

# Package ‘HandTill2001’

May 31, 2016

**Type** Package

**Title** Multiple Class Area under ROC Curve

**Version** 0.2-12

**Date** 2016-05-22

**Description** An S4 implementation of Eq. (3) and Eq. (7) by David J. Hand and Robert J. Till (2001) <DOI:10.1023/A:1010920819831>.

**URL** <https://github.com/fvafrCU/HandTill2001>

**Depends** R (>= 2.14), methods, utils

**Suggests** MASS, rpart, mda, nnet

**License** GPL (>= 2)

**NeedsCompilation** no

**Author** Andreas Dominik Cullmann [aut, cre],  
Edgar Kublin [ctb]

**Maintainer** Andreas Dominik Cullmann <r-package\_handtill2001@arcor.de>

**Repository** CRAN

**Date/Publication** 2016-05-31 11:10:11

## R topics documented:

HandTill2001-package . . . . .	2
auc-methods . . . . .	2
bincap . . . . .	4
bincap-class . . . . .	5
cap-class . . . . .	6
ht01.multipleclass . . . . .	6
ht01.twoclass . . . . .	7
multcap . . . . .	9
multcap-class . . . . .	10

<b>Index</b>	<b>11</b>
--------------	-----------

---

HandTill2001-package    *Multiple Class Area under ROC Curve*

---

### Description

A very lean package implementing merely  $M$  given by *Hand and Till (2001)*, Eq. (7).

### Details

$M$  given by *Hand and Till (2001)* defines a multiple class version of the area under curve of the receiver operating characteristic.

### References

*David J. Hand and Robert J. Till (2001). A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems. Machine Learning 45(2), p. 171–186. DOI: 10.1023/A:1010920819831.*

### See Also

`help(package="HandTill2001")`, especially "`methods?auc`"; various packages that calculate two class AUC ("`ROCR`") or multiple class AUC (pROC, "`caTools`").

### Examples

```
library(HandTill2001)
data(ht01.multipleclass)
auc(
  multcap(
    response = ht01.multipleclass$observed
    , predicted = as.matrix(ht01.multipleclass[, levels(ht01.multipleclass$observed)])
  )
)
```

---

auc-methods

*Methods for Function auc in Package HandTill2001*

---

### Description

Calculate area under curve of the receiver operating characteristic for two or more prediction classes.

### Details

Depending on whether object is of class "`bincap`" or of class "`multcap`", a two class or multiple class AUC is calculated.

**Value**

An object of class "numeric".

**Methods**

signature(object = "bincap") calculates the AUC statistic for a two class response following *Hand and Till (2001)*, Eq. (3).

signature(object = "multcap") calculates the AUC statistic for a multiple class response following *Hand and Till (2001)*, Eq. (7).

**References**

*David J. Hand and Robert J. Till (2001). A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems. Machine Learning 45(2), p. 171–186. DOI: 10.1023/A:1010920819831.*

**See Also**

["class?bincap"](#), ["class?multcap"](#)

**Examples**

```
library(HandTill2001)
data(ht01.twoclass)
data(ht01.multipleclass)
message(" == AUC for a two class response")
## Not run:
message(" == == ROCR result:")
library(ROCR)
performance(prediction(labels=ht01.twoclass$observed
                      , predictions=ht01.twoclass$predicted
                      )
            , measure = "auc")

## End(Not run)
message(" == == HandTill2001 result:")
auc(bincap(
  response = as.factor(ht01.twoclass$observed)
  , predicted = ht01.twoclass$predicted
  , true = "1"
))
message(" == AUC for a multiple class response")
auc(multcap(
  response = ht01.multipleclass$observed
  , predicted = as.matrix(ht01.multipleclass[, levels(ht01.multipleclass$observed)])
))
```

---

**bincap***a ui-constructor for Class "bincap" in Package **HandTill2001***

---

**Description**

bincap(...) is an alias to new("bincap", ...).

**Usage**

```
bincap(response, predicted, true = "1")
```

**Arguments**

response	Object of class "factor".
predicted	Object of class "numeric".
true	Object of class "character".

**Details**

There is no casting or conversion of data. bincap(...) is just an alias to new("bincap", ...).

**Value**

An object of class "bincap"

**Author(s)**

Andreas Dominik Cullmann

**See Also**

["class?bincap"](#)

**Examples**

```
library(HandTill2001)
data(ht01.twoclass)
str(ht01.twoclass$observed)
message("note that ht01.twoclass$observed is not a factor; we have to convert it.")
bincap(
  response = as.factor(ht01.twoclass$observed)
  , predicted = ht01.twoclass$predicted
  , true = c("1")
)
```

---

`bincap-class`*Class "bincap" in Package **HandTill2001***

---

### Description

S4 class for a two class response and corresponding (predicted) probabilities.

### Objects from the Class

Objects can be created by calls of the form `new("bincap", ...)`. They are used to store a two class response (one of the two levels of which is supposed to be true), the information which of the two levels of the two class response is thought of as 'true'/'positive'/'present' (the other one would then be thought of as 'false'/'negative'/'absence') and the predicted probabilities that response is true.

### Slots

`predicted`: Object of class "numeric", probabilities for response. Of length  $n$

`true`: Object of class "character", indicating which of the two *levels* of response is to be treated as 'true'/'positive'/'presence' (the other one would then accordingly be called 'false'/'negative'/'absence').  
Of length 1.

`response`: Object of class "factor", two class observations. Of length  $n$ .

### Extends

Class "`cap`", directly.

### Methods

`auc` signature(object = "bincap"): ...

### Note

No defaults are set. Especially, you have to explicitly initialize `true`, there is no trying to guess it from the levels of response.

### Author(s)

Andreas Dominik Cullmann

### See Also

`"class?cap"`, `"class?multcap"`, `"?bincap"`

### Examples

```
showClass("bincap")
```

cap-class

*Class "cap" in Package **HandTill2001***

---

**Description**

A virtual class for "bincap" and "multcap".

**Objects from the Class**

A virtual Class: No objects may be created from it.

**Slots**

response: Object of class "factor", typically two class or multiple class observations.

**Methods**

No methods defined with class "cap" in the signature.

**Author(s)**

Andreas Dominik Cullmann

**See Also**

["class?bincap"](#), ["class?multcap"](#)

---

ht01.multipleclass*ht01.multipleclass data in Package **HandTill2001***

---

**Description**

multiple class data and probability predictions thereof.

**Usage**

data(ht01.multipleclass)

**Format**

A data frame with 214 observations on the following 7 variables.

observed a factor with levels Con Head Tabl Veh WinF WinNF

WinF a numeric vector

WinNF a numeric vector

Veh a numeric vector

Con a numeric vector

Tabl a numeric vector

Head a numeric vector

**Details**

multiple class data ('observed': MASS::fgl\$type) and probability predictions (predict(fgl.rp4), cf. Venables and Ripley (2002), p. 264 and 'Source') from rpart::rpart.

**Source**

```
## From: Forensic glass example Venables and Ripley (2002) pp. 261--265
library(MASS);library(rpart);data(fgl);set.seed(123)
fgl.rp4 <- rpart(type ~ ., data = fgl, cp = 0.03
                , parms = list(split = "information"))
ht01.multipleclass <- data.frame(observed = fgl$type, predict(fgl.rp4))
write.table(ht01.multipleclass, file = "ht01.multipleclass.txt")
```

**References**

Venables, W. N and Ripley, B. D. (2002), *Modern Applied Statistics with S* (4th edition). Springer, ISBN 0-387-95457-0

**Examples**

```
library(HandTill2001)
data(ht01.multipleclass)
str(ht01.multipleclass)
```

---

ht01.twoclass

ht01.twoclass data in Package **HandTill2001**

---

**Description**

two class data and probability predictions thereof.

**Usage**

```
data(ht01.twoclass)
```

**Format**

A data frame with 189 observations on the following 2 variables.

observed a numeric vector

predicted a numeric vector

**Details**

two class data ('observed': MASS::birthwt\$low) and probability predictions

(predict(birthwt.step2, type = "response"), cf. Venables and Ripley (2002), pp. 195f and 'Source') from stats::glm.

**Source**

```
## From: A two class data example Venables and Ripley pp. 194--199
library(MASS); data("birthwt"); attach(birthwt)
race <- (factor(race, labels = c("white", "black", "other")))
ptd <- factor(ptl > 0)
ftv <- factor(ftv)
levels(ftv)[-1:2] <- "2+"
bwt <- data.frame(low = factor(low), age, lwt, race, smoke = (smoke > 0)
, ptd, ht = (ht > 0), ui = (ui > 0), ftv)
detach(birthwt)
birthwt.glm <- glm(low ~ ., family=binomial(link=logit), data=bwt)
birthwt.step2 <- stepAIC(birthwt.glm, ~ .^2
+ I(scale(age)^2) + I(scale(lwt)^2), trace = F )
ht01.twoclass <- data.frame(observed = bwt$low
, predicted = predict(birthwt.step2
, type = "response"))
write.table(ht01.twoclass, file = "ht01.twoclass.txt")
```

**References**

Venables, W. N and Ripley, B. D. (2002), *Modern Applied Statistics with S* (4th edition). Springer, ISBN 0-387-95457-0

**Examples**

```
library(HandTill2001)
data(ht01.twoclass)
str(ht01.twoclass)
```



---

`multcap`*a ui-constructor for Class "multcap" in Package **HandTill2001***

---

### Description

`multcap(...)` is an alias to `new("multcap", ...)`.

### Usage

```
multcap(response, predicted)
```

### Arguments

<code>response</code>	Object of class "factor".
<code>predicted</code>	Object of class "matrix".

### Details

There is no casting or conversion of data. `multcap(...)` is just an alias to `new("multcap", ...)`.

### Value

An object of class "bincap"

### Author(s)

Andreas Dominik Cullmann

### See Also

["class?multcap"](#)

### Examples

```
library(HandTill2001)
data(ht01.multipleclass)
str(ht01.multipleclass$observed)
message("note that ht01.multipleclass$observed is a factor; we do not have to convert it.")
multcap(
  response = ht01.multipleclass$observed
  , predicted = as.matrix(ht01.multipleclass[, levels(ht01.multipleclass$observed)])
)
```

---

`multcap-class`*Class "multcap" in Package **HandTill2001***

---

**Description**

S4 class for a multiple class response and corresponding (predicted) probabilities.

**Objects from the Class**

Objects can be created by calls of the form `new("multcap", ...)`. They are used to store a multiple class response and the predicted probabilities for each of the `levels(response)`.

**Slots**

**predicted:** Object of class "matrix", probabilities for response. Of dimension  $(n, \text{length}(\text{unique}(\text{response})))$ .

The columns (`dimnames()[[2]]`) of the prediction matrix have to be named with the values of `unique(response)`. This is the default for example with `predict.rpart(type="prob", ...)`.

**response:** Object of class "factor", multiple class observations. Of length  $n$ .

**Extends**

Class "`cap`", directly.

**Methods**

**auc** signature(object = "multcap"): ...

**Author(s)**

Andreas Dominik Cullmann

**See Also**

`"class?cap"`, `"class?bincap"`, `"?multcap"`

**Examples**

```
showClass("multcap")
```

# Index

- \*Topic **AUC, ROC**
  - HandTill2001-package, [2](#)
- \*Topic **auc**
  - auc-methods, [2](#)
- \*Topic **classes**
  - bincap-class, [5](#)
  - cap-class, [6](#)
  - multcap-class, [10](#)
- \*Topic **datasets**
  - ht01.multipleclass, [6](#)
  - ht01.twoclass, [7](#)
- \*Topic **methods**
  - auc-methods, [2](#)
- \*Topic **ui-constructor**
  - bincap, [4](#)
  - multcap, [9](#)
- ?bincap, [5](#)
- ?multcap, [10](#)
  
- auc (auc-methods), [2](#)
- auc, bincap-method (auc-methods), [2](#)
- auc, multcap-method (auc-methods), [2](#)
- auc-methods, [2](#)
  
- bincap, [4](#)
- bincap-class, [5](#)
  
- cap, [5](#), [10](#)
- cap-class, [6](#)
- caTools, [2](#)
- class?bincap, [3](#), [4](#), [6](#), [10](#)
- class?cap, [5](#), [10](#)
- class?multcap, [3](#), [5](#), [6](#), [9](#)
  
- HandTill2001 (HandTill2001-package), [2](#)
- HandTill2001-package, [2](#)
- ht01.multipleclass, [6](#)
- ht01.twoclass, [7](#)
  
- methods?auc, [2](#)
- multcap, [9](#)
  
- multcap-class, [10](#)
- ROCR, [2](#)