

Bradley-Terry models in *R*

David Firth
University of Warwick, United Kingdom

<http://www.warwick.ac.uk/go/dfirth>

January 6, 2005

Abstract

This paper describes the *R* add-on package *BradleyTerry*, which facilitates the specification and fitting of Bradley-Terry logit models to pair-comparison data. Included are the standard ‘unstructured’ Bradley-Terry model, structured versions in which the parameters are related through a linear predictor to explanatory variables, and the possibility of an order or ‘home advantage’ effect. Model fitting is either by maximum likelihood or by bias-reduced maximum likelihood in which the first-order asymptotic bias of parameter estimates is eliminated. Also provided are a simple and efficient approach to handling missing covariate data, and suitably-defined residuals for diagnostic checking of the linear predictor; these are new methodological contributions which will be discussed in greater detail elsewhere.

Key words: generalized linear model, logistic regression, Jeffreys prior, ranking, tournament analysis, working residuals

1 Bradley-Terry model

1.1 Introduction

The Bradley-Terry model (Bradley and Terry, 1952) assumes that in a ‘contest’ between any two ‘players’, say player i and player j ($i, j \in \{1, \dots, K\}$), the odds that i beats j is α_i/α_j , where α_i and α_j are positive-valued parameters which might be thought of as representing ‘ability’. For a good general introduction see Agresti (2002). Applications are many, ranging from experimental psychology to the analysis of sports tournaments to genetics (for example, the allelic transmission/disequilibrium test of Sham and Curtis (1995) is based on a Bradley-Terry model in which the ‘players’ are alleles). The model can alternatively be expressed in the logit-linear form

$$\text{logit}[\text{pr}(i \text{ beats } j)] = \lambda_i - \lambda_j, \quad (1)$$

where $\lambda_i = \log \alpha_i$ for all i . Thus, assuming independence of all contests, the parameters λ_i , λ_j , etc., can be estimated by maximum likelihood using standard software for generalized linear models, with a suitably specified model matrix. The primary purpose of the *BradleyTerry* package, implemented in the *R* statistical computing environment (Ihaka and Gentleman, 1996; R Development Core Team, 2003), is to facilitate the specification and fitting of such models, including special cases in which the ability parameters are related to available explanatory variables through a linear predictor of the form $\lambda_i = \sum_{r=1}^p \beta_r x_{ir}$.

1.2 Example: analysis of journal citations

The following comes from page 448 of Agresti (2002), extracted from the larger table of Stigler (1994). The data are counts of citations among four prominent journals of statistics:

```
> data(citations)
> citations
      winner      loser Freq
1  Biometrika Biometrika  NA
2  Comm Statist Biometrika  33
3      JASA Biometrika  320
4  JRSS-B Biometrika  284
5  Biometrika Comm Statist  730
6  Comm Statist Comm Statist  NA
7      JASA Comm Statist  813
8  JRSS-B Comm Statist  276
9  Biometrika      JASA  498
10 Comm Statist      JASA   68
11      JASA      JASA   NA
12  JRSS-B      JASA  325
13  Biometrika JRSS-B  221
14 Comm Statist JRSS-B   17
15      JASA JRSS-B  142
16  JRSS-B JRSS-B   NA
```

Here 'winner' means the cited journal, 'loser' the journal in which the citation appears; thus, for example, *Biometrika* was cited 498 times by papers in *JASA* during the period under study. The Bradley-Terry model can now be fitted by using function `BTm` from the *BradleyTerry* package. Here we fit the model and store the result as an object named `citeModel`:

```
> library(BradleyTerry)
> citeModel <- BTm(citations ~ ..)
> citeModel
```

```
Call: BTm(formula = citations ~ ..)
```

```
Coefficients:
..Comm.Statist      ..JASA      ..JRSS.B
      -2.9491      -0.4796      0.2690
```

```
Degrees of Freedom: 6 Total (i.e. Null); 3 Residual
```

```
Null Deviance: 1925
```

```
Residual Deviance: 4.293 AIC: 46.39
```

The coefficients here are maximum likelihood estimates of $\lambda_2, \lambda_3, \lambda_4$, with λ_1 (the log-ability for *Biometrika*) set to zero as an identifying convention.

Note the use of the special right-hand-side formula '`..`', which is used to specify the linear predictor $\lambda_i - \lambda_j$ of the standard Bradley-Terry model.

If a different 'reference' journal is required, this can be achieved using the optional `refcat` argument: for example, making use of `update` to avoid re-specifying the whole model,

```
> update(citeModel, . ~ ., refcat = "JASA")
```

```
Call: BTm(formula = citations ~ .., refcat = "JASA")
```

```
Coefficients:
..Biometrika ..Comm.Statist      ..JRSS.B
      0.4796      -2.4695      0.7485
```

```
Degrees of Freedom: 6 Total (i.e. Null); 3 Residual
```

```
Null Deviance: 1925
```

```
Residual Deviance: 4.293 AIC: 46.39
```

— the same model in a different parameterization.

The use of the standard Bradley-Terry model for this application is of course rather questionable — for example, citations within a published paper can hardly be considered independent, and the model discards potentially important information on self-citation. Stigler (1994) provides arguments to defend the model's use despite such concerns.

2 Abilities predicted by explanatory variables

In some application contexts there may be 'player-specific' explanatory variables available, and it is then natural to consider model simplification of the form

$$\lambda_i = \sum_{r=1}^p \beta_r x_{ir}, \quad (2)$$

in which ability of each player i is related to explanatory variables x_{i1}, \dots, x_{ip} through a linear predictor with coefficients β_1, \dots, β_p . See, for example, Springall (1973). The `BTm` function allows such models to be specified in a natural way using the standard *S*-language model formulae.

As a very simple illustration with just one predictor, consider the citations model above but with ability determined by the journal's country of origin:

```
> journalNames <- levels(citations$winner)
> journalData <- data.frame(origin = c("UK", "USA", "USA", "UK"),
+                             row.names = journalNames)
> citeModel2 <- BTm(citations ~ origin, data = journalData)
> citeModel2
```

```
Call: BTm(formula = citations ~ origin, data = journalData)
```

```
Coefficients:
```

```
originUSA
-1.273
```

```
Degrees of Freedom: 6 Total (i.e. Null); 5 Residual
```

```
Null Deviance: 1925
```

```
Residual Deviance: 1139 AIC: 1177
```

The UK journals have an estimated advantage in (log) ability of 1.273 over the USA journals. This model saves two parameters, but at the expense of severe lack of fit: clearly journals' ability to be cited varies significantly within at least one of the two countries of origin.

The 'standard' Bradley-Terry model from §1.2 above could have been specified in the same way:

```
> journal <- as.factor(row.names(journalData))
> BTm(citations ~ journal)
```

```
Call: BTm(formula = citations ~ journal)
```

```
Coefficients:
```

```
journalComm.Statist    journalJASA    journalJRSS.B
-2.9491             -0.4796             0.2690
```

```
Degrees of Freedom: 6 Total (i.e. Null); 3 Residual
```

```
Null Deviance: 1925
```

```
Residual Deviance: 4.293 AIC: 46.39
```

The special model formula ‘`..`’ used in §1.2 provides a convenient shorthand for the specification of this model.

3 Missing values

The `NA` values in the journal-citation data above appear in data rows that are not used in the Bradley-Terry model. Such rows in the data frame of contest results (i.e., the left-hand side of the model formula) are simply discarded by `BTm`.

Where there are missing values in player-specific *predictor* (or *explanatory*) variables which appear on the right-hand side of the model formula, it will typically be very wasteful to discard all contests involving players for which some values are missing. Instead, such cases are accommodated by the inclusion of one or more parameters in the model. If, for example, player 1 has one or more of its predictor values x_{11}, \dots, x_{1p} missing, then the combination of (1) and (2) above yields

$$\text{logit}[\text{pr}(1 \text{ beats } j)] = \lambda_1 - \sum_{r=1}^p \beta_r x_{jr},$$

for all other players j . This results in the inclusion of a ‘direct’ ability parameter for each player having missing predictor values, in addition to the common coefficients β_1, \dots, β_p — an approach which will be appropriate when the missingness mechanism is unrelated to contest success. The same device can be used also to accommodate specified departures from a structured Bradley-Terry model, whereby some players have their abilities determined by the linear predictor but others do not.

As a simple illustration, consider the previous citations model in which country of origin is unknown for one of the journals (say, *Communications in Statistics*):

```
> is.na(journalData$origin[2]) <- TRUE
> journalData
      origin
Biometrika      UK
Comm Statist    <NA>
JASA             USA
JRSS-B           UK
> update(citeModel2, . ~ .)

Call:  BTm(formula = citations ~ origin, data = journalData)

Coefficients:
..Comm.Statist      originUSA
      -3.0317          -0.5726

Degrees of Freedom: 6 Total (i.e. Null);  4 Residual
Null Deviance:      1925
Residual Deviance: 18.86  AIC: 58.96
```

The fit of this model — which in effect allows distinct abilities for *JASA* and *Communications in Statistics*, is better (as evidenced by the much-reduced deviance) than the previous model, but is still unacceptable. The two UK journals differ significantly in ability, as may be seen from a summary of the original three-parameter fit:

```
> summary(citeModel)

Call:
BTm(formula = citations ~ ..)

Deviance Residuals:
```

Comm.Statist vs Biometrika	JASA vs Biometrika
-0.8476	0.5198
JASA vs Comm.Statist	JRSS.B vs Biometrika
0.0930	-0.2022
JRSS.B vs Comm.Statist	JRSS.B vs JASA
-1.6201	0.7941

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
..Comm.Statist	-2.94907	0.10255	-28.759	< 2e-16
..JASA	-0.47957	0.06059	-7.915	2.47e-15
..JRSS.B	0.26895	0.07083	3.797	0.000146

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1925.2329 on 6 degrees of freedom
 Residual deviance: 4.2934 on 3 degrees of freedom
 AIC: 46.394

Number of Fisher Scoring iterations: 4

The estimated difference of 0.269 between *JRSS-B* and the 'reference' journal *Biometrika* is highly significant (although the correlations likely in this dataset have probably caused the significance of all such comparisons to be overstated in these results).

4 Order effect

In certain types of application some or all contests have an associated 'bias', related to the order in which items are presented to a judge or with the location in which a contest takes place, for example. A natural extension of the Bradley-Terry model (1) is then

$$\text{logit}[\text{pr}(i \text{ beats } j)] = \lambda_i - \lambda_j + \delta z,$$

where $z = 1$ if i has the supposed advantage and $z = -1$ if j has it. (If the 'advantage' is in fact a disadvantage, δ will be negative.) The scores λ_i then relate to ability in the absence of any such advantage.

As an example, consider the baseball data given in Agresti (2002), p438:

```
> data(baseball)
> baseball
      winner    loser Freq home.adv
1 Milwaukee Milwaukee  NA        1
2 Milwaukee  Detroit    4        1
3 Milwaukee  Toronto    4        1
4 Milwaukee New York    4        1
5 Milwaukee  Boston    6        1
6 Milwaukee Cleveland    4        1
7 Milwaukee Baltimore    6        1
8  Detroit Milwaukee    3        1
...
48 Baltimore Cleveland    3        1
49 Baltimore Baltimore  NA        1
50 Milwaukee Milwaukee  NA       -1
51 Milwaukee  Detroit    3       -1
52 Milwaukee  Toronto    5       -1
...
```

```
97 Baltimore Cleveland    4      -1
98 Baltimore Baltimore   NA      -1
```

Here there are 7 teams, and for example Milwaukee beat Detroit 4 times at home (`home.adv` is 1) and 3 times away from home (`home.adv` is -1). The 'standard' Bradley-Terry model without a home-advantage parameter is fitted as before:

```
> baseballModel <- BTm(baseball ~ ..)
> summary(baseballModel)
```

Call:

```
BTm(formula = baseball ~ ..)
```

Deviance Residuals:

	Min	1Q	Median	3Q	Max
	-1.50067	-0.52962	-0.02198	0.32184	2.06170

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
..Boston	1.1077	0.3339	3.318	0.000908
..Cleveland	0.6839	0.3319	2.061	0.039345
..Detroit	1.4364	0.3396	4.230	2.34e-05
..Milwaukee	1.5814	0.3433	4.607	4.09e-06
..New.York	1.2476	0.3359	3.715	0.000203
..Toronto	1.2945	0.3367	3.845	0.000121

(Dispersion parameter for binomial family taken to be 1)

```
Null deviance: 49.699 on 21 degrees of freedom
Residual deviance: 15.737 on 15 degrees of freedom
AIC: 87.324
```

Number of Fisher Scoring iterations: 4

The reference team is Baltimore, estimated to be the weakest of these seven, with Milwaukee and Detroit the strongest.

Now add the home-advantage effect:

```
> baseballModel <- update(baseballModel, order.effect = baseball$home.adv)
> summary(baseballModel)
```

Call:

```
BTm(formula = baseball ~ .., order.effect = baseball$home.adv)
```

Deviance Residuals:

	Min	1Q	Median	3Q	Max
	-2.0381908	-0.3143223	0.0007748	0.7621586	2.2600074

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
..Boston	1.1438	0.3378	3.386	0.000710
..Cleveland	0.7047	0.3350	2.104	0.035417
..Detroit	1.4754	0.3446	4.282	1.85e-05
..Milwaukee	1.6196	0.3474	4.662	3.13e-06
..New.York	1.2813	0.3404	3.764	0.000167
..Toronto	1.3271	0.3403	3.900	9.64e-05
.order	0.3023	0.1309	2.308	0.020981

(Dispersion parameter for binomial family taken to be 1)

```
Null deviance: 78.015  on 42  degrees of freedom
Residual deviance: 38.643  on 35  degrees of freedom
AIC: 137.11
```

This reproduces the results given on page 438 of Agresti (2002): the home team has an estimated odds-multiplier of $\exp(0.3023) = 1.35$ in its favour.

5 Ability scores

The function `BTabilities` extracts estimates and standard errors for the log-ability scores $\lambda_1, \dots, \lambda_K$. These will either be 'direct' estimates, as in the standard Bradley-Terry model or for players with one or more missing predictor values, or 'model-based' estimates of the form $\hat{\lambda}_i = \sum_{r=1}^p \hat{\beta}_r x_{ir}$ for players whose ability is predicted by explanatory variables.

As a simple illustration, estimates in the origin-predicts-ability model for journal citation data are obtained by:

```
> BTabilities(citeModel2)
              ability      s.e.
Biometrika      0.0000 0.00000000
Comm.Statist -1.2732 0.04999872
JASA           -1.2732 0.04999872
JRSS.B          0.0000 0.00000000
```

Here precision is of course overstated (the reported standard errors are too small), since this particular model was a poor fit to the data.

6 Residuals

There are two main types of residuals available for a Bradley-Terry model object.

First, there are residuals obtained by the standard methods for models of class `glm`. These all deliver one residual for each contest or type of contest. For example, Pearson residuals for the model `citeModel2` can be obtained simply by

```
> residuals(citeModel2)
Comm.Statist vs Biometrika      JASA vs Biometrika
              -13.741187              11.121312
JASA vs Comm.Statist      JRSS.B vs Biometrika
              27.245984              2.807120
JRSS.B vs Comm.Statist      JRSS.B vs JASA
              7.624182              -4.286953
```

— from which the lack of fit is immediately apparent!

More useful for diagnostics on the linear predictor $\sum \beta_r x_{ir}$ are 'player'-level residuals, obtained by using the function `BTresiduals`:

```
> BTresiduals(citeModel2)
Biometrika Comm.Statist      JASA      JRSS.B
-0.09767683 -1.38124450  1.25835736  0.15117998
attr(,"weights")
Biometrika Comm.Statist      JASA      JRSS.B
396.4048    400.6950    439.8255    256.1157
```

These residuals estimate the error in the linear predictor; they are obtained by suitable aggregation of the so-called 'working' residuals from the `glm` fit. From these residuals it is immediately

evident, for example, that the origin-predicts-ability model understates the ability of *JASA* and overstates the ability of *Communications of Statistics* (and similarly for *JRSS-B* versus *Biometrika*). The `weights` attribute indicates the relative information in these residuals — weight is roughly inversely proportional to variance — which may be useful for plotting and/or interpretation; for example, a large residual may be of no real concern if based on very little information. Weighted least-squares regression of these residuals on any variable already in the model is null. For example:

```
> resid<- BTresiduals(citeModel2)
> journalData$origin[2] <- "USA" ## ie the previous value is restored
> lm(resids ~ origin, weights = attr(resids, "weights"),
+   data = journalData)
```

Call:

```
lm(formula = resid~ origin, weights = attr(resids, "weights"),
    data = journalData)
```

Coefficients:

```
(Intercept)      originUSA
 1.690e-16    -4.391e-16
```

7 Bias-reduced estimates

Model-fitting in `BTm` is by default computed by maximum likelihood, using an internal call to the `glm` function. An alternative is to fit by bias-reduced maximum likelihood (Firth, 1993): this requires additionally the `brlr` package, and is specified by the optional argument `br = TRUE`. The resultant effect, namely removal of first-order asymptotic bias in the estimated coefficients, is often quite small. One notable feature of bias-reduced fits is that all estimated coefficients and standard errors are necessarily finite, even in situations of ‘complete separation’ where MLEs take infinite values (Heinze and Schemper, 2002).

8 Model search

In addition to `update()` as illustrated above, methods for the generic functions `add1()` and `drop1()` are provided. These can be used in the standard way for model elaboration or specialization, and their availability also allows the use of `step()` for automated exploration of a set of candidate player-specific predictors.

9 Setting up the data

9.1 Contest results

The left-hand side of the model formula supplied to `BTm` is a data frame with at least two columns. The `citations` object shown in §1 above is an example; `baseball` in §4 is another. Each row represents a contest result. One column (either named `"winner"`, or the first column if no column has that name), is a factor indicating contest winners; another (either `"loser"`, or column 2) indicates contest losers. An optional numeric column named `"Freq"` contains the frequency of each result; if this column is absent, all frequencies are taken to be 1.

If `order.effect` is specified, it should be a numeric vector of the same length as the number of rows in the contest-results data frame. It may be convenient to store such a vector in the same data frame, as was done in the `baseball` dataset above. Values should be 1 where the winner is advantaged by the effect, -1 where the loser is advantaged, and 0 where neither player is advantaged.

To use only certain rows of the data in the analysis, the `subset` argument may be used in the call to `BTm`. This should either be a logical vector of the same length as the number of rows in the contest-results data frame, or a numeric vector containing the indices of rows to be used.

9.2 Predictors

Variables which appear in the right-hand side of the model formula are ‘player’-level predictor variables. The safest approach is to put all potential predictor (explanatory) variables—including factors and any offset term—into a data frame like `journalData` above, with one row per (potential) player, and with row names the names of players exactly as used in the `"winner"` and `"loser"` columns of the contest-results data frame. The `data` argument to `BTm`, which applies only to right-hand side variables, is then used to identify the data frame in which predictors (and any offset) can be found.

An offset in the model can be specified using the `offset` argument to `BTm`, which should be a vector of length equal to that of the other right-hand side variables (and which should, for tidiness, come from the same data frame as other predictors).

10 A list of the functions provided in *BradleyTerry*

The standard *R* help files provide the definitive reference. Here we simply list all of the packaged functions and their arguments, as a convenient overview:

```
BTm(formula, refcat = NULL, offset = NULL, contrasts = NULL, data = NULL,
     subset = NULL, br = FALSE, order.effect = NULL, ...)
BTabilities(model)
BTresiduals(model)
add1.BTm(object, scope, scale = 0, test = c("none", "Chisq", "F"),
          x = NULL, k = 2, ...)
drop1.BTm(object, scope, scale = 0, test = c("none", "Chisq", "F"),
          k = 2, ...)
formula.BTm(x, ...)
terms.BTm(x, ...)
```

11 What is *not* in the *BradleyTerry* package?

The *BradleyTerry* package does not provide:

- any methods for dealing with ties, i.e., contests in which neither player wins.
- any facilities either for handling contest-specific (as opposed to player-specific) predictor variables, except for the possibility of an order effect as described above.

These extensions to the Bradley-Terry model can be achieved in *R* (or elsewhere) by fitting suitably constructed log-linear models—see, for example, Critchlow and Fligner (1991) and Dittrich et al. (1998). They are outside the scope of the *BradleyTerry* package, whose purpose is to simplify the specification and fitting of Bradley-Terry models with *player-specific* predictors (including of course the ‘saturated’ case of the standard Bradley-Terry model (1)).

A useful extension of the *BradleyTerry* package would be to allow the inclusion of a player-specific random effect, as in

$$\lambda_i = \sum_{r=1}^p \beta_r x_{ir} + U_i,$$

with the $\{U_i\}$ distributed independently as $N(0, \sigma_U)$ for example, to allow for imperfect representation of ability by the linear predictor $\sum \beta_r x_{ir}$. Work on this is in progress.

Acknowledgment

The author is grateful to Simon Blomberg of the Australian National University who asked a question which motivated this work, and to two anonymous referees for thorough and constructive comments which greatly improved both the paper and the software.

References

- Agresti, A. (2002). *Categorical Data Analysis* (Second edition). Wiley.
- Bradley, R. A. and M. E. Terry (1952). Rank analysis of incomplete block designs I: The method of paired comparisons. *Biometrika* 39, 324–45.
- Critchlow, D. E. and M. A. Fligner (1991). Paired comparison, triple comparison, and ranking experiments as generalized linear models, and their implementation in GLIM. *Psychometrika* 56, 517–533.
- Dittrich, R., R. Hatzinger, and W. Katzenbeisser (1998). Modelling the effect of subject-specific covariates in paired comparison studies with an application to university rankings. *Applied Statistics* 47, 511–525.
- Firth, D. (1993). Bias reduction of maximum likelihood estimates. *Biometrika* 80, 27–38.
- Heinze, G. and M. Schemper (2002). A solution to the problem of separation in logistic regression. *Statistics in Medicine* 21, 2409–2419.
- Ihaka, R. and R. Gentleman (1996). R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics* 5(3), 299–314.
- R Development Core Team (2003). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing. ISBN 3-900051-00-3.
- Sham, P. C. and D. Curtis (1995). An extended transmission/disequilibrium test (TDT) for multi-allele marker loci. *Annals of Human Genetics* 59(3), 323–336.
- Springall, A. (1973). Response surface fitting using a generalization of the Bradley-Terry paired comparisons model. *Applied Statistics* 22, 59–68.
- Stigler, S. (1994). Citation patterns in the journals of statistics and probability. *Statistical Science* 9, 94–108.