An **R** Package for Assessing Drug Synergism/Antagonism

John C. Boik
New Earth BioMed

Balasubramanian Narasimhan
Stanford University

Abstract

Synergistic and antagonistic drug interactions are important to consider when developing mixtures of anticancer or other types of drugs. Boik, Newman, and Boik (2008) proposed the MixLow method as an alternative to the Median-Effect method of Chou and Talalay (1984) for estimating drug interaction indices. One advantage of the MixLow method is that the nonlinear mixed-effects model used to estimate parameters of concentration-response curves can provide more accurate parameter estimates than the log linearization and least-squares analysis used in the Median-Effect method. This paper introduces the **mixlow** package in R, an implementation of the MixLow method. Results are reported for a small simulation study.

Keywords: synergism, antagonism, Loewe index, Mixlow method, cytotoxicity, R.

1. Introduction

In many branches of medicine, drugs are administered in the form of mixtures. In order to develop new and more effective mixtures it is useful to analytically assess, with respect to the mixture's intended effect, the interactions that occur within a mixture. Such drug interactions can be of an additive nature, or can be antagonistic or synergistic (sub-additive or supra-additive, respectively). The approaches used to assess drug interactions vary depending on the design of the experiment, the expected shape of the concentration-response curve, and other factors. For reviews see Berenbaum (1989); Greco, Bravo, and Parsons (1995); Merlin (1994); Tallarida (2001). The approach described here is applicable to the common in-vitro situation where within-unit and between-unit measurements are available, responses follow a sigmoidal pattern, and ratios between drugs in a mixture are fixed (that is, various dilutions of the mixture and its component drugs are tested). While such data could be generated in many types of experiments, the data discussed in this paper are obtained from in-vitro cytotoxicity

experiments, where cancer cells are exposed to a drug for a specified length of time (typically 72 hours) and then cell viability is indirectly measured, usually via fluorescence readings after addition of a suitable dye. Such cytotoxicity assays use multi-well incubation trays, where each tray receives one drug or mixture, each column of the tray might receive a different drug concentration, and replicate trays are tested for each drug and mixture. Responses are typically modeled as a sigmoidal function of the drug concentration.

Several indices for assessing drug interactions have been proposed, but perhaps the most popular one is the Loewe index. The Loewe index for a mixture is a function of the mixture's concentration-response curve as well as the curves of its component drugs. Therefore, estimating the parameters of these curves is a first step in estimating the index. A common method to estimate both the curve parameters and the index is the Median-Effect method developed by Chou and Talalay (1984). This method linearizes the sigmoidal response curve and then estimates concentration-response parameters using ordinary least squares regression. The interaction index is estimated based on regression results. The Median-Effect method has been criticized on several points, including the linearization step and the need for extensive preprocessing of data Greco *et al.* (1995). Recently, Boik *et al.* (2008) proposed the MixLow method, which is an improvement over the Median-Effect method. The MixLow method utilizes a nonlinear mixed-effects model to more accurately estimate concentration-response parameters. In addition, data preprocessing is minimal.

The three components of the MixLow method are a nonlinear mixed-effects model, the Loewe index, and a method to calculate confidence intervals for the index. See Boik *et al.* (2008) for complete details. In some applications it is not necessary to estimate a Loewe index and only curve parameter estimates are desired (IC50 values, for example). In such cases, estimation of the Loewe index can be omitted. Note that the R package **drm** by Ritz and Streibig (2005) also analyzes concentration-response data, but does not use a mixed-effects model to do so.

Concentration-effect data are modeled using a sigmoidal function. The utility of the sigmoidal function stems from its simplicity (as few as two parameters) and its empirical usefulness in modelling cytotoxicity data. For these reasons, sigmoidal functions are often used for modelling cytotoxicity data. The MixLow method is novel in its use of a nonlinear mixed-effects model for estimating sigmoidal curve parameters from concentration-response data, for the purpose of estimating drug interaction indices and associated confidence intervals.

This paper describes the **mixlow** package, an implementation of the MixLow method in R (R Development Core Team 2009). An overview of the mathematical model is given, components of the package are summarized, and results from a small simulation study are reported. The package is available from the Comprehensive R Archive Network at `http://CRAN.R-project.org/package=mixlow`.

2. MixLow method

Let the random variable fa signify the fraction of cells affected by a drug concentration, and define $\phi = \mathrm{E}[fa]$. In some contexts, ϕ is estimated based on the data and in other contexts a concentration is estimated that produces a fixed value of ϕ. Denote the ϕ-effective log concentration of drug d by $\psi_{d,\phi}$. This is the concentration that produces a fraction affected equal to a fixed ϕ. For example, $\exp(\psi_{d,0.2})$ is the concentration that inhibits proliferation of a population by 20 percent. By convention, this is referred to as the IC20.

2.1. Basic MixLow model

The MixLow method utilizes a nonlinear mixed-effects model to estimate parameters of the sigmoidal concentration-response curve. Responses $\{Y_{d,t,w}\}$ are modeled as:

$$Y_{d,t,w} = \exp(\mu + b_t)\,(1 - \phi_{d,w,t}) + \epsilon_{d,t,w}\,, \tag{1}$$

where

$$\phi_{d,t,w} = 1 - \frac{1}{1 + \left(\frac{\exp(c_{d,t,w})}{\exp(\psi_{d,0.5})}\right)^{\gamma_d}}\,, \tag{2}$$

and the subscripts d, t, w refer to the d-th drug, t-th tray, and w-th well. In addition, γ_d indexes the steepness of the curve for drug d at the IC50, and $c_{d,t,w}$ is the log concentration of drug d in well w and tray t. Here drug d could refer to a specific mixture or could refer to a component drug. Model (1) can be used to estimate parameters for single drugs or a mixture, or can be used to estimate parameters simultaneously for all component drugs and the mixture. For treatment control wells (those that receive media, cells, and a drug concentration of zero), $\phi_{d,t,w} = 0$. For other wells, $0 < \phi_{d,t,w} < 1$. The expected value of $\exp(\mu + b_t)$ refers to the expected mean response of control wells over all trays assessed, where b_t is a random deviate specific to tray t (i.e., a "tray" effect). Values $\{b_t\}$ are independently distributed as $b_t \sim \mathrm{N}(0, \sigma_b^2)$. The error terms are independently distributed as $\epsilon_{d,t,w} \sim \mathrm{N}(0, f(\cdot))$, where $f(\cdot)$ is a function discussed later.

The Loewe index for a mixture of n drugs is given by:

$$L_\phi = \sum_{d=1}^{n} \frac{\exp(m_{d,\phi})}{\exp(\psi_{d,\phi})} = \sum_{d=1}^{n} \tau_d \frac{\exp(\psi_{m,\phi})}{\exp(\psi_{d,\phi})}\,, \tag{3}$$

where $m_{d,\phi}$ is an unknown constant signifying the log concentration of drug d in the mixture when the mixture is at its ϕ-effective log concentration, τ_d is the fraction of the mixture composed of drug d, and $\psi_{m,\phi}$ is the ϕ-effective log concentration of the mixture. The mixture is synergistic if $L_\phi < 1$, it is additive if $L_\phi = 1$, and it is antagonistic if $L_\phi > 1$. Estimate $\hat{L}_\phi$ is obtained by using estimates $\hat{\psi}_{m,\phi}$ and $\hat{\psi}_{d,\phi}$ in Equation (3). To obtain expressions for $\hat{\psi}_{m,\phi}$ and $\hat{\psi}_{d,\phi}$, note that if $c_{d,t,w}$ in (2) is equated to $\hat{\psi}_{d,\phi}$, then $\hat{\phi}_{d,t,w}$ becomes ϕ. That is,

$$\phi = 1 - \frac{1}{1 + \left(\frac{\exp(\hat{\psi}_{d,\phi})}{\exp(\hat{\psi}_{d,0.5})}\right)^{\hat{\gamma}_d}}\,. \tag{4}$$

To write the Loewe index as an explicit function of $\hat{\psi}_{d,\phi}$, first solve (4) for $\hat{\psi}_{d,\phi}$ to obtain

$$\hat{\psi}_{d,\phi} = \log\left(\left(\frac{\phi}{1 - \phi}\right)^{\frac{1}{\hat{\gamma}_d}}\right) + \hat{\psi}_{d,0.5}\,. \tag{5}$$

Using (3) and (5), the estimator of the index is given by

$$\hat{L}_\phi = \sum_{d=1}^{n} \tau_d \frac{\exp\left(\log\left(\left(\frac{\phi}{1-\phi}\right)^{\frac{1}{\hat{\gamma}m}}\right) + \hat{\psi}_{m,0.5}\right)}{\exp\left(\log\left(\left(\frac{\phi}{1-\phi}\right)^{\frac{1}{\hat{\gamma}d}}\right) + \hat{\psi}_{d,0.5}\right)} . \tag{6}$$

Confidence intervals based on $\mathrm{SE}\left(\log(\hat{L}_\phi)\right)$ are obtained using the Delta method as follows:

$$\mathrm{SE}\left(\log(\hat{L}_\phi)\right) \approx \left(\left(\begin{array}{c}\frac{\partial \log(\hat{L}_\phi)}{\partial \hat{\psi}} \\ \frac{\partial \log(\hat{L}_\phi)}{\partial \hat{\gamma}}\end{array}\right)^\top \mathrm{v\hat{a}r}\left(\begin{array}{c}\hat{\psi} \\ \hat{\gamma}\end{array}\right)\left(\begin{array}{c}\frac{\partial \log(\hat{L}_\phi)}{\partial \hat{\psi}} \\ \frac{\partial \log(\hat{L}_\phi)}{\partial \hat{\gamma}}\end{array}\right)\right)^{\frac{1}{2}}, \tag{7}$$

where the superscript $\top$ refers to transpose. The term $\mathrm{v\hat{a}r}\left(\begin{array}{c}\hat{\psi} \\ \hat{\gamma}\end{array}\right)$ is obtained from the observed information matrix, which is produced when Model (1) is fit by maximizing the likelihood function. The confidence interval for $\hat{L}_\phi$ is

$$\exp\left(\log(\hat{L}_\phi) \pm t_{df,1-\frac{\alpha}{2}}\mathrm{SE}\left(\log(\hat{L}_\phi)\right)\right), \tag{8}$$

where $t_{df,1-\frac{\alpha}{2}}$ is a multiplier obtained from the t distribution with df degrees of freedom and α significance level.

Equation (8) is a piecewise confidence interval. In a typical linear regression setting, a simultaneous interval can be constructed using the Working-Hotelling procedure. If directly applied to the Loewe index, the Working-Hotelling interval would be

$$\exp\left(\log(\hat{L}_\phi) \pm \sqrt{(p)F_{.95,p,df}}\mathrm{SE}\left(\log(\hat{L}_\phi)\right)\right), \tag{9}$$

where p is the number of parameters used to construct the interval and df is the degrees of freedom. The interval (9) has unknown properties, however. Model (1) is nonlinear and contains random effects. The standard error is obtained at ϕ using a first-order approximation to a linear model. Nevertheless, we can use the Working-Hotelling procedure to obtain a rough estimate of the difference in widths between the pointwise and continuous intervals. In a typical two-drug experiment, the interval will be based on six parameters (ψ and γ for each drug and the mixture). With three replicate 96-well trays per drug (see discussion below), the degrees of freedom will be roughly 540. Therefore, the multiplier in the continuous 95 percent interval will be larger than that in the piecewise interval by a factor of roughly

$$\frac{\sqrt{(p)F_{0.95,p,df}}}{t_{df,0.975}}, \tag{10}$$

or about 1.81.

2.2. Modified MixLow model

In some data sets, responses follow a sigmoidal curve but the curve appears to have a non-zero asymptote. An example is illustrated in Figure 1. The solid curve in the figure is the sum of
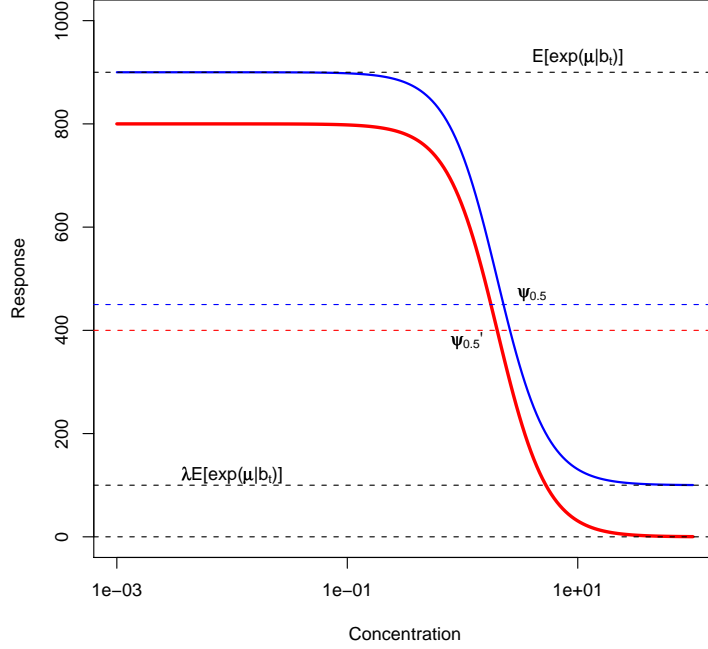
Figure 1: Example concentration-response curve with non-zero asymptote.

the dotted sigmoidal curve and the offset from zero. This type of pattern can be modeled by altering Model (1) to the following:

$$Y_{d,t,w} = (1 - \lambda_d) \exp(\mu + b_t) (1 - \phi'_{d,w,t}) + \lambda_d \exp(\mu + b_t) + \epsilon_{d,t,w}, \tag{11}$$

where

$$\phi'_{d,w,t} = 1 - \frac{1}{1 + \left(\frac{\exp(c_{d,t,w})}{\exp(\psi'_{d,0.5})} \right)^{\gamma'_d}}. \tag{12}$$

Here, $0 \leq \lambda_d \leq 0.5$ is a drug-dependent weight, and ϕ', ψ', and γ' all refer to the sigmoidal curve with zero asymptote. When $\lambda_d = 0$, Model (11) collapses to Model (1). To write the Loewe index as a function of ϕ, $\hat{\psi}'_{d,0.5}$, and $\hat{\gamma}'_d$, an expression for $\hat{\psi}'_{d,\phi}$ is obtained in a manner similar to (5). Specifically,

$$\hat{\psi}_{d,\phi} = \log \left(\left(\frac{\phi}{1 - \phi - \lambda} \right)^{\frac{1}{\hat{\gamma}'_d}} \right) + \hat{\psi}'_{d,0.5}. \tag{13}$$

Using Model (11), and without additional assumptions, the Loewe index can only be estimated for $\phi < 1 - \max_d(\lambda_d)$. This is because $(1 - \phi)$ is not defined for values below $\max_d(\lambda_d)$ for all drugs.

Function	Purpose
`trayData <- readDataFile(filename)`	reads data from file
`mixlowData <- prepareData(trayData)`	prepare raw data
`cellLines <- getCellLines(mixlowData)`	get a vector of cell line names
`drugs <- getDrugs(mixlowData)`	get a vector of drug names
`trays <- getTrays(mixlowData)`	get a vector of tray names
`parameterDefaults <- getNlsParameterDefaults(trays)`	make data frame of parameter default values for NLS
`nlsData <- doNls(mixlowData, parameterDefaults)`	run nonlinear least-squares (NLS) model
`NlmePrintVarFunctions()`	print variance function options for nonlinear mixed-effects (NLME) model
`nlmeData <- doNlme(mixlowData, nlsData)`	run NLME model
`loeweData <- doLoewe(mixlowData, nlmeData)`	run Loewe analysis

Table 1: Functions in the **mixlow** package in order of use.

In Figure 1, the maximal response ($E[\exp(\mu | b_t)]$) is 900, the half-maximal response (at concentration $\psi_{0.5}$) is 450, and the asymptote occurs at a response of 100. The total response (solid curve) is equal to the asymptotic response plus the sigmoidal response (dotted curve). The half-maximal response of the dotted curve is 400, which occurs at a concentration $\psi'_{0.5}$.

3. Summary of the mixlow package

The **mixlow** package contains ten functions, listed in Table 1 and ordered by chronological use. In addition, there are `plot`, `print`, and `summary` methods for objects `trayData`, `mixlowData`, `nlsData`, `nlmeData`, and/or `loeweData`. Detailed information on the functions is available via the help commands in R. A summary of their use and actions is provided below. To use the **mixlow** package, the following six steps are performed in sequence:

1. Design the experiment, collect the data, and prepare the data file.

2. Read the data file using the function `readDataFile`.

3. Prepare and subset the data using the function `prepareData`.

4. Obtain starting values for the fixed-effects parameters of the nonlinear mixed-effects (NLME) model by using nonlinear least squares (NLS) analysis. This is done using the function `doNls`.

5. Use the NLS parameters as starting values for the NLME model. The mixed-effects model is estimated using the function `doNlme`.

6. Use the NLME parameters to estimate the Loewe index. This is done using the function `doLoewe`.

3.1. Experimental design and preparation of the data file

The appropriate experimental design for use with the **mixlow** package is the "ray" design, which has the following two characteristics:

- all drugs are tested individually at a variety of concentrations

- a mixture is made of two or more drugs and this mixture is tested at a variety of concentrations.

As a result, ratios between component drugs of a mixture are constant over all concentrations tested. The MixLow method will not work for checkerboard designs where ratios between drugs in a mixture vary. Each tray should test only one drug or one mixture, and each drug or mixture should be tested in replicate trays.

In a typical synergism experiment, two drugs and their mixture might be tested, although the MixLow model can be used on larger mixtures. Three replicate trays might be used for each drug and the mixture, or nine trays in total. Within each tray, 12 of the wells might be *treatment-control wells* that receive cells and media but no drug, and 60 might be *treatment wells* that receive drug, media, and cells. Ten different concentrations of the drug might be tested in the treatment wells (6 wells per concentration). Lastly, 24 of wells might be *optical control wells* (called *blanks* here for convenience). These can be of two types:

- Type `bbt`: optical control wells contain only media

- Type `bbc`: optical control wells contain media plus drug, and each drug concentration tested in treatment wells is also tested in optical control wells.

The first type is referred to here as `bbt`, or blanks-by-tray, and the second type is referred to as `bbc`, or blanks-by-concentration. The use of blanks-by-concentration is recommended when drugs can induce concentration-dependent responses (e.g., autofluorescence). For blanks-by-tray, treatment- and treatment-control-well responses are adjusted by subtracting the mean response of all `bbt` wells in a tray. For blanks-by-concentration, a 1st to 4th degree polynomial is fit to the `bbc` responses and predicted values for each concentration are subtracted from treatment- and treatment-control-well responses.

The only preprocessing of data used in the Mixlow method is subtraction of an optical correction factor for each well based on `bbt` or `bbc` responses. Relative to the variance of treatment-well responses at each drug concentration, the variance of optical control well responses is typically very small. Therefore, only a few `bbt` wells are needed per tray, or a few `bbc` wells are needed per concentration. Because of the low variance, use of optical correction factors is not likely to introduce significant uncertainty into the model.

In many cytotoxicity experiments, a reasonable arrangement for a 96-well tray is to use the last two rows of every column as `bbc` wells, use the first six rows of the first two columns as treatment-control wells, and use the first six rows of the third to twelfth columns as treatment wells, with each column receiving a different drug concentration. This layout is shown in Table 2. In the table, `rx` signifies treatment wells and `rx*` signifies treatment-control wells. If there is a drug-response pattern in plates that is not related to drug concentration

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12
R1	rx*	rx*	rx	rx	rx	rx	rx	rx	rx	rx	rx	rx
R2	rx*	rx*	rx	rx	rx	rx	rx	rx	rx	rx	rx	rx
R3	rx*	rx*	rx	rx	rx	rx	rx	rx	rx	rx	rx	rx
R4	rx*	rx*	rx	rx	rx	rx	rx	rx	rx	rx	rx	rx
R5	rx*	rx*	rx	rx	rx	rx	rx	rx	rx	rx	rx	rx
R6	rx*	rx*	rx	rx	rx	rx	rx	rx	rx	rx	rx	rx
R7	bbc	bbc	bbc	bbc	bbc	bbc	bbc	bbc	bbc	bbc	bbc	bbc
R8	bbc	bbc	bbc	bbc	bbc	bbc	bbc	bbc	bbc	bbc	bbc	bbc

Table 2: Example layout for 96-well plate.

(e.g., responses in the center wells are different from responses in the peripheral wells), then a randomized concentration assignment should be employed.

To use the `readDataFile` function, data must be stored in a specially formatted ASCII file. An industry standard XML format for cytotoxicity data has not yet been developed, and the interim format described here was designed to facilitate cut-and-paste operations from plate reader data displayed in spreadsheet format. If a XML standard format is developed in the future, then the package can be modified to take advantage of this. Two sample data files are included with the package and can be used as templates for constructing new data files, as well as for testing out the capabilities of the program. One of these files contains simulated data, and one contains data from a synergism experiment described in Boik *et al.* (2008).

The data file can be created in a spreadsheet (in which case it must be saved in tab-delimited plain text format without automatic enclosing quotations). Row labels (the first column in a data file if a spreadsheet is used) are required, and the order of row labels must not change from that given in the sample files. The required layout for the first three columns is shown in Table 3. The first eight row labels (up to and including `col`) occur only once. Other row labels are repeated for each tray. To conserve space, only three rows and two columns are shown for the `conc`, `label`, and `resp` entries. Entries for the following labels are optional, but all row labels, including the `space` labels, must still be present: `global_notes`, `name`, `Institution`, `email`, `assay`, `date`, `drug_name_full`, `seed_density`, `assay_time`, `tray_notes`, `media`, `drug_solvent`, `solvent_drug_ratio`.

The first eight rows contain general information about the experiments and the remainder of the file occurs in blocks, one block for each tray. Entries for `tray_label` and `drug_name_short` should contain only digits, letters, and the underscore character. The `drug_name_short` entry should consist of a unique, abbreviated drug name containing no more than 20 characters. This will be the drug name that appears in plots and in text output. The `composition` entry specifies the fraction of each drug in a mixture. For each drug, a string such as "`drug_A = 0.5`" must be given (for a hypothetical `drug_A`). The values provided should sum to 1.0 over all drugs, so if a single drug is tested in a tray the string would look like "`drug_A = 1`". The `conc` entries specify the drug concentration in each well and the `label` entries specify the type of well (`rx` for treatment, `bbt` or `bbc` for blanks). Treatment-control wells are also given the label `rx`. The `resp` entries specify the responses measured in each well.

The user does not need to manually alter the raw data in any way. The only exception is when there is good reason to believe that data from certain wells in a tray are anomalies. In

Column 1	Column 2	Column 3
`global_notes`	general notes about the experiment	
`name`	users name	
`Institution`	users institution	
`email`	users e-mail	
`assay`	the type of assay used	
`conc_units`	the units for the response	
`rows`	the number rows/tray	
`cols`	the number of columns/tray	
`space`		
`tray_label`	the label for the first tray	
`tray_replicate`	an identifier for the replicate number	
`date`	date for this tray	
`cell_line`	cell line for this tray	
`drug_name_full`	complete drug name	
`drug_name_short`	abbreviated drug name	
`composition`	e.g., `drug_1 = 0.4`	e.g., `drug_2 = 0.6`
`seed_density`	seeding density for well in this tray	
`assay_time`	length of incubation time	
`tray_notes`	any other notes on this tray	
`media`	type of media used	
`drug_solvent`	solvent used (DMSO, for example)	
`solvent_drug_ratio`	ratio of solvent to drug	
`space`		
`conc`	drug conc. in row 1, col 1	drug conc. in row 1, col 2
`conc`	drug conc. in row 2, col 1	drug conc. in row 2, col 2
`conc`	drug conc. in row 3, col 1	drug conc. in row 3, col 2
`label`	label for row 1, col 1	label for row 1, col 2
`label`	label for row 2, col 1	label for row 2, col 2
`label`	label for row 3, col 1	label for row 3, col 2
`resp`	response in row 1, col 1	response in row 1, col 2
`resp`	response in row 2, col 1	response in row 2, col 2
`resp`	response in row 3, col 1	response in row 3, col 2

Table 3: Layout of the data file.

this case the responses for these wells can be replaced by periods ("`.`") in the `resp` entries of the data file, which will cause these wells to be ignored. If desired, the `excludeWells` argument of the `readDataFile` function can be used to exclude a list of wells from all trays. The row and column of each cell to be excluded need to be specified in the excludeWell list.

3.2. Generating starting values for fixed-effects parameters

Once a file is read using the `readDataFile` function, and data are subset and prepared using the `prepareData` function, the next step is to generate rough estimates of concentration-response curve parameters (μ, $\psi_{d,0.5}$, $\log \gamma_d$, and λ_d). These rough estimates will be used

as starting values for fixed effects parameters in the nonlinear mixed-effects model. The estimates are generated using the `doNls` function, which is a wrapper for R's `nls` function. The `doNls` function also produces output that can be plotted, and these plots can be useful for understanding the data and estimating parameter values graphically. An `nls` analysis is performed separately on each tray.

By default, only Model (1) is estimated. If the user provides a starting value for λ_d, then Model (11) will also be estimated. If Model (11) is estimated, the results for Model (1) will be compared to those for Model (11) and the one that produced the lowest BIC score will automatically be selected. If Model (11) is estimated, the user can specify a threshold value (lower bound) for λ_d. If the `nls` function estimates a value for λ_d that is below this threshold, Model (1) will be selected. Final parameter estimates are obtained from the selected model.

The `nls` function requires starting values. As a convenience, the `getNlsParameterDefaults` function can be used to generate a data frame of parameter defaults for each drug. Parameter values of `NaN` indicate the following default values will be used:

- $\psi_{d,0.5}$ will be set at the closest tested log concentration corresponding to $1/2$ the mean of control wells

- std($\psi_{d,0.5}$) will be set at the absolute value of $\psi_{d,0.5}/2$

- log(γ_d) will be set to zero

- std(log(γ_d) will be set to one

- μ will be set to the log of the mean of all control wells

- λ_d will be set to `NaN`, which indicates that a lambda term will not be used. If a lambda term is used, its value should be set between zero and 0.5.

In order to increase the chance of model convergence, the `doNls` function will sample $\psi_{d,0.5}$ and γ_d starting values from a normal distribution with mean and standard deviation equal to the user-supplied or default-generated values. The user can specify the number of random samples (`numRandomTries`) to draw. After fitting, the models are ranked by BIC scores. The starting values for the top few models are averaged and used as starting values for a final model.

If `verbose` is set equal to TRUE, information on the various models constructed will be printed, including a summary table that contains estimates for γ_d, $\psi_{d,0.5}$, μ, and λ_d. These parameters are noted as g*1*, p*1*, u*1*, and *lambda* in the printout, respectively. (Values for λ_d will be included only if a λ-model was estimated). Parameters γ_d, $\psi_{d,0.5}$, and μ are in log scale. The `print` and `summary` methods for the data objects returned from the `prepareData`, `doNls`, `doNlme`, and `doLoewe` functions can also be used to examine the results.

3.3. Running the mixed-effects model

The function `doNlme` reads output from the NLS model and runs the NLME model. The `doNlme` function is a convenience wrapper for the `nlme` function from the `nlme` package Pinheiro, Bates, DebRoy, Sarkar, and R Development Core Team (2009). The user supplies an `analysis` argument, which can be either `single` or `multiple`. If `single`, each drug will be

Variance Function	Analysis Type	$\epsilon_{d,t,w} \sim N(o, \sigma^2_{d,t,w})$, where $\sigma_{d,t,w}$ equals:
1	`multiple`	σ
2	`multiple`	$\sigma\alpha_d$, where parameter α_d is drug dependent
3	`multiple`	$\sigma E[Y_{d,t,w}\,|b_t]$
4	`multiple`	$\sigma E[Y_{d,t,w}\,|b_t]^{\beta_d}$, where parameter β_d is drug dependent
1	`single`	σ
2	`single`	$\sigma E[Y_{d,t,w}\,|b_t]$
3	`single`	$\sigma E[Y_{d,t,w}\,|b_t]^{\beta}$
4	`single`	$\sigma\left(\beta + E[Y_{d,t,w}\,|b_t]\right)$

Table 4: Variance functions for use with the `doNlme` function.

analyzed separately. If `multiple`, all drugs will be analyzed together. The user must also specify the variance function to be used from the list given in Table 4. If analysis is `multiple`, the user can specify more than one variance function in the form of a vector, in which case one model will be estimated for each variance function and the best model will be identified using the BIC criterion. If analysis is `single`, the user must provide a list, named by drug, which specifies the vector of variance functions to be used for each drug. For a given drug, if more than one variance function is specified, one model will be estimated for each variance function and the best model will be identified using the BIC criterion. For heteroscedastic errors, variance function `3` (multiple drugs) or `2` (single drugs) may be appropriate in many cases. Simpler error functions could be tried if these do not allow convergence.

As with the `doNls` function, if the verbose argument is set equal to TRUE, information on the models constructed will be printed. Parameter values and standard errors are printed, along with other information. The `rc` values given at the bottom of the summary for each set of drugs is a goodness-of-fit statistic similar to the r^2-statistic (it is the average model concordance correlation Vonesh, Chinchilli, and Pu 1996).

Lastly, for testing purposes, if only one tray of data is available for a drug, the function will duplicate concentrations and responses in that tray to make two (equivalent) replicates so that analysis can be completed. This can be used, for example, to obtain rough IC50 estimates when only one tray is available.

3.4. Loewe analysis

After parameter estimates for the concentration-response curves have been made by the `doNlme` function, these can be passed to the `doLoewe` function so that Loewe indices can be estimated. The primary result from the Loewe analysis is a data frame of fraction affected values and their corresponding Loewe index estimates. Standard errors and upper and lower confidence limits for the index at each fraction affected value are also produced. By default, fraction affected values range from 0.02 to 0.98 (in steps of 0.01).

The output of the `doLoewe` function also includes a measure of the degree of statistically significant synergism and antagonism at each fraction affected value. For synergism, this value is zero at any fraction affected value where the upper confidence interval limit is greater than one. Otherwise, the value is one minus the upper confidence interval limit. For antagonism, this value is zero at any fraction affected value where the lower confidence interval limit is

less than one. Otherwise, the value is the lower confidence interval limit minus one. These values can be useful for summarizing the index over a range of fraction affected values. For an example application, see Boik and Newman (2008).

3.5. Analysis of example data

The following code illustrates use of **mixlow** functions on a data set included with the package. These data are for vincristine (vin), topotecan (topo), and their 1:1 mixture (mixture) tested against A549 human lung cancer cells. The response curve for vincristine suggests a non-zero asymptote.

1. Read the data file located in the mixlow/inst/exdata folder:

```
R> library(mixlow, lib.loc = "/home/john/MixLow_R/Naras-JB_April2010/lib")
R> dataFile <- "A549_vin_topo_data.txt"
R> trayData <- readDataFile(system.file("exdata", dataFile,
+    package = "mixlow"))
R> trays <- getTrays(data = trayData)
R> drugs <- getDrugs(data = trayData)
R> cellLines <- getCellLines(data = trayData)
```

2. Prepare the data:

```
R> mixlowData <- prepareData(trayData = trayData, trays = trays[1:9],
+      cellLines = cellLines[1])
R> summary(mixlowData)
```

```
 ===================== Drug Ratios =====================


        tray    drug cell Units rows cols mixture topo vin
1 mixture_tr1 mixture A549 ug/ml    8   12       0  0.5 0.5
2 mixture_tr2 mixture A549 ug/ml    8   12       0  0.5 0.5
3 mixture_tr3 mixture A549 ug/ml    8   12       0  0.5 0.5
4    topo_tr1    topo A549 ug/ml    8   12       0  1.0 0.0
5    topo_tr2    topo A549 ug/ml    8   12       0  1.0 0.0
6    topo_tr3    topo A549 ug/ml    8   12       0  1.0 0.0
7     vin_tr1     vin A549 ug/ml    8   12       0  0.0 1.0
8     vin_tr2     vin A549 ug/ml    8   12       0  0.0 1.0
9     vin_tr3     vin A549 ug/ml    8   12       0  0.0 1.0
```

3. Run NLS analysis:

```
R> parameterDefaults <- getNlsParameterDefaults(trays = trays[1:9])
R> parameterDefaults["vin_tr1", "param.lambda"] <- 0.2
R> parameterDefaults["vin_tr2", "param.lambda"] <- 0.2
R> parameterDefaults["vin_tr3", "param.lambda"] <- 0.2
R> nlsData <- doNls(mixlowData = mixlowData, parameterDefaults)
R> summary(nlsData)
```

```
==================== NLS Parameter Estimates ====================

          tray    drug cell            g         p        u    lambda
1     topo_tr1    topo A549 -0.1480674528 -4.124183 7.063048 0.0000000
2      vin_tr1     vin A549  1.3818841128 -4.392709 6.881664 0.3609912
3 mixture_tr1 mixture A549 -0.1955504845 -3.524586 7.246545 0.0000000
4     topo_tr2    topo A549 -0.3452055775 -4.753108 8.919553 0.0000000
5      vin_tr2     vin A549  0.9289258175 -4.398356 8.334986 0.1973117
6 mixture_tr2 mixture A549 -0.2026590099 -4.536981 8.379571 0.0000000
7     topo_tr3    topo A549 -0.0008971358 -4.454763 8.407929 0.0000000
8      vin_tr3     vin A549  0.4025827528 -4.746523 7.966266 0.2241574
9 mixture_tr3 mixture A549 -0.1259721587 -3.649347 8.108534 0.0000000
```

4. Run NLME analysis:

```
R> nlmeData <- doNlme(mixlowData = mixlowData, nlsData = nlsData,
+       analysis = "multiple", varFunction = 4)
R> summary(nlmeData)


 ==================== NLME Models ====================



 -------- NLME Model item 1 --------

Nonlinear mixed-effects model fit by maximum likelihood
  Model: adj_resp ~ ifelse(conc > 0, ((1 - lambda) * exp(u) *
         1/(1 + (exp(log(conc) - p))^exp(g)) + exp(u) * lambda),
         ((1 - lambda) * exp(u) + exp(u) * lambda))
  Data: dat1
  Log-likelihood: -4364.634
  Fixed: as.list(fixedEffects)
      g.drg1       g.drg2       g.drg3       p.drg1       p.drg2       p.drg3
-0.432405263  0.662678782 -0.555394574 -4.668112915 -4.517747276 -4.193264752
           u  lambda.drg1  lambda.drg2  lambda.drg3
 8.045593900 -0.003167944  0.187697005 -0.008620041


Random effects:
 Formula: u ~ 1 | tray
                u Residual
StdDev: 0.5515612 2.321458


Variance function:
 Structure: Power of variance covariate, different strata
 Formula: ~(fitted(.)) | drg
 Parameter estimates:
        1         2         3
```

```
0.7224873 0.6821988 0.7323396
Number of Observations: 648
Number of Groups: 9
```

5. Run Loewe analysis:

```
R> loeweData <- doLoewe(mixlowData, nlmeData)
R> summary(loeweData)


Log Loewe index and confidence intervals:
   Fraction.Affected Log.Index SE.Log.Index  Lower.CI Upper.CI
1               0.02 0.3681126    0.6428543 0.1041667 1.300866
5               0.06 0.4773464    0.5203236 0.1718228 1.326131
9               0.10 0.5510492    0.4602465 0.2231895 1.360527
13              0.14 0.6154864    0.4187673 0.2704438 1.400748
17              0.18 0.6772566    0.3864187 0.3171027 1.446460
21              0.22 0.7393741    0.3595288 0.3649587 1.497907
25              0.26 0.8037164    0.3362802 0.4152500 1.555593
29              0.30 0.8717377    0.3156379 0.4690263 1.620222
33              0.34 0.9447528    0.2969569 0.5273046 1.692680
37              0.38 1.0240801    0.2798100 0.5911542 1.774055
41              0.42 1.1111240    0.2639052 0.6617495 1.865656
45              0.46 1.2074241    0.2490427 0.7403997 1.969035
49              0.50 1.3146638    0.2350945 0.8285461 2.085992
53              0.54 1.4346030    0.2219997 0.9276871 2.218513
57              0.58 1.5688290    0.2097737 1.0391356 2.368531
61              0.62 1.7180496    0.1985378 1.1633616 2.537211
65              0.66 1.8801612    0.1885850 1.2982618 2.722876
69              0.70 2.0447117    0.1805278 1.4344017 2.914697
73              0.74 2.1748904    0.1757187 1.5402012 3.071124
77              0.78 2.1292966    0.1792376 1.4975288 3.027591
```

```
R> loeweData <- doLoewe(mixlowData, nlmeData)
R> plot(loeweData)
```

The `plot` command will produce two graphs (see Figure 2), the first of which shows the estimated Loewe index versus fraction affected values.

NLME models can be compared by comparing their AIC and BIC values. These can be obtained by setting the `verbose` argument to TRUE in the function `doNlme`, or by use of the `summary` method on the object returned by the `doNlme` function. To more formally compare two models, one can use the `anova` function, as demonstrated in the code snippet below.

```
R> nlmeData1 <- doNlme(mixlowData = mixlowData, nlsData = nlsData,
+      drugs = drugs[1:3], analysis = "multiple", varFunction = 3)
R> nlmeData2 <- doNlme(mixlowData = mixlowData, nlsData = nlsData,
+      drugs = drugs[1:3], analysis = "multiple", varFunction = 4)
```
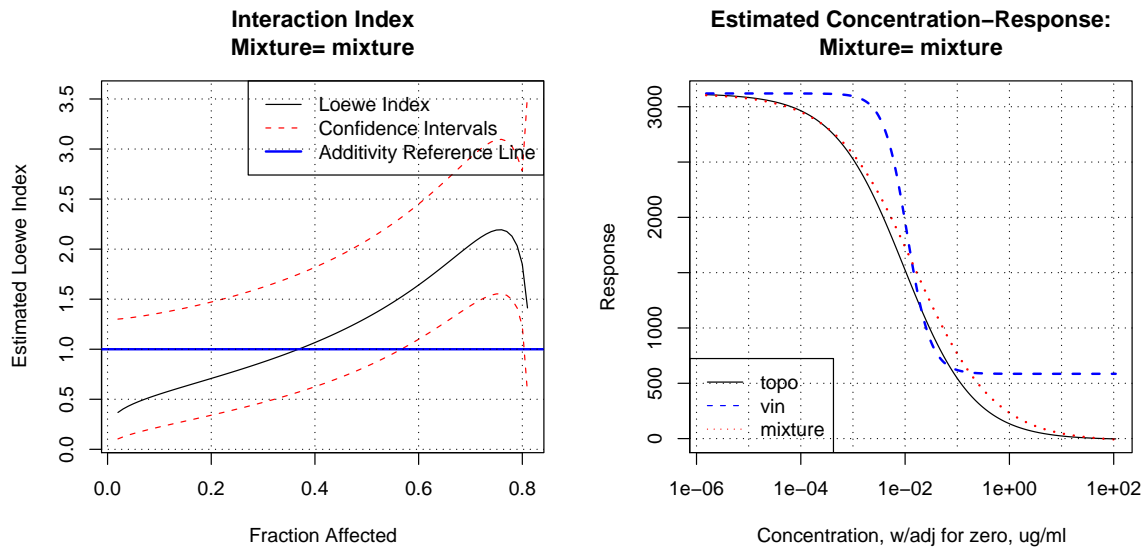
Figure 2: Loewe index for example data.

```
R> model1 <- nlmeData1$nlmeModels[[1]]
R> model2 <- nlmeData2$nlmeModels[[1]]
R> anova(model1, model2)

       Model df      AIC       BIC    logLik   Test  L.Ratio p-value
model1     1 12 8984.736 9038.423 -4480.368
model2     2 15 8759.268 8826.376 -4364.634 1 vs 2 231.4683  <.0001
```

4. Simulation study

Mixtures that contain more than a few drugs can be problematic in that the mixed-effects model may not converge or may take a long time to converge if all drugs and the mixture are analyzed together. If data are poorly behaved, convergence problems could occur for any size mixture. If convergence becomes a problem, one alternative is to estimate the concentration-response curve parameters separately for each drug and mixture. The user can instruct the `doNlme` function to perform this type of `single` analysis. The disadvantage of the single-drug approach is that control-well information is obtained from the trays of one drug only. In contrast, control-well data from all trays are used when all drugs and the mixture are analyzed together (`multiple` drug analysis). To determine if single-drug analysis makes a large difference in coverage of the confidence intervals of the Loewe index, a simulation was performed that compared the two approaches.

The simulation is based on Simulation C discussed in Boik *et al.* (2008). Model (1) was used to generate the simulation data using the parameters listed in Table 5. These parameters are typical for cytotoxicity assays. One thousand experiments were simulated. Each experiment involved three drugs tested in three trays, or nine trays in total. Each tray used 14 control

Parameter	Value
$\log(\gamma_d)$	-0.4138
$\psi_{d,0.5}$	-3.9841 μg/ml
μ	7.9429
σ_b	0.388
σ	2.1969
variance function	4
β_d	0.7429

Table 5: Parameters used to generate simulation data.

wells, two `bbt` wells, and 80 treatment wells (8 treatment wells for each of 10 drug concentrations). The experiment simulated a sham mixture of a drug with itself. Therefore, the parameters used to generate data for drug 1, drug 2, and drug 3 were identical. Data for drug 3 were taken to represent the sham mixture of a 50/50 mixture of drugs 1 and 2. As a sham mixture, the true Loewe index is 1.0 for all fraction affected values.

Average coverage of the confidence intervals for the Loewe index over all 1,000 experiments is shown in Figure 3 as a function of fraction affected values. The nominal coverage was 0.95. Mean coverage across all fraction affected values was 0.9612 for the multiple-drug analysis and 0.9607 for the single-drug analysis. Results suggest that coverage was near the nominal value
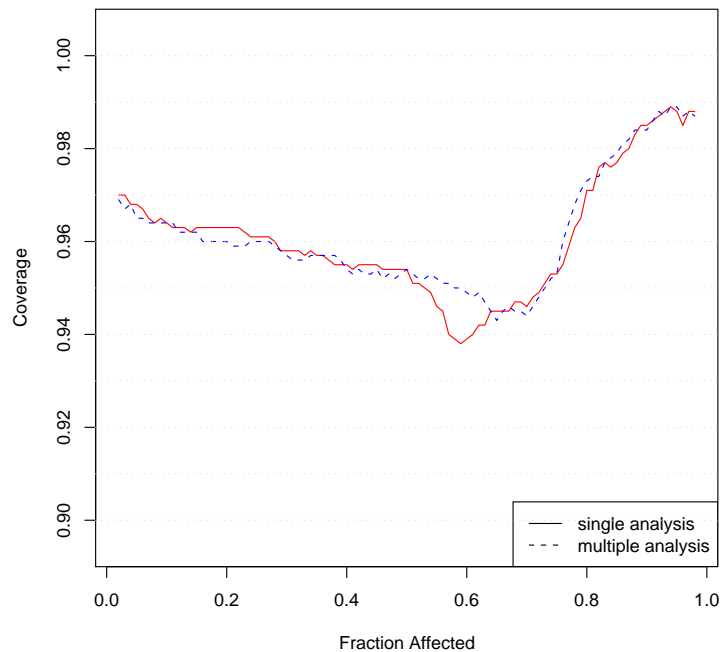

Figure 3: Average coverage of the Loewe index based on 1,000 simulated experiments. The nominal coverage coefficient was 0.95.

for all fraction affected values, regardless of the type of analysis used. At fraction affected values near 0.58, coverage for multiple-drug analysis was closer to nominal values than for single-drug analysis (0.950 vs. 0.938, respectively). In conclusion, both types of analysis produced excellent coverage. Multiple-drug analysis is recommended, however, if convergence is not an issue.

5. Summary

This paper has introduced the **mixlow** package in R. The package should be useful for students, researchers, and technicians who need to obtain accurate parameter estimates for sigmoidal concentration-response curves based on in-vitro experiments. In addition, it should be useful to those who must estimate a drug interaction index to determine if a mixture is antagonistic, additive, or synergistic.

Acknowledgments

References

Berenbaum MC (1989). "What is Synergy?" *Pharmacological Reviews*, **41**(2), 93–141.

Boik JC, Newman RA (2008). "A Classification Model to Predict Synergism/Antagonism of Cytotoxic Mixtures Using Protein-Drug Docking Scores." *BMC Pharmacology*, **8**, 13.

Boik JC, Newman RA, Boik RJ (2008). "Quantifying Synergism/Antagonism Using Nonlinear Mixed-Effects Modeling: A Simulation Study." *Statistics in Medicine*, **27**(7), 1040–1061.

Chou TC, Talalay P (1984). "Quantitative Analysis of Dose-Effect Relationships: The Combined Effects of Multiple Drugs or Enzyme Inhibitors." *Advances in Enzyme Regulation*, **22**, 27–55.

Greco WR, Bravo G, Parsons JC (1995). "The Search for Synergy: A Critical Review from a Response Surface Perspective." *Pharmacological Reviews*, **47**(2), 331–385.

Merlin JL (1994). "Concepts of Synergism and Antagonism." *Anticancer Research*, **14**(6A), 2315–2319.

Pinheiro J, Bates D, DebRoy S, Sarkar D, R Development Core Team (2009). "**nlme**: Linear and Nonlinear Mixed Effects Models. R Package Version 3.1-92." URL http://CRAN.R-project.org/package=nlme.

R Development Core Team (2009). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Ritz C, Streibig JC (2005). "Bioassay Analysis Using R." *Journal of Statistical Software*, **12**(5), 1–22. URL http://www.jstatsoft.org/v12/i05/.

Tallarida RJ (2001). "Drug Synergism: Its Detection and Applications." *The Journal of Pharmacology and Experimental Therapeutics*, **298**(3), 865–872.

Vonesh EF, Chinchilli VM, Pu K (1996). "Goodness-of-Fit in Generalized Nonlinear Mixed-Effects Models." *Biometrics*, **52**(2), 572–587.

Affiliation:

John C. Boik
New Earth BioMed
108 Rockview
Irvine, CA 92612, United States of America
E-mail: john.boik@newearthbiomed.org
URL: http://www.newearthbiomed.org/