Package 'validmind'

July 22, 2025

Type Package

Title Interface to the 'ValidMind' Platform

Version 0.1.2

Maintainer Andres Rodriguez <andres@validmind.ai>

Description Deploy, execute, and analyze the results of models hosted on the 'ValidMind' platform <https://validmind.com>. This package interfaces with the 'Python' client library in order to allow advanced diagnostics and insight into trained models all from an 'R' environment.

License AGPL-3

Encoding UTF-8

URL https://github.com/validmind/developer-framework

BugReports https://github.com/validmind/developer-framework/issues

RoxygenNote 7.3.2

Imports glue, reticulate, dplyr, plotly, htmltools, rmarkdown, DT, base64enc

NeedsCompilation no

Author Andres Rodriguez [aut, cre, cph]

Repository CRAN

Date/Publication 2024-11-07 19:00:02 UTC

Contents

`build_r_plotly` *Build an R Plotly figure from a JSON representation*

Description

Build an R Plotly figure from a JSON representation

Usage

```
build_r_plotly(plotly_figure)
```

Arguments

`plotly_figure`	A nested list containing plotly elements

Value

An R Plotly object derived from the JSON representation

`display_report` *Produce RMarkdown-compatible output of all results*

Description

Produce RMarkdown-compatible output of all results

Usage

```
display_report(processed_results)
```

Arguments

`processed_results`	A list of processed result objects

Value

A formatted list of RMarkdown widgets

Examples

```
## Not run:
vm_dataset = vm_r$init_dataset(
  dataset=data,
  target_column="Exited",
  class_labels=list("0" = "Did not exit", "1" = "Exited")
)

tabular_suite_results <- vm_r$run_test_suite("tabular_dataset", dataset=vm_dataset)

processed_results <- process_result(tabular_suite_results)
all_widgets <- display_report(processed_results)
for (widget in all_widgets) {
  print(widget)
}

## End(Not run)
```

`print_summary_tables` *Print a summary table of the ValidMind results*

Description

Print a summary table of the ValidMind results

Usage

```
print_summary_tables(result_summary)
```

Arguments

`result_summary`	A summary of the results

Value

A data frame containing the summary of the ValidMind results

process_result *Process a set of ValidMind results into parseable data*

Description

Process a set of ValidMind results into parseable data

Usage

```
process_result(results)
```

Arguments

results	A list of ValidMind result objects

Value

A nested list of ValidMind results (dataframes, plotly plots, and matplotlib plots)

Examples

```
## Not run:
vm_dataset = vm_r$init_dataset(
  dataset=data,
  target_column="Exited",
  class_labels=list("0" = "Did not exit", "1" = "Exited")
)

tabular_suite_results <- vm_r$run_test_suite("tabular_dataset", dataset=vm_dataset)

processed_results <- process_result(tabular_suite_results)
processed_results

## End(Not run)
```

register_custom_test *Register a Custom Test Function in ValidMind*

Description

Registers an R function as a custom test within the ValidMind testing framework, allowing it to be used as a custom metric for model validation.

Usage

```
register_custom_test(
  func,
  test_id = NULL,
  description = NULL,
  required_inputs = NULL
)
```

Arguments

func	An R function to be registered as a custom test.
test_id	A unique identifier for the test. If NULL, a default ID is generated based on the function name.
description	A description of the test. If NULL, the function's description attribute is used. Defaults to "No description" if not available.
required_inputs	A character vector specifying the required inputs for the test. If NULL, the function's formal argument names are used.

Details

The provided R function is converted into a Python callable using r_to_py. A Python class is then defined, inheriting from ValidMind's Metric class, which wraps this callable. This custom test is registered within ValidMind's test store and can be used in the framework for model validation purposes.

Value

The test store object containing the newly registered custom test.

See Also

r_to_py, import_main, py_run_string

Examples

```
## Not run:
# Define a custom test function in R
my_custom_metric <- function(predictions, targets) {
  # Custom metric logic
  mean(abs(predictions - targets))
}

# Register the custom test
register_custom_test(
  func = my_custom_metric,
  test_id = "custom.mae",
  description = "Custom Mean Absolute Error",
  required_inputs = c("predictions", "targets")
```

```
)

## End(Not run)
```

`run_custom_test` *Run a Custom Test using the ValidMind Framework*

Description

This function runs a custom test using the ValidMind framework through Python's 'validmind.vm_models'. It retrieves a custom test by 'test_id', executes it with the provided 'inputs', and optionally displays the result. The result is also logged.

Usage

```
run_custom_test(test_id, inputs, test_registry, show = FALSE)
```

Arguments

`test_id`	A string representing the ID of the custom test to run.
`inputs`	A list of inputs required for the custom test.
`test_registry`	A reference to the test register object which provides the custom test class.
`show`	A logical value. If TRUE, the result will be displayed. Defaults to FALSE.

Value

An object representing the result of the test, with an additional log function.

Examples

```
## Not run:
result <- run_custom_test("test123", my_inputs, test_registry, show = TRUE)

## End(Not run)
```

save_model *Save an R model to a temporary file*

Description

This function saves a given R model object to a randomly named '.RData' file in the '/tmp/' directory. The file is saved with a unique name generated using random letters.

Usage

```
save_model(model)
```

Arguments

model	The R model object to be saved.

Value

A string representing the full file path to the saved '.RData' file.

Examples

```
model <- lm(mpg ~ cyl, data = mtcars)
file_path <- save_model(model)
```

summarize_metric_result *Provide a summarization of a single metric result*

Description

Provide a summarization of a single metric result

Usage

```
summarize_metric_result(result)
```

Arguments

result	The ValidMind result object

Value

A list containing the summary of the ValidMind results

summarize_result *Provide a summarization of a single result (test or metric)*

Description

Provide a summarization of a single result (test or metric)

Usage

```
summarize_result(result)
```

Arguments

result	The ValidMind result object

Value

Based on the type of ‘result‘, either A list containing the summary of the ValidMind results, or a list containing the summary of the ValidMind results

summarize_test_result *Provide a summarization of a single test result*

Description

Provide a summarization of a single test result

Usage

```
summarize_test_result(result)
```

Arguments

result	The ValidMind result object

Value

A list containing the summary of the ValidMind test results

vm *Retrieve a validmind (vm) connection object using reticulate*

Description

Retrieve a validmind (vm) connection object using reticulate

Usage

```
vm(
  api_key,
  api_secret,
  model,
  python_version,
  api_host = "http://localhost:3000/api/v1/tracking"
)
```

Arguments

api_key	The ValidMind API key
api_secret	The ValidMind API secret
model	The ValidMind model
python_version	The Python Version to use
api_host	The ValidMind host, defaulting to local

Value

A validmind connection object, obtained from ‘reticulate‘, which orchestrates the connection to the ValidMind API

Examples

```
## Not run:
vm_r <- vm(
   api_key="<your_api_key_here>",
   api_secret="<your_api_secret_here>",
   model="<your_model_id_here>",
   python_version=python_version,
   api_host="https://api.dev.vm.validmind.ai/api/v1/tracking"
 )

## End(Not run)
```

Index