

Package ‘PTSDdiag’

February 13, 2026

Type Package

Title Optimize PTSD Diagnostic Criteria

Version 0.1.0

Description Provides tools for analyzing and optimizing PTSD (Post-Traumatic Stress Disorder) diagnostic criteria using PCL-5 (PTSD Checklist for DSM-5) data. Functions identify optimal subsets of PCL-5 items that maintain diagnostic accuracy while reducing assessment burden. Includes tools for both hierarchical (cluster-based) and non-hierarchical symptom combinations, calculation of diagnostic metrics, and comparison with standard DSM-5 criteria. Model validation is conducted using holdout and cross-validation methods to assess robustness and generalizability of the results. For more details see Weidmann et al. (2025) <[doi:10.31219/osf.io/6rk72_v1](https://doi.org/10.31219/osf.io/6rk72_v1)>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

Imports dplyr, magrittr, rlang, stats, utils, modelr

Depends R (>= 3.5.0)

Suggests DT, knitr, lattice, psych, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

URL <https://github.com/WeidmannL/PTSDdiag>

BugReports <https://github.com/WeidmannL/PTSDdiag/issues>

NeedsCompilation no

Author Laura Weidmann [aut] (ORCID: <<https://orcid.org/0009-0006-5509-6961>>),
Tobias R. Spiller [aut, cre] (ORCID:
<<https://orcid.org/0000-0002-0107-0743>>),
Flavio A. Schüepp [aut] (ORCID:
<<https://orcid.org/0000-0002-0750-3888>>)

Maintainer Tobias R. Spiller <tobias.spiller@access.uzh.ch>

Repository CRAN
Date/Publication 2026-02-13 07:50:02 UTC

Contents

analyze_best_six_symptoms_four_required	2
analyze_best_six_symptoms_four_required_clusters	4
binarize_data	6
calculate_ptsd_total	7
create_ptsd_diagnosis_binarized	8
create_ptsd_diagnosis_nonbinarized	9
create_readable_summary	11
cross_validation	12
holdout_validation	13
rename_ptsd_columns	15
simulated_ptsd	16
summarize_ptsd	18
summarize_ptsd_changes	19
Index	21

analyze_best_six_symptoms_four_required

Find optimal non-hierarchical six-symptom combinations for PTSD diagnosis

Description

Identifies the three best six-symptom combinations for PTSD diagnosis where any four symptoms must be present, regardless of their cluster membership. This function implements a simplified diagnostic approach compared to the full DSM-5 criteria.

Usage

```
analyze_best_six_symptoms_four_required(data, score_by = "false_cases")
```

Arguments

- | | |
|----------|--|
| data | A dataframe containing exactly 20 columns with PCL-5 item scores (output of rename_ptsd_columns). Each symptom should be scored on a 0-4 scale where: <ul style="list-style-type: none">• 0 = Not at all• 1 = A little bit• 2 = Moderately• 3 = Quite a bit• 4 = Extremely |
| score_by | Character string specifying optimization criterion: |

- "false_cases": Minimize total misclassifications
- "newly_nondiagnosed": Minimize false negatives only

Details

The function:

1. Tests all possible combinations of 6 symptoms from the 20 PCL-5 items
2. Requires 4 symptoms to be present (≥ 2 on original 0-4 scale) for diagnosis
3. Identifies the three combinations that best match the original DSM-5 diagnosis

Optimization can be based on either:

- Minimizing false cases (both false positives and false negatives)
- Minimizing only false negatives (newly non-diagnosed cases)

The symptom clusters in PCL-5 are:

- Items 1-5: Intrusion symptoms (Criterion B)
- Items 6-7: Avoidance symptoms (Criterion C)
- Items 8-14: Negative alterations in cognitions and mood (Criterion D)
- Items 15-20: Alterations in arousal and reactivity (Criterion E)

Value

A list containing:

- best_symptoms: List of three vectors, each containing six symptom numbers representing the best combinations found
- diagnosis_comparison: Dataframe comparing original DSM-5 diagnosis with diagnoses based on the three best combinations
- summary: Interactive datatable (DT) showing diagnostic accuracy metrics for each combination

Examples

```
# Create example data
ptsd_data <- data.frame(matrix(sample(0:4, 200, replace=TRUE), ncol=20))
names(ptsd_data) <- paste0("symptom_", 1:20)

# Find best combinations minimizing false cases
results <- analyze_best_six_symptoms_four_required(ptsd_data, score_by = "false_cases")

# Get symptom numbers
results$best_symptoms

# View raw comparison data
results$diagnosis_comparison
```

```
# View summary statistics
results$summary
```

```
analyze_best_six_symptoms_four_required_clusters
```

Find optimal hierarchical six-symptom combinations for PTSD diagnosis

Description

Identifies the three best six-symptom combinations for PTSD diagnosis where four symptoms must be present and must include at least one symptom from each DSM-5 criterion cluster. This approach maintains the hierarchical structure of PTSD diagnosis while reducing the total number of required symptoms.

Usage

```
analyze_best_six_symptoms_four_required_clusters(
  data,
  score_by = "false_cases"
)
```

Arguments

data	A dataframe containing exactly 20 columns with PCL-5 item scores (output of <code>rename_ptsd_columns</code>). Each symptom should be scored on a 0-4 scale where: • 0 = Not at all • 1 = A little bit • 2 = Moderately • 3 = Quite a bit • 4 = Extremely
score_by	Character string specifying optimization criterion: • "false_cases": Minimize total misclassifications • "newly_nondiagnosed": Minimize false negatives only

Details

The function:

1. Generates valid combinations ensuring representation from all clusters
2. Requires 4 symptoms to be present (≥ 2 on original 0-4 scale) for diagnosis
3. Validates that present symptoms include at least one from each cluster

4. Identifies the three combinations that best match the original DSM-5 diagnosis

DSM-5 PTSD symptom clusters:

- Cluster 1 (B) - Intrusion: Items 1-5
- Cluster 2 (C) - Avoidance: Items 6-7
- Cluster 3 (D) - Negative alterations in cognitions and mood: Items 8-14
- Cluster 4 (E) - Alterations in arousal and reactivity: Items 15-20

Optimization can be based on either:

- Minimizing false cases (both false positives and false negatives)
- Minimizing only false negatives (newly non-diagnosed cases)

Value

A list containing:

- best_symptoms: List of three vectors, each containing six symptom numbers representing the best combinations found
- diagnosis_comparison: Dataframe comparing original DSM-5 diagnosis with diagnoses based on the three best combinations
- summary: Interactive datatable (DT) showing diagnostic accuracy metrics for each combination

Examples

```
# Create example data
ptsd_data <- data.frame(matrix(sample(0:4, 200, replace=TRUE), ncol=20))
names(ptsd_data) <- paste0("symptom_", 1:20)

# Find best hierarchical combinations minimizing false cases
results <- analyze_best_six_symptoms_four_required_clusters(ptsd_data, score_by = "false_cases")

# Get symptom numbers
results$best_symptoms

# View raw comparison data
results$diagnosis_comparison

# View summary statistics
results$summary
```

binarize_data	<i>Binarize PCL-5 symptom scores</i>
---------------	--------------------------------------

Description

Converts PCL-5 symptom scores from their original 0-4 scale to binary values (0/1) based on the clinical threshold for symptom presence (≥ 2).

Usage

```
binarize_data(data)
```

Arguments

data A dataframe containing exactly 20 columns with PCL-5 item scores (output of `rename_ptsd_columns`). Each symptom should be scored on a 0-4 scale where:

- 0 = Not at all
- 1 = A little bit
- 2 = Moderately
- 3 = Quite a bit
- 4 = Extremely

Note: This function should only be used with raw symptom scores before calculating the total score, as it will convert all values in the dataframe to 0/1, which would invalidate any total score column if present.

Details

The function implements the standard clinical threshold for PTSD symptom presence where:

- Scores of 0-1 ("Not at all" and "A little bit") $\rightarrow$ 0 (symptom absent)
- Scores of 2-4 ("Moderately" to "Extremely") $\rightarrow$ 1 (symptom present)

Value

A dataframe with the same structure as input but with all symptom scores converted to binary values:

- 0 = Symptom absent (original scores 0-1)
- 1 = Symptom present (original scores 2-4)

Examples

```
# Create sample data
sample_data <- data.frame(
  matrix(sample(0:4, 20 * 10, replace = TRUE),
    nrow = 10,
    ncol = 20)
)
colnames(sample_data) <- paste0("symptom_", 1:20)

# Binarize scores
binary_data <- binarize_data(sample_data)
binary_data # Should only show 0s and 1s
```

calculate_ptsd_total	<i>Calculate PTSD total score</i>
----------------------	-----------------------------------

Description

Calculates the total PCL-5 (PTSD Checklist for DSM-5) score by summing all 20 symptom scores. The total score ranges from 0 to 80, with higher scores indicating greater symptom severity.

Usage

```
calculate_ptsd_total(data)
```

Arguments

data	A dataframe containing standardized PCL-5 item scores (output of <code>rename_ptsd_columns</code>). Each symptom should be scored on a 0-4 scale where: • 0 = Not at all • 1 = A little bit • 2 = Moderately • 3 = Quite a bit • 4 = Extremely
------	---

Details

Calculates the total score from PCL-5 items

Value

A dataframe with all original columns plus an additional column "total" containing the sum of all 20 symptom scores (range: 0-80)

Examples

```
# Create sample data
sample_data <- data.frame(
  matrix(sample(0:4, 20 * 10, replace = TRUE),
    nrow = 10,
    ncol = 20)
)
colnames(sample_data) <- paste0("symptom_", 1:20)

# Calculate total scores
scores_with_total <- calculate_ptsd_total(sample_data)
print(scores_with_total$total)
```

```
create_ptsd_diagnosis_binarized
```

Determine PTSD diagnosis based on DSM-5 criteria using binarized scores

Description

Determines whether DSM-5 diagnostic criteria for PTSD are met using binarized symptom scores (0/1) for PCL-5 items. This is an alternative to `determine_ptsd_diagnosis()` that works with pre-binarized data.

Usage

```
create_ptsd_diagnosis_binarized(data)
```

Arguments

data A dataframe containing exactly 20 columns of PCL-5 item scores (output of `rename_ptsd_columns`) named `symptom_1` to `symptom_20`. Each symptom should be scored on a 0-4 scale where:

- 0 = Not at all
- 1 = A little bit
- 2 = Moderately
- 3 = Quite a bit
- 4 = Extremely

Note: This function should only be used with raw symptom scores (output of `rename_ptsd_columns`) and not with data containing a total score column, as the internal binarization process would invalidate the total score.

Details

The function applies the DSM-5 diagnostic criteria for PTSD using binary indicators of symptom presence:

- Criterion B (Intrusion): At least 1 present symptom from items 1-5
- Criterion C (Avoidance): At least 1 present symptom from items 6-7
- Criterion D (Negative alterations in cognitions and mood): At least 2 present symptoms from items 8-14
- Criterion E (Alterations in arousal and reactivity): At least 2 present symptoms from items 15-20

Value

A dataframe with a single column "PTSD_orig" containing TRUE/FALSE values indicating whether DSM-5 diagnostic criteria are met based on binarized scores

Examples

```
# Create sample data
sample_data <- data.frame(
  matrix(sample(0:4, 20 * 10, replace = TRUE),
    nrow = 10,
    ncol = 20)
)
colnames(sample_data) <- paste0("symptom_", 1:20)

# Get diagnosis using binarized approach
diagnosis_results <- create_ptsd_diagnosis_binarized(sample_data)
diagnosis_results$PTSD_orig
```

```
create_ptsd_diagnosis_nonbinarized
```

Determine PTSD diagnosis based on DSM-5 criteria using non-binarized scores

Description

Determines whether DSM-5 diagnostic criteria for PTSD are met based on PCL-5 item scores, using the original non-binarized values (0-4 scale).

Usage

```
create_ptsd_diagnosis_nonbinarized(data)
```

Arguments

- data** A dataframe that can be either:
- Output of `rename_ptsd_columns()`: 20 columns named `symptom_1` to `symptom_20`
 - Output of `calculate_ptsd_total()`: 21 columns including `symptom_1` to `symptom_20` plus a 'total' column
- Each symptom should be scored on a 0-4 scale where:
- 0 = Not at all
 - 1 = A little bit
 - 2 = Moderately
 - 3 = Quite a bit
 - 4 = Extremely

Details

The function applies the DSM-5 diagnostic criteria for PTSD:

- Criterion B (Intrusion): At least 1 symptom ≥ 2 from items 1-5
- Criterion C (Avoidance): At least 1 symptom ≥ 2 from items 6-7
- Criterion D (Negative alterations in cognitions and mood): At least 2 symptoms ≥ 2 from items 8-14
- Criterion E (Alterations in arousal and reactivity): At least 2 symptoms ≥ 2 from items 15-20

A symptom is considered present when rated 2 (Moderately) or higher.

Value

A dataframe with all original columns (including 'total' if present) plus an additional column "PTSD_Diagnosis" containing TRUE/FALSE values indicating whether DSM-5 diagnostic criteria are met

Examples

```
# Example with output from rename_ptsd_columns
sample_data1 <- data.frame(
  matrix(sample(0:4, 20 * 10, replace = TRUE),
    nrow = 10,
    ncol = 20)
)
colnames(sample_data1) <- paste0("symptom_", 1:20)
diagnosed_data1 <- create_ptsd_diagnosis_nonbinarized(sample_data1)

# Check diagnosis results
diagnosed_data1$PTSD_Diagnosis

# Example with output from calculate_ptsd_total
sample_data2 <- calculate_ptsd_total(sample_data1)
```

```
diagnosed_data2 <- create_ptsd_diagnosis_nonbinarized(sample_data2)

# Check diagnosis results
diagnosed_data2$PTSD_Diagnosis
```

`create_readable_summary`*Create readable summary of PTSD diagnostic changes*

Description

Formats the output of `summarize_ptsd_changes()` into a more readable table with proper labels and formatting of percentages and metrics.

Usage

```
create_readable_summary(summary_stats)
```

Arguments

`summary_stats` A dataframe output from `summarize_ptsd_changes()` containing raw diagnostic metrics and counts

Details

Reformats the diagnostic metrics into a presentation-ready format:

- Combines counts with percentages for diagnosed/non-diagnosed cases
- Rounds diagnostic accuracy metrics to 4 decimal places
- Provides clear column headers for all metrics

Value

A formatted dataframe with the following columns:

- Scenario: Name of the diagnostic criterion
- Total Diagnosed: Count and percentage of diagnosed cases
- Total Non-Diagnosed: Count and percentage of non-diagnosed cases
- True Positive: Count of cases diagnosed under both criteria
- True Negative: Count of cases not diagnosed under either criterion
- Newly Diagnosed: Count of new positive diagnoses (false positive)
- Newly Non-Diagnosed: Count of new negative diagnoses (false negative)
- True Cases: Total correctly classified cases
- False Cases: Total misclassified cases
- Sensitivity, Specificity, PPV, NPV: Diagnostic accuracy metrics (4 decimals)

Examples

```
# Using the output from summarize_ptsd_changes
n_cases <- 100
sample_data <- data.frame(
  PTSD_orig = sample(c(TRUE, FALSE), n_cases, replace = TRUE),
  PTSD_alt1 = sample(c(TRUE, FALSE), n_cases, replace = TRUE)
)

# Generate and format summary
diagnostic_metrics <- summarize_ptsd_changes(sample_data)
readable_summary <- create_readable_summary(diagnostic_metrics)
print(readable_summary)
```

cross_validation	<i>Perform k-fold cross-validation for PTSD diagnostic models</i>
------------------	---

Description

Validates PTSD diagnostic models using k-fold cross-validation to assess generalization performance and identify stable symptom combinations.

Usage

```
cross_validation(data, k = 5, score_by = "newly_nondiagnosed", seed = 123)
```

Arguments

data	A dataframe containing exactly 20 columns with PCL-5 item scores (output of <code>rename_ptsd_columns</code>). Each symptom should be scored on a 0-4 scale.
k	Number of folds for cross-validation (default: 5)
score_by	Character string specifying optimization criterion: "false_cases": Minimize total misclassifications "newly_nondiagnosed": Minimize false negatives only (default)
seed	Integer for random number generation reproducibility (default: 123)

Details

The function:

1. Splits data into k folds
2. For each fold, trains on k-1 folds and tests on the held-out fold
3. Identifies symptom combinations that appear across multiple folds
4. Calculates average performance metrics for repeated combinations

Two models are evaluated:

- Model without cluster representation: Any 4 of 6 symptoms
- Model with cluster representation: 4 of 6 symptoms with at least one from each cluster

Value

A list containing:

- without_clusters: Results for model without cluster representation
 - fold_results: List of diagnostic comparisons for each fold
 - summary_by_fold: Detailed results for each fold
 - combinations_summary: Average performance for combinations appearing in multiple folds (NULL if no combinations repeat)
- with_clusters: Results for model with cluster representation
 - fold_results: List of diagnostic comparisons for each fold
 - summary_by_fold: Detailed results for each fold
 - combinations_summary: Average performance for combinations appearing in multiple folds (NULL if no combinations repeat)

Examples

```
# Create sample data
set.seed(42)
sample_data <- data.frame(
  matrix(sample(0:4, 20 * 200, replace = TRUE),
    nrow = 200,
    ncol = 20)
)
colnames(sample_data) <- paste0("symptom_", 1:20)

# Perform 5-fold cross-validation
cv_results <- cross_validation(sample_data, k = 5)

# View summary for each fold
cv_results$without_clusters$summary_by_fold

# View combinations that appeared multiple times
cv_results$without_clusters$combinations_summary
```

holdout_validation	<i>Perform holdout validation for PTSD diagnostic models</i>
--------------------	--

Description

Validates PTSD diagnostic models using a train-test split approach (holdout validation). Trains the model on a portion of the data and evaluates performance on the held-out test set.

Usage

```
holdout_validation(
  data,
  train_ratio = 0.7,
  score_by = "newly_nondiagnosed",
  seed = 123
)
```

Arguments

data	A dataframe containing exactly 20 columns with PCL-5 item scores (output of <code>rename_ptsd_columns</code>). Each symptom should be scored on a 0-4 scale.
train_ratio	Numeric between 0 and 1 indicating proportion of data for training (default: 0.7 for 70/30 split)
score_by	Character string specifying optimization criterion: "false_cases": Minimize total misclassifications "newly_nondiagnosed": Minimize false negatives only (default)
seed	Integer for random number generation reproducibility (default: 123)

Details

The function:

1. Splits data into training (70
2. Finds optimal symptom combinations on training data
3. Evaluates these combinations on test data
4. Compares results to original DSM-5 diagnoses

Two models are evaluated:

- Model without cluster representation: Any 4 of 6 symptoms
- Model with cluster representation: 4 of 6 symptoms with at least one from each cluster

Value

A list containing:

- `without_clusters`: Results for model without cluster representation
 - `best_combinations`: The 3 best six-symptom combinations from training
 - `test_results`: Diagnostic comparison on test data
 - `summary`: Formatted summary statistics
- `with_clusters`: Results for model with cluster representation
 - `best_combinations`: The 3 best six-symptom combinations from training
 - `test_results`: Diagnostic comparison on test data
 - `summary`: Formatted summary statistics

Examples

```
# Create sample data
set.seed(42)
sample_data <- data.frame(
  matrix(sample(0:4, 20 * 200, replace = TRUE),
    nrow = 200,
    ncol = 20)
)
colnames(sample_data) <- paste0("symptom_", 1:20)

# Perform holdout validation
validation_results <- holdout_validation(sample_data, train_ratio = 0.7)

# Access results
validation_results$without_clusters$summary
validation_results$with_clusters$summary
```

rename_ptsd_columns	<i>Rename PTSD symptom (= PCL-5 item) columns</i>
---------------------	---

Description

Standardizes column names in PCL-5 (PTSD Checklist for DSM-5) data by renaming them to a consistent format (symptom_1 through symptom_20). This standardization is essential for subsequent analyses using other functions in the package.

Usage

```
rename_ptsd_columns(data)
```

Arguments

data	A dataframe containing exactly 20 columns, where each column represents a PCL-5 item score. The scores should be on a 0-4 scale where: • 0 = Not at all • 1 = A little bit • 2 = Moderately • 3 = Quite a bit • 4 = Extremely
------	--

Details

The function assumes the input data contains exactly 20 columns corresponding to the 20 items of the PCL-5. The columns are renamed sequentially from symptom_1 to symptom_20, maintaining their original order. The PCL-5 items correspond to different symptom clusters:

- symptom_1 to symptom_5: Intrusion symptoms (Criterion B)
- symptom_6 to symptom_7: Avoidance symptoms (Criterion C)
- symptom_8 to symptom_14: Negative alterations in cognitions and mood (Criterion D)
- symptom_15 to symptom_20: Alterations in arousal and reactivity (Criterion E)

Value

A dataframe with the same data but renamed columns following the pattern 'symptom_1' through 'symptom_20'

Examples

```
# Example with a sample PCL-5 dataset
sample_data <- data.frame(
  matrix(sample(0:4, 20 * 10, replace = TRUE),
    nrow = 10,
    ncol = 20)
)
renamed_data <- rename_ptsd_columns(sample_data)
colnames(renamed_data) # Shows new column names
```

simulated_ptsd

Simulated PCL-5 (PTSD Checklist) Data

Description

A dataset containing simulated responses from 5,000 patients on the PCL-5 (PTSD Checklist for DSM-5). Each patient rated 20 PTSD symptoms on a scale from 0 to 4.

Usage

```
simulated_ptsd
```

Format

A data frame with 5,000 rows and 20 columns:

S1 Intrusive memories

S2 Nightmares

S3 Flashbacks

- S4** Emotional reactivity to reminders
- S5** Physical reactions to reminders
- S6** Avoiding memories/thoughts/feelings
- S7** Avoiding external reminders
- S8** Amnesia
- S9** Strong negative beliefs
- S10** Distorted blame
- S11** Negative trauma-related emotions
- S12** Decreased interest in activities
- S13** Detachment or estrangement
- S14** Trouble experiencing positive emotions
- S15** Irritability/aggression
- S16** Risk-taking behavior
- S17** Hypervigilance
- S18** Heightened startle reaction
- S19** Difficulty concentrating
- S20** Sleep problems

Details

The symptoms are rated on a 5-point scale:

- 0 = Not at all
- 1 = A little bit
- 2 = Moderately
- 3 = Quite a bit
- 4 = Extremely

The symptoms correspond to DSM-5 PTSD criteria:

- Symptoms 1-5: Criterion B (Intrusion)
- Symptoms 6-7: Criterion C (Avoidance)
- Symptoms 8-14: Criterion D (Negative alterations in cognitions and mood)
- Symptoms 15-20: Criterion E (Alterations in arousal and reactivity)

Source

Simulated data for demonstration purposes

summarize_ptsd	<i>Summarize PTSD scores and diagnoses</i>
----------------	--

Description

Creates a summary of PCL-5 total scores and PTSD diagnoses, including mean total score, standard deviation, and number of positive diagnoses.

Usage

```
summarize_ptsd(data)
```

Arguments

data	A dataframe containing at minimum: • A 'total' column with PCL-5 total scores (from calculate_ptsd_total) • A 'PTSD_Diagnosis' column with TRUE/FALSE values (from determine_ptsd_diagnosis)
------	---

Details

This function calculates key summary statistics for PCL-5 data:

- Mean total score (severity indicator)
- Standard deviation of total scores (variability in severity)
- Count of positive PTSD diagnoses (prevalence in the sample)

Value

A dataframe with one row containing:

- mean_total: Mean PCL-5 total score
- sd_total: Standard deviation of PCL-5 total scores
- n_diagnosed: Number of positive PTSD diagnoses

Examples

```
# Create sample data
sample_data <- data.frame(
  total = sample(0:80, 100, replace = TRUE),
  PTSD_Diagnosis = sample(c(TRUE, FALSE), 100, replace = TRUE)
)

# Generate summary statistics
summary_stats <- summarize_ptsd(sample_data)
print(summary_stats)
```

`summarize_ptsd_changes`*Summarize changes in PTSD diagnostic metrics*

Description

Compares different PTSD diagnostic criteria by calculating diagnostic accuracy metrics and changes in diagnosis status relative to a baseline criterion.

Usage

```
summarize_ptsd_changes(data)
```

Arguments

`data`

A dataframe where:

- Each column represents a different diagnostic criterion
- Must include a column named "PTSD_orig" as the baseline criterion
- Values are logical (TRUE/FALSE) indicating whether PTSD criteria are met
- Each row represents one case/participant

Details

The function calculates multiple diagnostic metrics comparing each diagnostic criterion to a baseline criterion (PTSD_orig):

Basic counts:

- Number and percentage of diagnosed/non-diagnosed cases per criterion
- Number of newly diagnosed (false positive) and newly non-diagnosed (false negative) cases
- True positive and true negative cases

Diagnostic accuracy metrics:

- Sensitivity: Proportion of true PTSD cases correctly identified
- Specificity: Proportion of non-PTSD cases correctly identified
- PPV (Positive Predictive Value): Probability that a positive diagnosis is correct
- NPV (Negative Predictive Value): Probability that a negative diagnosis is correct

Value

A dataframe containing the following columns for each diagnostic criterion:

- column: Name of the diagnostic criterion
- diagnosed: Number of cases diagnosed as PTSD
- non_diagnosed: Number of cases not diagnosed as PTSD
- diagnosed_percent: Percentage of cases diagnosed
- non_diagnosed_percent: Percentage of cases not diagnosed
- newly_diagnosed: Cases diagnosed under new but not baseline criterion (false positive)
- newly_nondiagnosed: Cases diagnosed under baseline but not new criterion (false negative)
- true_positive: Cases diagnosed under both criteria
- true_negative: Cases not diagnosed under either criterion
- true_cases: Sum of true positives and true negatives
- false_cases: Sum of newly diagnosed (false positive) and newly non-diagnosed (false negative)
- sensitivity, specificity, ppv, npv: Standard diagnostic accuracy metrics

Examples

```
# Create sample diagnostic data
set.seed(123)
n_cases <- 100
sample_data <- data.frame(
  PTSD_orig = sample(c(TRUE, FALSE), n_cases, replace = TRUE),
  PTSD_alt1 = sample(c(TRUE, FALSE), n_cases, replace = TRUE),
  PTSD_alt2 = sample(c(TRUE, FALSE), n_cases, replace = TRUE)
)

# Calculate diagnostic metrics
diagnostic_metrics <- summarize_ptsd_changes(sample_data)
diagnostic_metrics
```

Index

* **datasets**

simulated_ptsd, [16](#)

analyze_best_six_symptoms_four_required,
[2](#)

analyze_best_six_symptoms_four_required_clusters,
[4](#)

binarize_data, [6](#)

calculate_ptsd_total, [7](#)

create_ptsd_diagnosis_binarized, [8](#)

create_ptsd_diagnosis_nonbinarized, [9](#)

create_readable_summary, [11](#)

cross_validation, [12](#)

holdout_validation, [13](#)

rename_ptsd_columns, [15](#)

simulated_ptsd, [16](#)

summarize_ptsd, [18](#)

summarize_ptsd_changes, [19](#)