, no bin should ever have a zero theoretical probability. Hence, an actual bin count of zero will correspond to some non-zero theoretical bin count (this will be small, but non-zero). As such the inclusion of a constant is prudent. In general, we expect that only χ^2 will be dramatically affected by the removal of the constant(χ^2 can be very large when there is no constant, implying that the model would be biased without the constant).

In summary, the routine works; however, it must be cautioned that this algorithm is only considered an interim solution to the problem — one of many (cf., Eriksen & Eriksen, 1972; Eriksen & Yeh, 1985; Hartigan, 1974; Jones & McLachlen, 1990; Müller & Sawitzki, 1991; Roeder, 1990, 1994; Yantis, Meyer and Smith, 1991; Yellott, 1971). Hopefully, a proper fully parametric method for assessing bimodality will be developed, one that extracts all the information contained within each individual data point rather than working through the intermediary of a histogram. As has been shown, if not done carefully, the process of binning can alter the data. On the other hand, even if a proper parametric method is developed, it will necessarily be tied to particular parent distributions. As such, the algorithms developed herein will continue to serve some purpose with other parent distributions. We have ongoing research studying the methods presented herein in simulations. Forthcoming papers will discuss matters of fit indices and bin size.

References

Bevington, P. R., & Robinson, D. K. (1992). *Data Reduction and Error Analysis for the Physical Sciences*. Toronto: McGraw Hill, Inc.

Do, K., & McLachlan, G. L. (1984). Estimation of mixing proportions: A case study. *Applied Statistics, 33 (2)*, 134-140.

Eriksen, C. W., & Yeh, Y. Y. (1985). Allocation of attention in the visual field. *Journal of Experimental Psychology: Human Perception and Performance, 11 (5)*, 583-597.

Eriksen, C. W., & Eriksen B. A. (1972). Visual backward masking as measured by voice reaction time. *Perception and Psychophysics, 12 (1)*, 5-8.

Hartigan, J. A. (1974). Asymptotic distributions for clustering criteria. *The Annals of Statistics, 6 (1)*, 117-131.

Hayes, W. L. (1994). *Statistics*. NY: Holt, Rinehart and Winston.

Hoffman, J. P., & Miller, A. S. (1998). Denominational influences on socially divisive issues: Polarization or continuity? *Journal for the Scientific Study of Religion, 37 (3)*, 528-546.

Howell, D. (1995). *Statistical Methods for Psychology*. NY: Duxbury Press.

Johnson, D. N., & Yantis, S. (1995). Allocating visual attention: Tests of a two-process model. *Journal of Experimental Psychology: Human Perception & Performance*, 21 (6), 376-1390.

Jones, P. N., & McLachlan, G. J. (1990). Algorithm AS 254: Maximum likelihood estimation from grouped and truncated data with finite mixture models. *Applied Statistics, 39 (2)*, 273-312.

Knoll, J. L. IV, Garver, D. L., Ramberg, J. E., Kingsbury, S. J., Croissant, D., & McDermott, B. (1998). Heterogeneity of the psychoses: Is there a neurodegenerative psychosis? *Schizophrenia Bulletin, 24 (3)*, 365-379.

Müller, D. W., & Sawitzki, G. (1991). Excess mass estimates and tests for multimodality. *Journal of the American Statistical Association, 86*, 738-746.

Ottong, S. E., & Garver, D. L. (1997). A bimodal distribution of plasma HVA/MHPG in the psychoses. *Psychiatry Research, 69*, 97-103.

Reischies, F. M.., Schaub, R. T., & Schlattmann, P. (1996). Normal ageing, impaired cognitive functioning and senile dementia: A mixture distribution analysis. *Psychological Medicine, 26 (4)*, 785-790.

Roeder, K. (1990). Density estimation with confidence set exemplified by superclusters and voids in the galaxies. *Journal of the American Statistical Association, 85*, 617-624.

Roeder, K. (1994). A graphical technique for determining the number of components in a mixture of normals. *Journal of the American Statistical Association, 89*, 487-495.

Sussman, H. M. (1999), A neural mapping hypothesis to explain why velar stops have an allophonic split. *Brain and Language, 70 (2)*, 294-304.

Volbrecht, V. J., Nerger, J. L., & Harlow, C. E. (1997). The bimodality of unique green revisited. *Vision Research, 37 (4)*, 407-416.

Yantis, S., Meyer, D. E., & Smith, J. E. K. (1991). Analysis of multimodal mixture distributions: New tests for stochastic models of cognition and action. *Psychological Bulletin, 110*, 350-374.

Yellott, J. I. (1971) Correction for fast guessing and the speed-accuracy tradeoff in choice reaction time. *Journal of Mathematical Psychology, 8 (2)*, 159-199.

End Notes

1   The normal distribution can be considered a reasonable approximation by virtue of the central limit theorem. In addition, the limiting cases of the binomial, χ^2, Poisson, t and their derivatives are approximately normal.

2   Those who think this is not a serious concern should contemplate the standard use of ANOVA: Many psychological variables, when used as independent factors, have been forcefully cast as categorical in order to fit the analysis (e.g., clinical categories such as depression, drug levels); it is now difficult to present them in their true continuous form.

3   The determination of "some reasonable number" depends on the author: Ten for Bevington and Robinson (1992, p. 109), five to ten for Hayes (1994, p. 862), and five for Howell (1995, p. 115, 141). Values from Hayes and from Howell assume a χ^2 distribution within each cell, while those of Bevington and Robinson assume a Poisson distribution within each cell.