Package ‘MRMCbinary’

July 21, 2025

Title Multi-Reader Multi-Case Analysis of Binary Diagnostic Tests

Version 1.0.5

Description The goal of 'MRMCbinary' is to compare the performance of diagnostic tests (i.e., sensitivity and specificity) for binary outcomes in multi-reader multi-case (MRMC) studies. It is based on conditional logistic regression and Cochran’s Q test (or McNemar’s test when the number of modalities is equal to 2).

License MIT + file LICENSE

URL `https://github.com/seungjae2525/MRMCbinary`

BugReports `https://github.com/seungjae2525/MRMCbinary/issues`

Depends R (>= 4.2.0)

Imports DescTools, stats, survival (>= 3.8-1)

Encoding UTF-8

RoxygenNote 7.3.2

LazyData true

NeedsCompilation no

Author Seungjae Lee [aut, cre] (ORCID: <`https://orcid.org/0000-0001-5508-8634`>), Woojoo Lee [aut] (ORCID: <`https://orcid.org/0000-0001-7447-7045`>)

Maintainer Seungjae Lee <`seungjae2525@gmail.com`>

Repository CRAN

Date/Publication 2025-02-20 08:20:17 UTC

Contents

`MRMCbinary-package` *MRMCbinary: Multi-reader multi-case analysis of binary diagnostic tests*

Description

R package **MRMCbinary** is a package aimed at comparing the performance of diagnostic tests (i.e., sensitivity and specificity) for binary outcomes in multi-reader multi-case (MRMC) studies.

Author(s)

Seungjae Lee <`seungjae2525@gmail.com`> and Woojoo Lee <`lwj221@gmail.com`>

References

Lee, S., Jang, S., and Lee, W. (2025). Evaluating Diagnostic Accuracy of Binary Medical Tests in Multi-reader Multi-case Study.

See Also

Useful links:

- `https://github.com/seungjae2525/MRMCbinary`
- Report bugs at `https://github.com/seungjae2525/MRMCbinary/issues`

`MRMCbinary` *Multi-reader multi-case analysis of binary diagnostic tests*

Description

`MRMCbinary()` is the main function of `MRMCbinary` package and can be used to compare sensitivity and specificity of diagnostic tests for binary outcomes in multi-reader multi-case (MRMC) studies.

Usage

```
MRMCbinary(
  data,
  Modality,
  Reader,
  Case,
  D,
  Y,
  measure,
  effect,
  interaction = NULL,
  reference.Modality = NULL,
  reference.Reader = NULL
)
```

Arguments

`data`	A data frame in which contains the modality identifiers (`Modality`), the reader identifiers (`Reader`), the case identifiers (`Case`), the true disease status (`D`), and the binary diagnostic test result (`Y`).
`Modality`	Variable of the modality identifiers.
`Reader`	Variable of the reader identifiers.
`Case`	Variable of the case identifiers.
`D`	Variable of the true disease status. It should be set the value to 1 for cases diseased and to 0 for those non-diseased.
`Y`	Variable of the binary diagnostic test result. It should be set the value to 1 for cases diagnosed as positive and to 0 for those diagnosed as negative.
`measure`	Diagnostic accuracy measure (one of `"All"`, `"Sensitivity"`, and `"Specificity"`).
`effect`	Effect to compare sensitivity and specificity (one of `"Modality"`, `"Reader"`, and `"Both"`). See **Details**.
`interaction`	When evaluating the interaction effect between modality and reader, `interaction = TRUE`, otherwise `interaction = FALSE`. Specify only when `effect` is `"Both"`. Default: `NULL`. See **Details**.
`reference.Modality`	Reference in variable of the modality identifiers.
`reference.Reader`	Reference in variable of the reader identifiers.

Details

There are three effects that can be evaluated:

- `effect = "Modality"`: This is used when the goal is to exclusively evaluate the effects of multiple modalities. And, Cochran's Q test (when the number of modalities is greater than 2) or McNemar's test (when the number of modalities is equal to 2) result is reported. When `effect = "Modality"`, `interaction` must be set to NULL.
- `effect = "Reader"`: This is used when the goal is to exclusively evaluate the effects of multiple readers. And, Cochran's Q test (when the number of modalities is greater than 2) or McNemar's test (when the number of modalities is equal to 2) result is reported. When `effect = "Reader"`, `interaction` must be set to NULL.
- `effect = "Both"`: This is used when the goal is to simultaneously evaluate the effects of multiple modalities and multiple readers. In this case, `interaction` must be specified (TRUE or FALSE). If one want to evaluate the interaction effect between modality and reader in the conditional logistic regression, set `interaction = TRUE`, otherwise `interaction = FALSE`. When `interaction = TRUE`, Cochran's Q test result is reported. However, when `interaction = FALSE`, Cochran's Q test or McNemar's test result is not reported.

See Lee et al. (2025) for details.

Value

An object of class `MRMCbinary`. The object is a data.frame with the following components:

`CLR_sen`	Conditional logistic regression results for sensitivity.
`CLR_LRT_sen`	Likelihood ratio test from the conditional logistic regression results for sensitivity.
`CLR_Score_sen`	Score test from the conditional logistic regression results for sensitivity.
`CLR_Wald_sen`	Wald test from the conditional logistic regression results for sensitivity.
`Q_MN_sen`	Cochran's Q test (when the number of modalities is greater than 2) or McNemar's test (when the number of modalities is equal to 2) result for sensitivity. This is only reported if (1) `effect = "Modality"`, (2) `effect = "Reader"`, or (3) `effect = "Both"` and `interaction = TRUE`.
`CLR_spe`	Conditional logistic regression results for specificity.
`CLR_LRT_spe`	Likelihood ratio test from the conditional logistic regression results for specificity.
`CLR_Score_spe`	Score test from the conditional logistic regression results for specificity.
`CLR_Wald_spe`	Wald test from the conditional logistic regression results for specificity.
`Q_MN_spe`	Cochran's Q test (when the number of modalities is greater than 2) or McNemar's test (when the number of modalities is equal to 2) result for specificity. This is only reported if (1) `effect = "Modality"`, (2) `effect = "Reader"`, or (3) `effect = "Both"` and `interaction = TRUE`.
`formula`	Formula used in the conditional logistic regression.
`args`	List of arguments used in the `MRMCbinary` function.
`n.modality`	Total number of modalities.
`n.reader`	Total number of readers.
`n.case`	Total number of cases.
`effect`	Effect to compare sensitivity and specificity.
`measure`	Diagnostic accuracy measure.
`interaction`	This is only included in the `MRMCbinary` object when `effect = "Both"`. If one want to evaluate the interaction effect between modality and reader in the conditional logistic regression, `interaction = TRUE`, otherwise `interaction = FALSE`.
`reference.Modality`	Reference in variable of the modality identifiers.
`reference.Reader`	Reference in variable of the reader identifiers.
`n.diseased`	The number of diseased cases. If `measure = "Specificity"`, then n.diseased is NULL.
`n.nondiseased`	The number of non-diseased cases. If `measure = "Sensitivity"`, then n.nondiseased is NULL.
`n.pos.diseased`	The number of test positive cases among diseased cases. If `measure = "Specificity"`, then n.pos.diseased is NULL.

`n.pos.nondiseased`

The number of test positive cases among non-diseased cases. If `measure = "Sensitivity"`, then n.pos.nondiseased is NULL.

The results for the `MRMCbinary` are printed with the `print.MRMCbinary` function. Also, the results for the `MRMCbinary` are summarized with the `summary.MRMCbinary` function.

References

Lee, S., Jang, S., and Lee, W. Evaluating Diagnostic Accuracy of Binary Medical Tests in Multi-reader Multi-case Study.

See Also

`print.MRMCbinary`, `summary.MRMCbinary`

Examples

```
## Load example data
data(VanDyke)

## Return the first parts of an object
head(VanDyke)

## See unique readers
unique(VanDyke$reader)

## See unique modalities
unique(VanDyke$treatment)

## Create binary test results (Y_ijk)
VanDyke$Y <- as.numeric(VanDyke$rating >= 3)

## Example usage of MRMCbinary function:
# When comparing the sensitivities and specificities between modalities
modality_result <- MRMCbinary(data = VanDyke, Modality = treatment, Reader = reader,
                              Case = case, D = truth, Y = Y, measure = "All",
                              effect = "Modality", interaction = NULL,
                              reference.Modality = "1", reference.Reader = NULL)

# When comparing the sensitivities and specificities between readers
reader_result <- MRMCbinary(data = VanDyke, Modality = treatment, Reader = reader,
                            Case = case, D = truth, Y = Y, measure = "All",
                            effect = "Reader", interaction = NULL,
                            reference.Modality = NULL, reference.Reader = "1")

# When comparing the sensitivities and specificities
#  between modalities and between readers together
#  not considering interaction between modalities and readers
both_result_wo_int <- MRMCbinary(data = VanDyke, Modality = treatment, Reader = reader,
                                 Case = case, D = truth, Y = Y, measure = "All",
                                 effect = "Both", interaction = FALSE,
                                 reference.Modality = "1", reference.Reader = "1")
```

```
# When comparing the sensitivities and specificities
#  between modalities and between readers together
#  considering interaction between modalities and readers
both_result_with_int <- MRMCbinary(data = VanDyke, Modality = treatment, Reader = reader,
                                   Case = case, D = truth, Y = Y, measure = "All",
                                   effect = "Both", interaction = TRUE,
                                   reference.Modality = "1", reference.Reader = "1")
```

---

print.MRMCbinary *Print for* `MRMCbinary` *objects*

Description

Print the results for object of class `MRMCbinary`.

Usage

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

Arguments

`x`	An object for class `MRMCbinary`.
`...`	Further arguments (currently not used).

Details

Print the results for object of class `MRMCbinary`. From the conditional logistic regression results, "Estimate" corresponds to the log odds ratio and "SE" corresponds to the standard error of the log odds ratio.

Value

No return value, called for side effects.

See Also

`MRMCbinary`, `summary.MRMCbinary`, `print`

Examples

```
## Load example data
data(VanDyke)

## Return the first parts of an object
head(VanDyke)
```

```
## Extract unique modalities
unique(VanDyke$treatment)

## Extract Unique readers
unique(VanDyke$reader)

## Create binary test results (Y_ijk)
VanDyke$Y <- as.numeric(VanDyke$rating >= 3)

## Example usage of MRMCbinary function:
# When comparing the sensitivities and specificities between modalities
modality_result <- MRMCbinary(data = VanDyke, Modality = treatment, Reader = reader,
                              Case = case, D = truth, Y = Y, measure = "All",
                              effect = "Modality", interaction = NULL,
                              reference.Modality = "1", reference.Reader = NULL)
print(modality_result)

# When comparing the sensitivities and specificities between readers
reader_result <- MRMCbinary(data = VanDyke, Modality = treatment, Reader = reader,
                            Case = case, D = truth, Y = Y, measure = "All",
                            effect = "Reader", interaction = NULL,
                            reference.Modality = NULL, reference.Reader = "1")
print(reader_result)

# When comparing the sensitivities and specificities
#  between modalities and between readers together
#  not considering interaction between modalities and readers
both_result_wo_int <- MRMCbinary(data = VanDyke, Modality = treatment, Reader = reader,
                                 Case = case, D = truth, Y = Y, measure = "All",
                                 effect = "Both", interaction = FALSE,
                                 reference.Modality = "1", reference.Reader = "1")
print(both_result_wo_int)

# When comparing the sensitivities and specificities
#  between modalities and between readers together
#  considering interaction between modalities and readers
both_result_with_int <- MRMCbinary(data = VanDyke, Modality = treatment, Reader = reader,
                                   Case = case, D = truth, Y = Y, measure = "All",
                                   effect = "Both", interaction = TRUE,
                                   reference.Modality = "1", reference.Reader = "1")
print(both_result_with_int)
```

`print.SensSpec` *Print for* `SensSpec` *objects*

Description

Print the results for object of class `SensSpec`.

Usage

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

Arguments

`x`	An object for class `SensSpec`.
`...`	Further arguments (currently not used).

Details

Print the results for object of class `SensSpec`.

Value

No return value, called for side effects.

See Also

`SensSpec`, `print`

Examples

```
## Load example data
data(VanDyke)

## Return the first parts of an object
head(VanDyke)

## Extract unique modalities
unique(VanDyke$treatment)

## Extract Unique readers
unique(VanDyke$reader)

## Create binary test results (Y_ijk)
VanDyke$Y <- as.numeric(VanDyke$rating >= 3)

## Example usage of SensSpec function:
senspe_result1 <- SensSpec(data = VanDyke, Modality = treatment,
                           Reader = reader, Case = case,
                           D = truth, Y = Y, percentage = FALSE, digits = 3)
print(senspe_result1)

# Report results as percentage points
senspe_result2 <- SensSpec(data = VanDyke, Modality = treatment,
                           Reader = reader, Case = case,
                           D = truth, Y = Y, percentage = TRUE, digits = 1)
print(senspe_result2)
```

`SensSpec` *Calculate sensitivity and specificity*

Description

`SensSpec()` is the function that calculates overall sensitivity and specificity, modality-specific sensitivity and specificity, and modality- and reader-specific sensitivity and specificity.

Usage

```
SensSpec(
  data,
  Modality,
  Reader,
  Case,
  D,
  Y,
  percentage = FALSE,
  digits = max(1L, getOption("digits") - 3L)
)
```

Arguments

`data`	A data frame in which contains the modality identifiers (`Modality`), the reader identifiers (`Reader`), the case identifiers (`Case`), the true disease status (`D`), and the binary diagnostic test result (`Y`).
`Modality`	Variable of the modality identifiers.
`Reader`	Variable of the reader identifiers.
`Case`	Variable of the case identifiers.
`D`	Variable of the true disease status. It should be set the value to 1 for cases diseased and to 0 for those non-diseased.
`Y`	Variable of the binary diagnostic test result. It should be set the value to 1 for cases diagnosed as positive and to 0 for those diagnosed as negative.
`percentage`	Whether to report results as decimals or percentage points. Default: `FALSE`.
`digits`	Number of significant digits. Default: `max(1L, getOption("digits") - 3L)`.

Value

An object of class `SensSpec`. The object is a data.frame with the following components:

`Overall Result`	Overall sensitivity and specificity
`Modality-specific Result`	Modality-specific sensitivity and specificity
`Reader-specific Modality-specific Result`	Modality- and reader-specific sensitivity and specificity

`digits`	The number of significant digits

The results for the SensSpec are printed with the `print.SensSpec` function.

References

Yerushalmy, J. (1947). Statistical Problems in Assessing Methods of Medical Diagnosis, with Special Reference to X-Ray Techniques. *Public Health Reports (1896-1970)*, 62(40), 1432–1449.

See Also

`print.SensSpec`

Examples

```
## Load example data
data(VanDyke)

## Return the first parts of an object
head(VanDyke)

## Extract unique modalities
unique(VanDyke$treatment)

## Extract Unique readers
unique(VanDyke$reader)

## Create binary test results (Y_ijk)
VanDyke$Y <- as.numeric(VanDyke$rating >= 3)

## Example usage of SensSpec function:
# Report results as decimals
senspe_result1 <- SensSpec(data = VanDyke, Modality = treatment,
                           Reader = reader, Case = case,
                           D = truth, Y = Y, percentage = FALSE, digits = 3)

# Report results as percentage points
senspe_result2 <- SensSpec(data = VanDyke, Modality = treatment,
                           Reader = reader, Case = case,
                           D = truth, Y = Y, percentage = TRUE, digits = 1)
```

`summary.MRMCbinary` *Summary for* `MRMCbinary` *objects*

Description

Summary the results for object of class `MRMCbinary`.

Usage

```
## S3 method for class 'MRMCbinary'
summary(object, digits = max(1L, getOption("digits") - 3L), ...)
```

Arguments

`object`	An object for class `MRMCbinary`.
`digits`	Number of significant digits. Default: `max(1L, getOption("digits") - 3L)`.
`...`	Further arguments (currently not used).

Details

Summary the results for object of class `MRMCbinary`. From the conditional logistic regression results, the odds ratio, confidence interval of the odds ratio, and P value are reported.

Value

No return value, called for side effects.

See Also

`MRMCbinary`, `print.MRMCbinary`, `summary`

Examples

```
## Load example data
data(VanDyke)

## Return the first parts of an object
head(VanDyke)

## Extract unique modalities
unique(VanDyke$treatment)

## Extract Unique readers
unique(VanDyke$reader)

## Create binary test results (Y_ijk)
VanDyke$Y <- as.numeric(VanDyke$rating >= 3)

## Example usage of MRMCbinary function:
# When comparing the sensitivities and specificities between modalities
modality_result <- MRMCbinary(data = VanDyke, Modality = treatment, Reader = reader,
                              Case = case, D = truth, Y = Y, measure = "All",
                              effect = "Modality", interaction = NULL,
                              reference.Modality = "1", reference.Reader = NULL)
summary(modality_result, digits = 3)

# When comparing the sensitivities and specificities between readers
reader_result <- MRMCbinary(data = VanDyke, Modality = treatment, Reader = reader,
                            Case = case, D = truth, Y = Y, measure = "All",
```

```
                            effect = "Reader", interaction = NULL,
                            reference.Modality = NULL, reference.Reader = "1")
summary(reader_result, digits = 3)

# When comparing the sensitivities and specificities
#  between modalities and between readers together
#  not considering interaction between modalities and readers
both_result_wo_int <- MRMCbinary(data = VanDyke, Modality = treatment, Reader = reader,
                                 Case = case, D = truth, Y = Y, measure = "All",
                                 effect = "Both", interaction = FALSE,
                                 reference.Modality = "1", reference.Reader = "1")
summary(both_result_wo_int, digits = 3)

# When comparing the sensitivities and specificities
#  between modalities and between readers together
#  considering interaction between modalities and readers
both_result_with_int <- MRMCbinary(data = VanDyke, Modality = treatment, Reader = reader,
                                   Case = case, D = truth, Y = Y, measure = "All",
                                   effect = "Both", interaction = TRUE,
                                   reference.Modality = "1", reference.Reader = "1")
summary(both_result_with_int, digits = 3)
```

`VanDyke` *Multi-reader multi-case dataset*

Description

Example data from a study comparing the relative performance of cinematic presentation of magnetic resonance imaging (CINE MRI) to single spin-echo magnetic resonance imaging (SE MRI) for the detection of thoracic aortic dissection (Van Dyke et al., 1993).

Usage

```
VanDyke
```

Format

A data frame with 1140 rows and 7 variables:

reader Reader identifier for the five radiologists

treatment Treatment identifier for the two imaging modalities

case Case identifiers for 114 cases

case2 Case identifier (cases nested within readers)

case3 Case identifier (cases nested within treatments)

truth Indicator for thoracic aortic dissection (i.e., true disease status): 1 = performed (i.e., patients with aortic dissection imaged with both SE MRI and CINE MRI) or 0 = not performed (i.e., patients without dissection imaged with both SE MRI and CINE MRI)

rating Five-point ratings given to case images by the radiologists (i.e., diagnostic test result): 1 = definitely no aortic dissection, 2 = probably no aortic dissection, 3 = unsure about aortic dissection, 4 = probably aortic dissection, or 5 = definitely aortic dissection

Details

This example compares the relative performance of SE MRI with the CINE MRI in detecting thoracic aortic dissection. There are 45 patients with an aortic dissection and 69 patients without a dissection imaged with both SE MRI and CINE MRI. One can directly use this data from `MRMCaov` package. See **Source**.

Source

This data are available at `https://perception.lab.uiowa.edu` and `https://github.com/brian-j-smith/MRMCaov/tree/master/data`.

References

Van Dyke, C. W., White, R. D., Obuchowski, N. A., Geisinger, M. A., Lorig, R. J., & Meziane, M. A. (1993). Cine MRI in the diagnosis of thoracic aortic dissection. 79th RSNA Meetings. *Chicago, IL*, 28.

Examples

```
## Load example data
data(VanDyke)

## Return the first parts of an object
head(VanDyke)

## Extract unique modalities
unique(VanDyke$treatment)

## Extract Unique readers
unique(VanDyke$reader)

## Create binary test results (Y_ijk)
VanDyke$Y <- as.numeric(VanDyke$rating >= 3)
```

Index