Package 'ecap'

July 23, 2020

Type Package

Title Excess Certainty Adjusted Probability Estimate

Version 0.1.2

Author Bradley Rava, Peter Radchenko, and Gareth M. James

Maintainer Bradley Rava <brava@marshall.usc.edu>

Description Implements the Excess Certainty Adjusted Probability adjustment procedure as described in the paper ``Irrational Exuberance: Correcting Bias in Probability Estimates'' by Gareth James, Peter Radchenko, and Bradley Rava (Journal of the American Statistical Association, 2020; <doi:10.1080/01621459.2020.1787175>). The package includes a function that preforms the ECAP adjustment and a function that estimates the parameters needed for implementing ECAP. For testing and reproducibility, the ESPN and FiveThirtyEight data used in the paper are also included.

License GPL-3

URL <http://faculty.marshall.usc.edu/gareth-james/Research/Probs.pdf>

Depends splines (>= 3.6.0), quadprog (>= 1.5.7), ggplot2, R (>= 3.5.0)

Suggests knitr, rmarkdown, testthat (>= 2.1.0)

VignetteBuilder knitr

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

NeedsCompilation no

Repository CRAN

Date/Publication 2020-07-23 08:10:02 UTC

R topics documented:

Index

---

ecap | *Estimate parameters for the ECAP adjustment*

Description

This function estimates 3 parameters that are needed to preform the ECAP adjustment. As such, it is meant to be used together with the predict.ecap() function included in the "ecap" package. The parameters estimated are the level of corruption and bias in the given unadjusted probability estimates, along with a tuning parameter needed to adjust the level of smoothness in the ECAP estimation.

Usage

```
ecap(
  unadjusted_prob,
  win_var,
  win_id,
  bias_indicator = F,
  lambda_grid = 10^seq(-6, 0, by = 0.5),
  gamma_grid = seq(0.001, 0.05, by = 0.001),
  theta_grid = seq(-4, 2, 0.1)
)
```

Arguments

unadjusted_prob
: Numeric vector of probability estimates that you want estimate the ECAP parameters from.

win_var
: A binary vector of wins and losses that correspond to the probabilities in the unadjusted_prob vector

win_id
: A value that denotes a "win" (or if the event occurred) in the win_var vector.

bias_indicator
: Set this equal to F if you don't want to consider bias in your estimation. Set it equal to T if you do.

lambda_grid
: This is already predefined. However, you can adjust the grid of tuning parameters lambda that ECAP searches over if needed.

gamma_grid
: This is already predefined. However, you can adjust the grid of gamma that ECAP searches over if needed.

theta_grid
: This is already predefined. However, you can adjust the grid of theta that ECAP searches over if needed.

Value

An ecap object that can be used to adjust new probability estimates. It contains all of the tuning parameters needed to calibrate ECAP as well as diagnostic information on the estimate of g. The probabilities used to calibrate ECAP have also been ECAP corrected and are given as part of the output.

Author(s)

Bradley Rava, Peter Radchenko and Gareth M. James.

References

http://faculty.marshall.usc.edu/gareth-james/Research/Probs.pdf

Examples

```
set.seed(1)
p_obs <- runif(1000, 0, 1)
win_var <- rbinom(length(p_obs), 1, p_obs)
ecap_fit <- ecap(unadjusted_prob = p_obs, win_var = win_var, win_id = 1, bias_indicator = FALSE)
```

---

elections_2018	*FiveThirtyEight probabilities from the 2018 election cycle.*

Description

A dataset containing FiveThirtyEight's probability estimates for the 2018 US elections. The races covered seats in the House, Senate, and for Governor. They also used three different methods to compute these estimates. Classic, Deluxe, and Lite. All of these have been included. The probabilities can be viewed from the perspective of a Democrat winning or from the perspective Republican winning any given race. The final results of each race are also included.

Usage

```
elections_2018
```

Format

A data frame with 1518 rows and 1 variables:

cycle the year the race took place

branch what branch of the government the race was for

race the specific position the election was for

forecastdate date of forecast

version type of prediction method FiveThirtyEight used

Democrat_WinProbability probability of a democrat winning the race

Republican_WinProbability probability of a republican winning the race

category category

Democrat_Won binary variable indicating if a democrat won or not

Republican_Won binary variable indicating if a republican won or not

uncalled binary variable indicating if the race was uncalled or not

Source

https://github.com/fivethirtyeight/data/tree/master/forecast-review

---

espn	*ESPN's win probability estimates from the last minute of the NCAA football 2016 and 2017 seasons.*

Description

A dataset containing ESPN's win probability estimates from the last minute of the NCAA football 2016 and 2017 seasons. The time stamps are not perfect, rather the probability that occured closest to the time stamp was recorded from each game. The eventual result of the game is also included in the dataset along with what year it came from.

Usage

```
espn
```

Format

A data frame with 10314 rows and 4 variables:

p.tilde ESPN's win probability estimate.

home_win The eventual result of the game (if the home team won or not.)

time_left The amount of time left in the 4th quarter of the game when p.tilde was estimated.

year The year / season the probability was referring to.

---

plot.ecap   *Plotting ECAP Object*

Description

Plots diagnostic information of an ECAP object. Two plots are produced. The first plot displays the estimate of the function g that the ecap procedure produced. The second compares the unadjusted probability estimates to the ECAP adjusted probability estimates that were used to train the model.

Usage

```
## S3 method for class 'ecap'
plot(x, ...)
```

Arguments

x   An object of class ecap.

...   Additional arguments

Author(s)

Bradley Rava, Peter Radchenko and Gareth M. James.

References

http://faculty.marshall.usc.edu/gareth-james/Research/Probs.pdf

Examples

```
set.seed(1)
p_obs <- runif(1000, 0, 1)
win_var <- rbinom(length(p_obs), 1, p_obs)
ecap_fit <- ecap(unadjusted_prob = p_obs, win_var = win_var, win_id = 1, bias_indicator = FALSE)
plot(ecap_fit)
```

---

predict.ecap   *Implementing the ECAP procedure*

Description

Takes in an ECAP object and a new set of probability estimates that the user wishes to adjust. The model uses the calibration from the ecap object to ECAP adjust the new probability estimates given to the function predict.

Usage

```
## S3 method for class 'ecap'
predict(object, new_unadjusted, ...)
```

Arguments

`object`	An object of class ecap.
`new_unadjusted`	A numerical vector of unadjusted probabilities that you want to ECAP adjust.
`...`	Additional arguments

Value

A vector of ECAP adjusted probability estimates.

Author(s)

Bradley Rava, Peter Radchenko and Gareth M. James.

References

http://faculty.marshall.usc.edu/gareth-james/Research/Probs.pdf

Examples

```
set.seed(1)
p_obs <- runif(1000, 0, 1)
win_var <- rbinom(length(p_obs), 1, p_obs)
ecap_fit <- ecap(unadjusted_prob = p_obs, win_var = win_var, win_id = 1, bias_indicator = FALSE)

p_new <- runif(1000, 0, 1)
ecap_new <- predict(object=ecap_fit, new_unadjusted=p_new)
```

---

print.ecap	*Printing ECAP Object*

Description

Prints summary information about the ECAP object

Usage

```
## S3 method for class 'ecap'
print(x, digits, ...)
```

Arguments

x	An object of class ecap.
digits	The number of significant digits that should be displayed.
...	Additional arguments

Author(s)

Bradley Rava, Peter Radchenko and Gareth M. James.

References

http://faculty.marshall.usc.edu/gareth-james/Research/Probs.pdf

Examples

```
set.seed(1)
p_obs <- runif(1000, 0, 1)
win_var <- rbinom(length(p_obs), 1, p_obs)
ecap_fit <- ecap(unadjusted_prob = p_obs, win_var = win_var, win_id = 1, bias_indicator = TRUE)
print(ecap_fit)
```

---

summary.ecap	*Summary of ECAP Object*

Description

Prints summary information about the calibration of an ECAP object. *** Denotes that one of the parameter estimates has hit the end of the given grid of tuning parameters. The grid can be adjusted in the ecap function.

Usage

```
## S3 method for class 'ecap'
summary(object, digits, ...)
```

Arguments

object	An object of class ecap.
digits	The number of significant digits that should be displayed.
...	Additional arguments

Author(s)

Bradley Rava, Peter Radchenko and Gareth M. James.

References

http://faculty.marshall.usc.edu/gareth-james/Research/Probs.pdf

Examples

```
set.seed(1)
p_obs <- runif(1000, 0, 1)
win_var <- rbinom(length(p_obs), 1, p_obs)
ecap_fit <- ecap(unadjusted_prob = p_obs, win_var = win_var, win_id = 1, bias_indicator = FALSE)
summary(ecap_fit)
```

Index