Package ‘recorder’

October 14, 2022

Title Toolkit to Validate New Data for a Predictive Model
Version 0.8.2
Description A lightweight toolkit to validate new observations when computing their predictions with a predictive model. The validation process consists of two steps: (1) record relevant statistics and meta data of the variables in the original training data for the predictive model and (2) use these data to run a set of basic validation tests on the new set of observations.

URL https://github.com/smaakage85/recorder
Depends R (>= 3.4.0)
License MIT + file LICENSE
Encoding UTF-8
LazyData true
RoxygenNote 6.1.1
Imports data.table, crayon
Suggests testthat, knitr, rmarkdown
VignetteBuilder knitr
NeedsCompilation no
Author Lars Kjeldgaard [aut, cre]
Maintainer Lars Kjeldgaard <lars_kjeldgaard@hotmail.com>
Repository CRAN
Date/Publication 2019-06-13 08:40:03 UTC

R topics documented:

  compress_detailed_tests ........................................ 2
  concatenate_test_failures .................................... 3
  create_tests_meta_data ....................................... 3
  create_test_results_df .................................... 4
  get_clean_rows ........................................... 4
compress_detailed_tests

Compress Results of Detailed Tests

Description
Subsets results of the tests, where at least one row failed.

Usage
compress_detailed_tests(dt)

Arguments
- dt: list results of detailed tests.

Value
list with test failures.
**concatenate_test_failures**

*Concatenate Validation Test Failures Descriptions*

**Description**
Concatenates validation test failures descriptions to a single character vector.

**Usage**
```r
concatenate_test_failures(test_failures)
```

**Arguments**
- `test_failures` data.frame with test results as columns.

**Value**
character concatenated descriptions of test failures with one string pr. row.

---

**create_tests_meta_data**

*Create Meta Data of Validation Tests*

**Description**
Creates meta data of available validation tests as a list. The list has as many elements as the number of available validation test - one for each test. Entries are named after the different tests.

**Usage**
```r
create_tests_meta_data()
```

**Details**
The meta data of a validation test consists of:

- **evaluate_level** is the test evaluated on column level ('col') or on row level ('row')?
- **evaluate_class** what classes of variables are being tested with this specific test?
- **description** a short description of what a test failure means for the given test

**Value**
list meta data of validation tests.

**Examples**
```r
create_tests_meta_data()
```
### create_test_results_df

*Create Data Frame with Test Results*

**Description**

Create Data Frame with Test Results

**Usage**

```r
create_test_results_df(x)
```

**Arguments**

- `x` list results of tests.

**Value**

data.table with test results as columns.

### get_clean_rows

*Get Clean Rows*

**Description**

Get Clean Rows

**Usage**

```r
generic.get_clean_rows(playback, ignore_tests = NULL, ignore_cols = NULL, ignore_combinations = NULL)
```

**Arguments**

- `playback` data.playback to extract failed tests from.
- `ignore_tests` character ignore test results from tests with these names.
- `ignore_cols` character ignore test results from tests of columns with these names.
- `ignore_combinations` list ignore test results from specific tests of specific columns.

**Details**

Look up the descriptions and other meta data of the available validation tests with `get_tests_meta_data`.
get_failed_tests

Value

logical with the same length as the number of rows in new data. The value is TRUE, if the row passed all tests, otherwise FALSE.

Examples

# record tape from 'iris'.
tape <- record(iris)
# load data.
data(iris_newdata)
# validate new data by playing new tape on it.
playback <- play(tape, iris_newdata)

get_clean_rows(playback)
get_clean_rows(playback, ignore_tests = "outside_range")
get_clean_rows(playback, ignore_cols = "junk")
get_clean_rows(playback, ignore_combinations = list(outside_range = "Sepal.Width"))

get_failed_tests

Get Failed Tests

Description

Get Failed Tests

Usage

get_failed_tests(playback, ignore_tests = NULL, ignore_cols = NULL, ignore_combinations = NULL)

Arguments

playback data.playback to extract failed tests from.
ignore_tests character ignore test results from tests with these names.
ignore_cols character ignore test results from tests of columns with these names.
ignore_combinations

Value

data.table with test results as logicals for all of the tests with at least one failure. A failed test for any given row is equivalent to a value of TRUE. If all tests passed, the function will simply return a data.table with one column, ‘any failures’, that is always FALSE, to ensure that the output is (type) stable and consistent.
Examples

```
# record tape from `iris`.
tape <- record(iris)
# load data.
data(iris_newdata)
# validate new data by playing new tape on it.
playback <- play(tape, iris_newdata)

get_failed_tests(playback)
get_failed_tests(playback, ignore_tests = "outside_range")
get_failed_tests(playback, ignore_cols = "junk")
get_failed_tests(playback, ignore_combinations = list(outside_range = "Sepal.Width"))
```

---

**get_failed_tests_string**

*Get Failed Tests as a String*

**Description**

Concatenates information of the tests that failed into one single character vector.

**Usage**

```
get_failed_tests_string(playback, ignore_tests = NULL, ignore_cols = NULL, ignore_combinations = NULL)
```

**Arguments**

- **playback** data.playback to extract failed tests from.
- **ignore_tests** character ignore test results from tests with these names.
- **ignore_cols** character ignore test results from tests of columns with these names.
- **ignore_combinations** list ignore test results from specific tests of specific columns.

**Details**

Look up the descriptions and other meta data of the available validation tests with `get_tests_meta_data`.

**Value**

character with one entry for each row in new data. Each entry concatenates information of the tests, that did NOT pass for the corresponding row in new data.
Examples

# record tape from 'iris'.
tape <- record(iris)
# load data.
data(iris_newdata)
# validate new data by playing new tape on it.
playback <- play(tape, iris_newdata)

get_failed_tests_string(playback)
get_failed_tests_string(playback, ignore_tests = "outside_range")
get_failed_tests_string(playback, ignore_cols = "junk")
get_failed_tests_string(playback, ignore_combinations = list(outside_range = "Sepal.Width"))

---

get_tests_meta_data  Get Meta Data of Validation Tests in a Data Frame

Description

Gets meta data of available validation tests as a data.frame.

Usage

get_tests_meta_data()

Details

The meta data of a validation test consists of:

- **test_name**  name of the test
- **evaluate_level**  is the test evaluated on column level (‘col’) or on row level (‘row’)?
- **evaluate_class**  what classes of variables are being tested with this specific test?
- **description**  a short description of what a test failure means for the given test

Value

data.frame meta data of validation tests.

Examples

get_tests_meta_data()
ignore_clos

Description
Ignore Test Results from Tests of Specific Columns

Usage
ignore_clos(tests, col_names, variables_newdata)

Arguments

tests list test results.
col_names character names of columns for which test results should be ignored.
variables_newdata character names of variables in new data.
**ignore_combinations**

**Value**

list results after removing tests.

---

Ignore Test Results from Specific Tests of Specific Columns

**Usage**

ignore_combinations(tests, combinations, variables_newdata)

**Arguments**

tests list test results.
combinations list combinations of tests and columns from which test results should be ignored.
variables_newdata character names of variables in new data.

**Value**

list test results after removals.

---

Ignore Results from Specific Tests

**Usage**

ignore_tests(tests, test_names = NULL)

**Arguments**

tests list test results.
test_names character names of tests to be ignored.

**Value**

list results after removing specific tests.
### iris_newdata

**Simulated Iris New Data**

**Description**

A mutated version of the famous ‘iris’ data set.

**Usage**

```r
iris_newdata
```

**Format**

A data.frame with 150 rows and 5 columns.

**Source**

Script attached.

### order_by_tests

**Order Test Results by Test Names**

**Description**

Order Test Results by Test Names

**Usage**

```r
order_by_tests(dt)
```

**Arguments**

- `dt` list test results.

**Value**

list test results ordered by test names.
**Validate New Data by Playing a Data Tape on It**

**Description**

Runs a set of validation tests on new data to be predicted with an existing predictive model. These tests are based on statistics and meta data of the variables in the training data - recorded with `record`.

**Usage**

```r
play(tape, newdata, verbose = TRUE)
```

**Arguments**

- `tape`: `data.tape` statistics and meta data recorded from training data.
- `newdata`: `data.frame` new data to be predicted with an existing predictive model.
- `verbose`: logical should messages be printed?

**Details**

Look up the descriptions and other meta data of the available validation tests with `get_tests_meta_data`.

**Value**

data.playback results from validation tests.

**Examples**

```r
# record tape from 'iris'.
tape <- record(iris)
# load data.
data(iris_newdata)
# validate new data by playing new tape on it.
play(tape, iris_newdata)
```

**Print Data Playback**

**Description**

Print Data Playback

**Usage**

```r
## S3 method for class 'data.playback'
print(x, ...)
```
Arguments

x A `data.playback` object.
...

Value

The original object (invisibly)

Examples

# record tape from `iris`.
tape <- record(iris)
# load data.
data(iris_newdata)
# validate new data by playing new tape on it.
playback <- play(tape, iris_newdata)
# print it.
print(playback)

record Record Statistics and Meta Data of Variables in Training Data

Description

Records statistics and meta data of variables in the training data for a predictive model. The recorded
data can then be used to compute a set of validation tests on new data with `play`.

Usage

record(x, ...)

Arguments

x training data (or just a single variable from the training data) to record the statistics and other relevant meta data of.
...

Value

list recorded statistics and meta data. The list will inherit from the `data.tape` class when the
function is invoked with a `data.frame`.

Examples

record(iris)
**record.character**  
*Record Statistics and Meta Data of a Character*

**Description**  
Records statistics and meta data of a character.

**Usage**  
```r  
## S3 method for class 'character'  
record(x, ...)  
```

**Arguments**  
- `x`: character  
- `...`: all further arguments.

**Value**  
list recorded statistics and meta data.

**Examples**  
```r  
record(letters)  
```

---

**record.data.frame**  
*Record Statistics and Meta Data of a Data Frame*

**Description**  
Records Statistics and meta data of a data.frame.

**Usage**  
```r  
## S3 method for class 'data.frame'  
record(x, verbose = TRUE, ...)  
```

**Arguments**  
- `x`: data.frame training data for predictive model.  
- `verbose`: logical should messages be printed?  
- `...`: all further arguments.

**Value**  
list recorded statistics and meta data.
Examples

```r
record(iris)
```

---

**record.default**  
*Record Statistics and Meta Data*

**Description**

Records statistics and meta data.

**Usage**

```r
## Default S3 method:
record(x, ...)
```

**Arguments**

- `x`: anything.
- `...`: all further arguments.

**Value**

`list` recorded statistics and meta data.

**Examples**

```r
some_junk_letters <- letters[1:10]
class(some_junk_letters) <- "junk"
record(some_junk_letters)
```

---

**record.factor**  
*Record Statistics and Meta Data of a Factor*

**Description**

Records statistics and meta data of a factor.

**Usage**

```r
## S3 method for class 'factor'
record(x, ...)
```

**Arguments**

- `x`: factor
- `...`: all further arguments.
Value

list recorded statistics and meta data.

Examples

record(iris$Species)

record.integer  Record Statistics and Meta Data of an Integer

Description

Records statistics and meta data of an integer.

Usage

## S3 method for class 'integer'
record(x, ...)

Arguments

x integer
...
all further arguments.

Value

list recorded statistics and meta data.

Examples

record(c(1:10, NA_integer_))

record.numeric  Record Statistics and Meta Data of a Numeric

Description

Records statistics and meta data of a numeric.

Usage

## S3 method for class 'numeric'
record(x, ...)

# record integer

# record numeric
run_validation_tests

Arguments

x numeric
...

Value

list recorded statistics and meta data.

Examples

record(iris$Sepal.Length)

Description

Runs a set of validation tests on a variable in new data. These tests are based on statistics and meta data of the same variable recorded (with `record`) from the training data.

Usage

run_validation_tests(x, parameters, ...)

Arguments

x variable in new data.

parameters list statistics and meta data of the same variable recorded from training data (with `record`).
...

Details

Look up the descriptions and other meta data of the available validation tests with `get_tests_meta_data`.

Value

list results from validation tests.
run_validation_tests.character

Run Validation Tests on Character

Description

Runs a set of validation tests on a character in new data. These tests are based on statistics and meta data of the same variable recorded (with record) from the training data.

Usage

## S3 method for class 'character'
run_validation_tests(x, parameters, ...)

Arguments

x character in new data.

parameters list statistics and meta data of the same variable recorded from training data (with record).

... further arguments passed to or from other methods. Not used at the moment.

Value

list results from validation tests.

run_validation_tests.default

Run Validation Tests on Variable

Description

Runs a set of validation tests on variable in new data. These tests are based on statistics and meta data of the same variable recorded (with record) from the training data.

Usage

## Default S3 method:
run_validation_tests(x, parameters, ...)

Arguments

x anything.

parameters list statistics and meta data of the same variable recorded from training data (with record).

... further arguments passed to or from other methods. Not used at the moment.
### run_validation_tests.factor

**Run Validation Tests on Factor**

**Description**

Runs a set of validation tests on a factor in new data. These tests are based on statistics and meta data of the same variable recorded (with `record`) from the training data.

**Usage**

```r
## S3 method for class 'factor'
run_validation_tests(x, parameters, ...)
```

**Arguments**

- `x` factor in new data.
- `parameters` list statistics and meta data of the same variable recorded from training data (with `record`).
- `...` further arguments passed to or from other methods. Not used at the moment.

**Value**

list results from validation tests.

### run_validation_tests.integer

**Run Validation Tests on Integer**

**Description**

Runs a set of validation tests on a integer in new data. These tests are based on statistics and meta data of the same variable recorded (with `record`) from the training data.

**Usage**

```r
## S3 method for class 'integer'
run_validation_tests(x, parameters, ...)
```
run_validation_tests.numeric

Arguments

x integer in new data.
parameters list statistics and meta data of the same variable recorded from training data (with record).
... further arguments passed to or from other methods. Not used at the moment.

Value

list results from validation tests.

run_validation_tests.numeric

Run Validation Tests on a Numeric

Description

Runs a set of validation tests on a numeric in new data. These tests are based on statistics and meta data of the same variable recorded (with record) from the training data.

Usage

## S3 method for class 'numeric'
run_validation_tests(x, parameters, ...)

Arguments

x numeric in new data.
parameters list statistics and meta data of the same variable recorded from training data (with record).
... further arguments passed to or from other methods. Not used at the moment.

Value

list results from validation tests.
* datasets
  iris_newdata, 10

compress_detailed_tests, 2
concatenate_test_failures, 3
create_test_results_df, 4
create_tests_meta_data, 3

get_clean_rows, 4
get_failed_tests, 5
get_failed_tests_string, 6
get_tests_meta_data, 4, 6, 7, 8, 11, 16

ignore, 8
ignore_cols, 8
ignore_combinations, 9
ignore_tests, 9
iris_newdata, 10

order_by_tests, 10

play, 11, 12
print.data.playback, 11

record, 11, 12, 16–19
record.character, 13
record.data.frame, 13
record.default, 14
record.factor, 14
record.integer, 15
record.numeric, 16
run_validation_tests, 16
run_validation_tests.character, 17
run_validation_tests.default, 17
run_validation_tests.factor, 18
run_validation_tests.integer, 18
run_validation_tests.numeric, 19