'Extended' Bradley-Terry models

David Firth and Heather Turner

Department of Statistics
University of Warwick

Munich, 2010–02–25

Pair-comparison studies

Sport: player i beats player j
Psychometrics: object i is preferred to object j

Sport (etc.): interest in *players* and their attributes
Psychometrics (etc.): interest in *judges* (*subjects*) and their attributes

Bradley-Terry model

The basic model:

$$\mathrm{pr}(i \text{ beats } j) = \frac{\alpha_i}{\alpha_i + \alpha_j},$$

with α_i the relative 'ability' of object i.

Work with *log* abilities:

$$\begin{aligned} \mathrm{logit}[\mathrm{pr}(i \text{ beats } j)] &= \log(\alpha_i) - \log(\alpha_j) \\ &= \lambda_i - \lambda_j. \end{aligned}$$

Extensions?

We will focus here on three possible directions from the basic model:

1. (Log-)abilities λ_i determined/predicted by object covariate vector x_i.
2. $\lambda_i \to \lambda_{ik}$: the ability of object i *varies* between different comparisons k.
3. i versus j, no preference? ('tied' comparisons)

'Structured' Bradley-Terry model

$$\begin{aligned} \lambda_i &= f_i(\beta) + U_i \\ &= \sum_r \beta_r x_{ir} + U_i \qquad \text{(for example)} \end{aligned}$$

- attributes of objects/players predict ability
- U_i is random error, with variance σ^2, say — needed in order to allow for imperfect prediction
- $\Rightarrow$ complex random effects model, with linear predictor

$$\sum_r (x_{ir} - x_{jr})\beta_r + (U_i - U_j)$$

Ability varying between comparisons

$$\lambda_i \to \lambda_{ik}$$

e.g., time-varying covariates,

$$\lambda_{ik} = \sum_r \beta_r x_{ikr} + U_i$$

e.g., subject-specific abilities,

$$\lambda_{ik} = \lambda_{is},$$

where $s = s(k)$ identifies the subject who makes comparison k.

e.g., abilities predicted by subject covariates,

$$\lambda_{is} = \sum_t \gamma_{it} z_{st} + E_{is}$$

Ability varying between comparisons (continued)

e.g., still with abilities λ_{is} varying between subjects, a particular form likely to be useful is *multiplicative* interaction,

$$\lambda_{is} = \lambda_i \exp\left(\sum_t \gamma_t z_{st}\right) + E_{is}$$

This last form is not yet implemented in the *BradleyTerry2* package; it will require features from the *gnm* (generalized nonlinear models) package.

Ties

What to do when neither i nor j is preferred?

Elaborate the Bradley-Terry model? (Rao and Kupper, 1967; Davidson, 1970)

A crude alternative approach/approximation:

$$\text{tie} = \text{half a `win' for each of } i \text{ and } j$$

Suggests a generalization: half → some other fraction?

Implementation in *R*: The *BradleyTerry2* package

Main new features

- flexible formula interface to modelling fitting function `BTm()`: allows object-specific, subject-specific, contest-specific variables and random effects [limited implementation]
- efficient data management of multiple data frames

Best of original *BradleyTerry* package

- translation of formula to appropriate design matrix
- methods for fitted model object, e.g. anova, BTabilities
- missing data handling

CEMS Data

The `CEMS` data (Dittrich et al, 1998) concern the preferences of students in selecting a school from the Community of European Management Schools for their international visit.

- 6 CEMS schools are covered in the survey
- students were to choose between each pair of schools (ties allowed)
- further data collected on students e.g. type of degree, language skills

Data Structure

```
> library(BradleyTerry2); data(CEMS); str(CEMS)
List of 3
 $ preferences:'data.frame': 4545 obs. of  8 variables:
  ..$ student : num [1:4545] 1 1 1 1 1 1 1 1 1 1 ...
  ..$ school1 : Factor w/ 6 levels "Barcelona","London",..: 2 2 4
  ..$ school2 : Factor w/ 6 levels "Barcelona","London",..: 4 3 3
  ..$ win1    : num [1:4545] 1 1 NA 0 0 0 1 1 0 1 ...
  ...
 $ students   :'data.frame': 303 obs. of  8 variables:
  ..$ STUD: Factor w/ 2 levels "other","commerce": 1 2 1 2 1 1 1 2
  ..$ ENG : Factor w/ 2 levels "good","poor": 1 1 1 1 2 1 1 1 2 1
  ...
 $ schools    :'data.frame': 6 obs. of  7 variables:
  ..$ Barcelona: num [1:6] 1 0 0 0 0 0
  ..$ London   : num [1:6] 0 1 0 0 0 0
...
```


Model Specification

Model specifiation is controlled by four arguments to `BTm()`

`outcome` a binomial response as accepted by `glm()`.

`player1, player2` specify the players in each contest and any other player-specific contest variables in data frames with the same attributes.

`id` the name of the factor in `player1`/`player2` that gives the identity of the player.

`formula` a one-sided formula for player ability.

Standard Bradley Terry Model

A Bradley-Terry model with a separate ability for each player can be specified as follows

```
> standardBT <-  BTm(outcome = cbind(win1.adj, win2.adj),
      player1 = data.frame(school = school1),
      player2 = data.frame(school = school2),
      id = "school", formula = ~ school,
      refcat = "Stockholm",
      data = CEMS$preferences)
```

Or we can use the default id, ".."

```
> standardBT <-  BTm(outcome = cbind(win1.adj, win2.adj),
      player1 = school1, player2 = school2,
      formula = ~ .., refcat = "Stockholm",
      data = CEMS$preferences)
```


Model Summaries

For models with no random effects, BTm returns an object which is essentially a "glm" object, hence the usual model summaries can be obtained, e.g. print():

```
Bradley Terry model fit by glm.fit

Call:  BTm(outcome = cbind(win1.adj, win2.adj), player1 = school1,
player2 = school2, formula = ~.., refcat = "Stockholm",
data = CEMS$preferences)

Coefficients:
..Barcelona     ..London     ..Milano      ..Paris  ..St.Gallen
     0.5379       1.5975       0.3878       0.9064       0.5251

Degrees of Freedom: 4454 Total (i.e. Null);  4449 Residual
  (91 observations deleted due to missingness)
Null Deviance:      5499
Residual Deviance: 4929   AIC: 5854
Warning message:
In eval(expr, envir, enclos) : non-integer counts in a binomial glm!
```


Object and Subject Variables

The final model in Dittrich et al, incorporating interactions with subject-covariates, can be estimated as follows

```
> interactionBT <-  BTm(outcome = cbind(win1.adj, win2.adj),
      player1 = school1, player2 = school2,
      formula = ~ .. +
      WOR[student] * LAT[..] +
      DEG[student] * St.Gallen[..] +
      STUD[student] * (Paris[..] + St.Gallen[..]) +
      ENG[student] * St.Gallen[..] +
      FRA[student] * (London[..] + Paris[..]) +
      SPA[student] * Barcelona[..] +
      ITA[student] * (London[..] + Milano[..]) +
      SEX[student] * Milano[..],
      refcat = "Stockholm", data = CEMS)
```


Interaction Model

```
> summary(interactionBT)$coef[, 1:2]/1.75

                                  Estimate Std. Error
..Barcelona                      1.0363917 0.10184195
..London                         1.2734839 0.10523535
..Milano                         1.1136211 0.10030192
..Paris                          0.6453467 0.05797807
..St.Gallen                      0.2487781 0.05663021
WOR[student]yes:LAT[..]          0.5933091 0.12278686
DEG[student]yes:St.Gallen[..]    0.2726479 0.06875424
STUD[student]commerce:Paris[..]  0.4073965 0.07352900
St.Gallen[..]:STUD[student]commerce -0.1984449 0.07089058
St.Gallen[..]:ENG[student]poor   0.1449582 0.07241576
FRA[student]poor:London[..]     -0.1607138 0.07519284
Paris[..]:FRA[student]poor      -0.7142351 0.07132559
SPA[student]poor:Barcelona[..]  -0.8409595 0.10336192
London[..]:ITA[student]poor     -0.2967857 0.10342156
ITA[student]poor:Milano[..]     -0.9603892 0.10386091
Milano[..]:SEX[student]male     -0.1743107 0.06848606
```


Baseball Data

The baseball data (Agresti, 2002) gives the results for 7 teams of the Eastern Division of the American League during the 1987 season:

```
> str(baseball)

'data.frame': 42 obs. of  4 variables:
 $ home.team: Factor w/ 7 levels "Baltimore","Boston",..: 5 5 5 5 5
 $ away.team: Factor w/ 7 levels "Baltimore","Boston",..: 4 7 6 2 3
 $ home.wins: int  4 4 4 6 4 6 3 4 4 6 ...
 $ away.wins: int  3 2 3 1 2 0 3 2 3 0 ...
```


Standard Bradley-Terry Model

```
> (baseballModel1 <- BTm(cbind(home.wins, away.wins), home.team,
away.team, data = baseball, id = "team"))

Bradley Terry model fit by glm.fit

Call:  BTm(outcome = cbind(home.wins, away.wins),
 player1 = home.team, player2 = away.team, id = "team",
 data = baseball)

Coefficients:
   teamBoston  teamCleveland    teamDetroit  teamMilwaukee
       1.1077         0.6839         1.4364         1.5814
 teamNew York    teamToronto
       1.2476         1.2945

Degrees of Freedom: 42 Total (i.e. Null);  36 Residual
Null Deviance:     78.02
Residual Deviance: 44.05   AIC: 140.5
```

Player-specific Contest Variables

```
> baseball$home.team <- data.frame(team = baseball$home.team,
+                                 at.home = 1)
> baseball$away.team <- data.frame(team = baseball$away.team,
+                                 at.home = 0)
> baseballModel2 <- update(baseballModel1,
+                         formula = ~ team + at.home)

...
Coefficients:
   teamBoston  teamCleveland    teamDetroit  teamMilwaukee
       1.1438         0.7047         1.4754         1.6196
 teamNew York    teamToronto        at.home
       1.2813         1.3271         0.3023

Degrees of Freedom: 42 Total (i.e. Null);  35 Residual
Null Deviance:     78.02
Residual Deviance: 38.64  AIC: 137.1
```


Comparing Models

```
> anova(baseballModel1, baseballModel2)

Analysis of Deviance Table

Response: cbind(home.wins, away.wins)

Model 1: ~team
Model 2: ~team + at.home
  Resid. Df Resid. Dev Df Deviance
1        36     44.053
2        35     38.643  1   5.4106
```


Springall Data

The `springall` data (Springall, 1973) gives the results of an experiment in which assessors were asked to determine which of two samples had the lesser flavour strength.

Samples were determined by a 3 x 3 factorial design, with factors flavour contentration and gel concentration.

The aim of the experiment was to describe the response surface over the two factors.

Random Effects

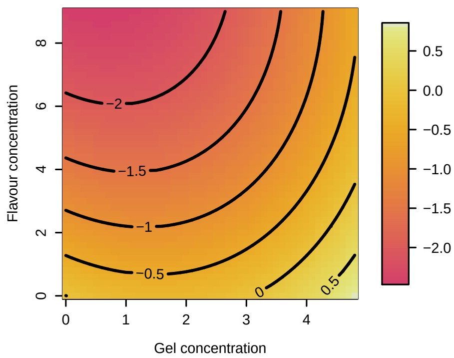
The flavour strength over the design region can be modelled by a second order response surface model, with random effects to allow for variation between samples with the same covariates:

```
> springall.model <- BTm(cbind(win.adj, loss.adj), col, row,
     ~ flav[..] + gel[..] +
       flav.2[..] + gel.2[..] + flav.gel[..] +
       (1 | ..),
       data = springall)
```


Response Surface Model

```
Bradley Terry model fit by glmmPQL.fit
PQL algorithm converged to fixed effects model

Call:  BTm(outcome = cbind(win.adj, loss.adj), player1 = col,
player2 = row, formula = ~flav[..] + gel[..] + flav.2[..] +
gel.2[..] + flav.gel[..] + (1 | ..), data = springall)

Coefficients:
    flav[..]      gel[..]    flav.2[..]     gel.2[..]  flav.gel[..]
    -0.41194     -0.32578       0.01565       0.10506       0.02376

Degrees of Freedom: 36 Total (i.e. Null);  31 Residual
Null Deviance:     327.9
Residual Deviance: 15.47  AIC: 113
```

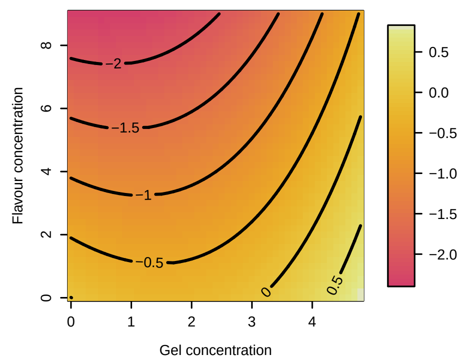

Second Order Response Surface



Simplified Model

```
> springall.model2 <- update(springall.model, ~ . - flav.2[..])
Bradley Terry model fit by glmmPQL.fit

Call:
BTm(outcome = cbind(win.adj, loss.adj), player1 = col, player2 = row,
    formula = ~flav[..] + gel[..] + gel.2[..] + flav.gel[..] +
        (1 | ..), data = springall)

Fixed effects:

    flav[..]       gel[..]     gel.2[..]  flav.gel[..]
    -0.26366      -0.32690       0.10416       0.02476

Random Effects Std. Dev.: 0.1406561
```


Fitted Response Surface



New work on ties (not yet in *BradleyTerry2*)

Davidson (1970) formulation:

$$\mathrm{pr}(\mathrm{tie}) = \frac{\nu\sqrt{\alpha_i\alpha_j}}{\alpha_i + \alpha_j + \nu\sqrt{\alpha_i\alpha_j}}$$

$$\mathrm{pr}(i \text{ beats } j \,|\, \text{not tied}) = \frac{\alpha_i}{\alpha_i + \alpha_j}$$

For inference: either

- discard ties, use the conditional likelihood (robust?)
- ML for all parameters including ν (efficient?)

A log-linear model. But too restrictive?

$$\nu \to \infty: \ \mathrm{pr}(\mathrm{tie}) \to 1$$

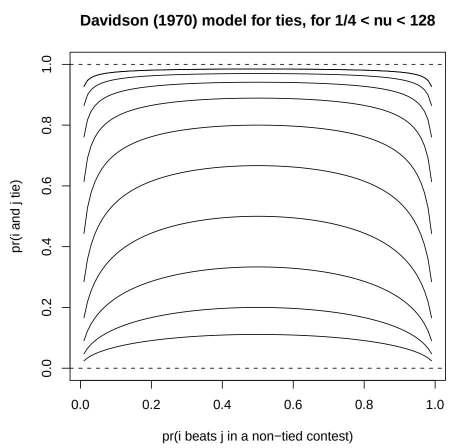
$$\nu \to 0: \ \mathrm{pr}(\mathrm{tie}) \propto \nu\sqrt{\alpha_i\alpha_j}/(\alpha_i + \alpha_j) \quad \text{(approx.)}$$

The single extra parameter ν conflates

- overall (max) probability of a tie
- strength of dependence of $\mathrm{pr}(\mathrm{tie})$ on α_i, α_j.

And the strongest dependence allowed (i.e., as $\nu \to 0$) is actually rather weak.

(Same comments apply to the Rao-Kupper model for ties.)



A '2-parameter' model for ties

Details omitted here — paper in preparation, preprint to appear soon at `http://go.warwick.ac.uk/dfirth`