Package: welo (via r-universe)

November 28, 2026

Title Weighted and Standard Elo Rates

Version 0.1.6

Description Estimates the standard and weighted Elo (WElo, Angelini et al., 2022 <doi:10.1016/j.ejor.2021.04.011>) rates. The current version provides Elo and WElo rates for tennis, according to different systems of weights (games or sets) and scale factors (constant, proportional to the number of matches, with more weight on Grand Slam matches or matches played on a specific surface). Moreover, the package gives the possibility of estimating the (bootstrap) standard errors for the rates. Finally, the package includes betting functions that automatically select the matches on which place a bet.

License GPL-3

Encoding UTF-8

LazyData true

RdMacros Rdpack

Depends R (>= 4.1.0)

Imports xts (>= 0.12.0), Rdpack (>= 1.0.0), boot (>= 1.3), rio (>= 0.5.29), ggplot2 (>= 3.4.0), reshape2 (>= 1.4.4)

Suggests knitr

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Vincenzo Candila [aut, cre]

Maintainer Vincenzo Candila <vcandila@unisa.it>

Repository https://cran.r-universe.dev

Date/Publication 2026-11-28 13:17:58 UTC

RemoteUrl https://github.com/cran/welo

RemoteRef HEAD

RemoteSha 3eebc2c73d3b6033493e120a0115a105b528a1c4

Contents

atp_2019 . . . 2
betting . . . 3
clean . . . 4
random_betting . . . 5
rank_plot . . . 7
tennis_data . . . 7
tennis_prob . . . 8
welo_plot . . . 9
welofit . . . 10
wta_2019 . . . 12

Index **13**

`atp_2019` *ATP matches in 2019*

Description

Tennis data for male matches played in 2019. Details can be found on https://www.tennis-data.co.uk/notes.txt

Usage

```
data(atp_2019)
```

Format

An object of class `"data.frame"`.

Source

Tennis archive from https://www.tennis-data.co.uk/

Examples

```
head(atp_2019)
str(atp_2019)
```

`betting` *Betting function*

Description

Places bets using the WElo and Elo probabilities, on the basis of two thresholds r and q, according to Angelini et al. (2022). By default, the amount of $1 is placed on the best odds (that is, the highest odds available) for player i for all the matches where it holds that

$$\frac{\hat{p}_{i,j}(t)}{q_{i,j}(t)} > r,$$

where $\hat{p}_{i,j}(t)$ is the estimated probability (coming from the WElo or Elo model) that player i wins the match t against player j and $q_{i,j}(t)$ is its implied probability obtained as the reciprical of the Bet365 odds. The implied probability $q_{i,j}(t)$ is assumed to be greater than q. If $q = 0$, all the players are considered. If q increases, heavy longshot players are excluded. In general, higher thresholds r and q imply less betting opportunities.

Usage

```
betting(
  x,
  r,
  q,
  model,
  bets = "Best_odds",
  R = 2000,
  alpha = 0.1,
  start_oos = NULL,
  end_oos = NULL
)
```

Arguments

`x`	an object of class 'welo', obtained from the `welofit` function
`r`	Vector or scalar identifying the threshold of the ratio between the estimated and the implied probability (see above)
`q`	Scalar parameter used to exclude the heavy underdogs signalled by Bet365 bookmaker. No bets will be placed on those matches where players have implied probabilities smaller than q
`model`	Valid choices are: "WELO" and "ELO"
`bets`	**optional** Parameter identifying on which type of odds the bet is placed. Default to "Best_odds". Valid choices are: "Best_odds", "Avg_odds" and "B365_odds". "Best_odds" are the highest odds available. "Avg_odds" are the average odds for that match and "B365_odds" are the Bet365 odds

R	**optional** Number of bootstrap replicates to calculate the confidence intervals. Default to 2000
alpha	**optional** Significance level for the boostrap confidence intervals. Default to 0.1
start_oos	**optional** Character parameter denoting the starting year for the bets. If included (default to NULL), then the bets will be placed on matches starting in that year. It has to be formatted as "YYYY"
end_oos	**optional** Character parameter denoting the ending year for the bets. If included (default to NULL), then the bets will be placed on matches included in the period "start_oos/end_oos". It has to be formatted as "YYYY"

Value

A matrix including the number of bets placed, the Return-on-Investiment (ROI), expressed in percentage, and its boostrap confidence interval, calculated using R replicates and the significance level α.

Examples

```
data(atp_2019)
db_clean<-clean(atp_2019)
db_est<-welofit(db_clean)
bets<-betting(db_est,r=c(1.1,1.2,1.3),q=0.3,model="WELO")
bets
```

---

clean *Cleaning function*

Description

Cleans the dataset in order to create a suitable data.frame ready to be used in the `welofit` function.

Usage

```
clean(x, MNM = 10, MRANK = 500)
```

Arguments

x	Data to be cleaned. It must be a data.frame coming from https://www.tennis-data.co.uk/.
MNM	**optional** Minimum number of matches played by each player to include in the cleaned dataset. Default to 10. This means that each player has to play at least 10 matches
MRANK	**optional** Maximum rank of the players to consider. Default to 500. This means that all the matches with players with ranks greater than 500 are dropped

Details

The cleaning operations are:

1. Remove all the uncompleted matches;
2. Remove all the NAs from B365 odds;
3. Remove all the NAs from the variable "ranking";
4. Remove all the NAs from the variable "games";
5. Remove all the NAs from the variable "sets";
6. Remove all the matches where the B365 odds are equal;
7. Define players i and j and their outcomes (Y_i and Y_j);
8. Remove all the matches of players who played less than MNM matches;
9. Remove all the matches of players with rank greater than MRANK;
10. Sort the matches by date.

Value

Data.frame cleaned

Examples

```
data(atp_2019)
db_clean<-clean(atp_2019)
str(db_clean)
```

---

random_betting *Random betting function*

Description

Places bets on players i and j randomly chosen, among all the matches selected by the following strategy: by default, the amount of \$1 is placed on the best odds (that is, the highest odds available) for player i for all the matches where it holds that

$$\frac{\hat{p}_{i,j}(t)}{q_{i,j}(t)} > r,$$

where $\hat{p}_{i,j}(t)$ is the estimated probability (coming from the WElo or Elo model) that player i wins the match t against player j and $q_{i,j}(t)$ is its implied probability obtained as the reciprical of the Bet365 odds. The implied probability $q_{i,j}(t)$ is assumed to be greater than q. If $q = 0$, all the players are considered. If q increases, heavy longshot players are excluded. Once got the number of matches satisfying the previously described strategy, each player (i and j) on which place a bet is randomly selected. Then the Return-on-Investiment (ROI) of this strategy is stored. Finally, the mean of the ROI obtained from repeating this operation B times is reported.

Usage

```
random_betting(
  x,
  r,
  q,
  model,
  bets = "Best_odds",
  B = 10000,
  start_oos = NULL,
  end_oos = NULL
)
```

Arguments

x	an object of class 'welo', obtained from the welofit function
r	Vector or scalar identifying the threshold of the ratio between the estimated and the implied probability (see above)
q	Scalar parameter used to exclude the heavy underdogs signalled by B365 bookmaker. No bets will be placed on those matches where players have odds smaller than q
model	Valid choices are: "WELO" and "ELO"
bets	**optional** Parameter identifying on which type of odds the bet is placed. Default to "Best_odds". Valid choices are: "Best_odds", "Avg_odds" and "B365_odds". "Best_odds" are the highest odds available. "Avg_odds" are the average odds and "B365_odds" are the Bet365 odds
B	**optional** Number of replicates to calculate the overall mean ROI. Default to 10000
start_oos	**optional** Character parameter denoting the starting year for the bets. If included (default to NULL), then the bets will be placed on matches starting in that year. It has to be formatted as "YYYY"
end_oos	**optional** Character parameter denoting the ending year for the bets. If included (default to NULL), then the bets will be placed on matches included in the period "start_oos/end_oos". It has to be formatted as "YYYY"

Value

A matrix reporting the number of bets and the mean of the ROI (in percentage) across the B values for every threshold r used

Examples

```
data(atp_2019)
db_clean<-clean(atp_2019)
db_est<-welofit(db_clean)
rand_bets<-random_betting(db_est,r=c(1.1,1.2,1.3),q=0.3,model="WELO",B=1000)
rand_bets
```

rank_plot *Plot for official (ATP or WTA) rates*

Description

Plots the official (ATP or WTA) rates.

Usage

```
rank_plot(x, players, line_width = 1.5, nbreaks = 1)
```

Arguments

x	An object of class 'welo', obtained after running the welofit function
players	A character vector including the players whose rates will be plotted. The indication of the player has to be: 'Surname N.'. For instance, 'Roger Federer' will be included in the 'players' vector as 'Federer R.'
line_width	**optional** Line width, by default it is 1.5
nbreaks	**optional** Number of breaks for y-axis, by default it is 1

Value

A ggplot2 plot

Examples

```
data(atp_2019)
db_clean <- clean(atp_2019, MNM = 5)
res_welo <- welofit(db_clean)
players <- c("Nadal R.", "Djokovic N.", "Federer R.")
rank_plot(res_welo, players, line_width = 1.5)
```

tennis_data *Download tennis data*

Description

Imports ATP or WTA data from the website https://www.tennis-data.co.uk/.

Usage

```
tennis_data(YEAR, Circuit)
```

Arguments

YEAR	Year to consider, in "YYYY" format.
Circuit	Tennis circuit. Valid choices are "ATP" and "WTA".

Details

The structure of the datasets provided by https://www.tennis-data.co.uk/ may vary across years. Therefore, before combining data from multiple seasons, users should verify that the variables are consistent across datasets and, if necessary, retain only the common columns.

An Internet connection is required to download the data. If the external resource is unavailable or the downloaded file cannot be imported, the function returns NULL and provides an informative message.

Value

A data frame containing the data for the selected year and circuit, or NULL if the data cannot be retrieved or imported.

Examples

```
## Not run:
db <- tennis_data("2022", "ATP")
if (!is.null(db)) {
  head(db)
}

## End(Not run)
```

---

tennis_prob *Probability of winning*

Description

Calculates the probability that player i wins over player j for match at time $t+1$ using the WElo or Elo rates at time t. Formally:

$$\hat{p}_{i,j}(t+1) = \frac{1}{1+10^{(E_j(t)-E_i(t))/400}},$$

where $E_i(t)$ and $E_j(t)$ are the WElo or Elo rates at time t.

Usage

```
tennis_prob(i, j)
```

Arguments

i	WElo or Elo rates for player i
j	WElo or Elo rates for player j

Value

Probability that player i wins the match against player j

Examples

```
tennis_prob(2000,2000)
tennis_prob(2500,2000)
```

---

welo_plot *Plot for WElo and Elo rates*

Description

Plots WElo and Elo rates.

Usage

```
welo_plot(x, players, rates = "WElo", SP = 1500, line_width = 1.5)
```

Arguments

x	An object of class 'welo', obtained after running the welofit function
players	A character vector including the players whose rates will be plotted. The indication of the player has to be: 'Surname N.'. For instance, 'Roger Federer' will be included in the 'players' vector as 'Federer R.'
rates	**optional** Rates to be plotted. Valid choices are 'WElo' (by default) and 'Elo'
SP	**optional** Starting points from which the rates originate. By default, SP is 1500
line_width	**optional** Line width, by default it is 1.5

Value

A ggplot2 plot

Examples

```
data(atp_2019)
db_clean <- clean(atp_2019, MNM = 5)
res_welo <- welofit(db_clean)
players <- c("Nadal R.", "Djokovic N.", "Federer R.")
welo_plot(res_welo, players, rates = "WElo",
          SP = 1500, line_width = 1.5)
```

`welofit` *Calculates the WElo and Elo rates*

Description

Calculates the WElo and Elo rates according to Angelini et al. (2022). In particular, the Elo updating system defines the rates (for player i) as:

$$E_i(t+1) = E_i(t) + K_i(t)\left[W_i(t) - \hat{p}_{i,j}(t)\right],$$

where $E_i(t)$ is the Elo rate at time t, $W_i(t)$ is the outcome (1 or 0) for player i in the match at time t, $K_i(t)$ is a scale factor, and $\hat{p}_{i,j}(t)$ is the probability of winning for match at time t, calculated using `tennis_prob`. The scale factor $K_i(t)$ determines how much the rates change over time. By default, according to Kovalchik (2016), it is defined as

$$K_i(t) = 250/\left(N_i(t)+5\right)^{0.4},$$

where $N_i(t)$ is the number of matches disputed by player i up to time t. Alternately, $K_i(t)$ can be multiplied by 1.1 if the match at time t is a Grand Slam match or is played on a given surface. Finally, it can be fixed to a constant value. The WElo rating system is defined as:

$$E_i^*(t+1) = E_i^*(t) + K_i(t)\left[W_i(t) - \hat{p}_{i,j}^*(t)\right] f(W_{i,j}(t)),$$

where $E_i^*(t+1)$ denotes the WElo rate for player i, $\hat{p}_{i,j}^*(t)$ the probability of winning using `tennis_prob` and the WElo rates, and $f(W_{i,j}(t))$ represents a function whose values depend on the games (by default) or sets won in the previous match. In particular, when parameter 'W' is set to "GAMES", $f(W_{i,j}(t))$ is defined as:

$$f(W_{i,j}(t)) \equiv f(G_{i,j}(t)) = \begin{cases} \frac{NG_i(t)}{NG_i(t)+NG_j(t)} & if\ player\ i\ has\ won\ match\ t; \\ \frac{NG_j(t)}{NG_i(t)+NG_j(t)} & if\ player\ i\ has\ lost\ match\ t, \end{cases}$$

where $NG_i(t)$ and $NG_j(t)$ represent the number of games won by player i and player j in match t, respectively. When parameter 'W' is set to "SET", $f(W_{i,j}(t))$ is:

$$f(W_{i,j}(t)) \equiv f(S_{i,j}(t)) = \begin{cases} \frac{NS_i(t)}{NS_i(t)+NS_j(t)} & if\ player\ i\ has\ won\ match\ t; \\ \frac{NS_j(t)}{NS_i(t)+NS_j(t)} & if\ player\ i\ has\ lost\ match\ t, \end{cases}$$

where $NS_i(t)$ and $NS_j(t)$ represent the number of sets won by player i and player j in match t, respectively. The scale factor $K_i(t)$ is the same as the Elo model.

Usage

```
welofit(
  x,
  W = "GAMES",
  SP = 1500,
  K = "Kovalchik",
  CI = FALSE,
  alpha = 0.05,
  B = 1000,
  new_data = NULL
)
```

Arguments

x	Data cleaned through the function clean or, if the parameter 'new_data' is present, a former estimated list coming from the welofit function
W	**optional** Weights to use for the WElo rating system. Valid choices are: "GAMES" (by default) and "SETS"
SP	**optional** Starting points for calculating the rates. 1500 by default
K	**optional** Scale factor determining how much the WElo and Elo rates change over time. Valid choices are: "Kovalchik" (by default), "Grand_Slam", "Surface_Hard", "Surface_Grass", "Surface_Clay" and, finally, a constant value K. The first option ("Kovalchik") is equal to what was suggested by Kovalchik (2016), Putting K to "Grand_Slam" lets the Kovalchik scale factor multiplied by 1.1, if the match is a Grand Slam match. Similarly, the choices "Surface_Hard", "Surface_Grass" and "Surface_Clay" make the Kovalchik scale factor increased by 1.1 if, respectively, the match is played on hard, grass or clay. Finally, K can be any scalar value, indipendently of the number of matches played before the match t
CI	**optional** Confidence intervals for the WElo and Elo rates. Default to FALSE. If 'CI' is set to "TRUE", then the confidence intervals are calculated, according to the procedure explained by Angelini et al. (2022)
alpha	**optional** Significance level of the confidence interval. Default to 0.05
B	**optional** Number of bootstrap samples used to calculate the confidence intervals. Default to 1000
new_data	**optional** New data, cleaned through the function clean, to append to an already estimated set of matches (included in the parameter 'x')

Value

welofit returns an object of class 'welo', which is a list containing the following components:

- results: The data.frame including a variety of variables, among which there are the estimated WElo and Elo rates, before and after the match t, for players i and j, the lower and upper confidence intervals (if CI=TRUE) for the WElo and Elo rates, labelled as '_lb' and '_ub', respectively, and the probability of winning the match for player i (labelled as 'WElo_pi_hat' and 'Elo_pi_hat', respectively, for the WElo and Elo models).
- matches: The number of matches analyzed.
- period: The sample period considered.
- loss: The Brier score (Brier 1950) and log-loss (used by Kovalchik (2016), among others) averages, calculated considering the distance with respect to the outcome of the match.
- highest_welo: The player with the highest WElo rate and the relative date.
- highest_elo: The player with the highest Elo rate and the relative date.
- dataset: The dataset used for the estimation of the WElo and Elo rates.

References

Angelini G, Candila V, De Angelis L (2022). "Weighted Elo rating for tennis match predictions." *European Journal of Operational Research*, **297**(1), 120–132.

Brier GW (1950). "Verification of forecasts expressed in terms of probability." *Monthly weather review*, **78**(1), 1–3.

Kovalchik SA (2016). "Searching for the GOAT of tennis win prediction." *Journal of Quantitative Analysis in Sports*, **12**(3), 127–138.

Examples

```
data(atp_2019)
db_clean<-clean(atp_2019)
res<-welofit(db_clean)
# append new data
db_clean_1<-db_clean[1:500,]
db_clean_2<-db_clean[501:1200,]
res_1<-welofit(db_clean_1)
res_2<-welofit(res_1,new_data=db_clean_2)
```

---

wta_2019 *WTA matches in 2019*

Description

Tennis data for female matches played in 2019. Details can be found on https://www.tennis-data.co.uk/notes.txt

Usage

```
data(wta_2019)
```

Format

An object of class `"data.frame"`.

Source

Tennis archive from https://www.tennis-data.co.uk/

Examples

```
head(wta_2019)
str(wta_2019)
```

Index

∗ **datasets**
atp_2019, 2
wta_2019, 12

atp_2019, 2

betting, 3

clean, 4, *11*

random_betting, 5
rank_plot, 7

tennis_data, 7
tennis_prob, 8, *10*

welo_plot, 9
welofit, *3, 4*, *6, 7*, *9*, 10, *11*
wta_2019, 12