Package ‘modeltime.h2o’

April 5, 2021

Title Modeltime 'H2O' Machine Learning

Version 0.1.1

Description Use the 'H2O' machine learning library inside of 'modeltime'. Available models include 'AutoML' for Automatic Machine Learning. Please see H2O.ai for more information <https://github.com/h2oai/h2o-3>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.1.1

Depends modeltime (>= 0.4.1), h2o

Imports magrittr, rlang (>= 0.1.2), tibble, timetk (>= 2.6.0), dplyr, parsnip (>= 0.1.4), purrr, tidyr, stringr, stats, glue, fs

Suggests tidymodels, workflows, tidyverse, knitr, rmarkdown, roxygen2, testthat (>= 3.0.0), covr

SystemRequirements Java (>= 8)

URL https://github.com/business-science/modeltime.h2o

BugReports https://github.com/business-science/modeltime.h2o/issues

NeedsCompilation no

Author Matt Dancho [aut, cre],
Business Science [cph]

Maintainer Matt Dancho <mdancho@business-science.io>

Repository CRAN

Date/Publication 2021-04-05 14:40:03 UTC

R topics documented:

automl_fit_impl *H2O AutoML Modeling Function (Bridge)*

Description

H2O AutoML Modeling Function (Bridge)

Usage

```
automl_fit_impl(x, y, ...)
```

Arguments

`x`	A dataframe of xreg (exogenous regressors)
`y`	A numeric vector of values to fit
`...`	Additional arguments passed to `h2o.automl()`.

Value

A fitted model with class `automl_fit_impl` and `modeltime_bridge`.

automl_leaderboard *H2O AutoML Leaderboard Utilities*

Description

The H2O AutoML Leaderboard lists any models that have been created during the `automl_reg()` training process.

- The training process automatically uses the top model.
- The available models can be shown with `automl_leaderboard()`
- The model change the model used using `automl_update_model()`.

Usage

```
automl_leaderboard(object)

automl_update_model(object, model_id)
```

Arguments

`object`	An object created by `automl_reg()` and trained (fitted).
`model_id`	An H2O Model ID (shown in the AutoML Leaderboard). Alternatively, the user can provide an H2O model.

Value

- `automl_leaderboard()`: A `tibble` containing the H2O AutoML Leaderboard
- `automl_update_model()`: An updated `parnsip` or `workflow` with the H2O Model updated

Examples

```
## Not run:
library(tidymodels)
library(modeltime.h2o)
library(h2o)
library(tidyverse)
library(timetk)

h2o.init(
    nthreads = -1,
    ip       = 'localhost',
    port     = 54321
)

# Model Spec
model_spec <- automl_reg(mode = 'regression') %>%
    set_engine(
        engine                     = 'h2o',
        max_runtime_secs           = 5,
        max_runtime_secs_per_model = 4,
        nfolds                     = 5,
        max_models                 = 3,
        exclude_algos              = c("DeepLearning"),
        seed                       = 786
    )


# Fit AutoML
model_fit <- model_spec %>%
    fit(value ~ ., data = training(m750_splits))

# Inspect the Leaderboard
leaderboard_tbl <- automl_leaderboard(model_fit)
leaderboard_tbl

# Swap an H2O Model Out (Using the 2nd model from the leaderboard)
model_id_2  <- leaderboard_tbl$model_id[[2]]
model_fit_2 <- automl_update_model(model_fit, model_id_2)
model_fit_2

# Shutdown H2O when Finished.
# Make sure to save any work before.
h2o.shutdown(prompt = FALSE)



## End(Not run)
```

automl_predict_impl *Bridge prediction Function for H2O AutoML Models*

Description

Bridge prediction Function for H2O AutoML Models

Usage

```
automl_predict_impl(object, new_data, ...)
```

Arguments

`object`	An object of class `model_fit`
`new_data`	A rectangular data object, such as a data frame.
`...`	Additional arguments passed to `h2o::h2o.predict()`

Value

A vector of values (predictions) with class `numeric`.

automl_reg *General Interface for H2O AutoML Time Series Models*

Description

`automl_reg()` is a way to generate a *specification* of a AutoML model before fitting and allows the model to be created using different packages. Currently the only package is `h2o`.

Usage

```
automl_reg(mode = "regression")
```

Arguments

`mode`	A single character string for the type of model. The only possible value for this model is "regression".

Details

Other options and arguments can be set using `set_engine()`.

The model can be created using the fit() function using the following engines:

- **H2O** "h2o" (the default)

Value

An updated model specification with classes `automl_reg` and `model_spec`.

Engine

h2o

The engine uses `h2o.automl()`.

Fit Details

The following features are REQUIRED to be available in the incoming data for the fitting process.

- **Fit:** `fit(y ~ .,data)`: Includes a target feature that is a function of a "date" feature.
- **Predict:** `predict(model,new_data)` where `new_data` contains a column named "date".

Date and Date-Time Variable

It's a requirement to have a date or date-time variable as a predictor. The `fit()` interface accepts date and date-time features and handles them internally.

See Also

`fit.model_spec()`, `set_engine()`

Examples

```
## Not run:
library(tidymodels)
library(modeltime.h2o)
library(h2o)
library(tidyverse)
library(timetk)

data_tbl <- walmart_sales_weekly %>%
    select(id, Date, Weekly_Sales)

splits <- time_series_split(
    data_tbl,
    assess     = "3 month",
    cumulative = TRUE
)

recipe_spec <- recipe(Weekly_Sales ~ ., data = training(splits)) %>%
    step_timeseries_signature(Date)

train_tbl <- bake(prep(recipe_spec), training(splits))
test_tbl  <- bake(prep(recipe_spec), testing(splits))

# Initialize H2O

h2o.init(
    nthreads = -1,
    ip = 'localhost',
    port = 54321
)



# ---- MODEL SPEC ----
model_spec <- automl_reg(mode = 'regression') %>%
    set_engine(
        engine                     = 'h2o',
        max_runtime_secs           = 30,
        max_runtime_secs_per_model = 30,
        project_name               = 'project_01',
        nfolds                     = 5,
        max_models                 = 1000,
        exclude_algos              = c("DeepLearning"),
        seed                       =  786
    )

model_spec

# ---- TRAINING ----
# Important: Make sure the date is included as regressor.

# This training process should take 30-40 seconds
model_fitted <- model_spec %>%
    fit(Weekly_Sales ~ ., data = train_tbl)

model_fitted

# ---- PREDICT ----
# - IMPORTANT: New Data must have date feature

predict(model_fitted, test_tbl)

# Shutdown H2O when Finished.
# Make sure to save any work before.
h2o.shutdown(prompt = FALSE)

## End(Not run)
```

`save_h2o_model` *Saving and Loading Modeltime H2O Models*

Description

H2O AutoML models require a special storage process that saves / loads the recipe used to recreate a model to / from a directory that the user defines.

Usage

```
save_h2o_model(object, path, overwrite = FALSE)

load_h2o_model(path)
```

Arguments

`object` A fitted model object

`path` A directory to store the H2O AutoML model files

`overwrite` Whether or not to allow overwriting a H2O AutoML model's directory. Default: FALSE.

Value

- `save_h2o_model()`: No return value, called for side effects (composes a directory of model files)
- `load_h2o_model()`: No return value, called for side effects (reads a directory of model files)

Examples

```
## Not run:
library(tidymodels)
library(tidyverse)
library(timetk)
library(modeltime.h2o)

h2o.init()

model_fit <- automl_reg(mode = 'regression') %>%
    set_engine(
        engine                     = 'h2o',
        max_runtime_secs           = 30,
        max_runtime_secs_per_model = 30,
        project_name               = 'project_01',
        nfolds                     = 5,
        max_models                 = 1000,
        exclude_algos              = c("DeepLearning"),
        seed                       =  786
    ) %>%
    fit(value ~ date + id, m750)

# Saves the related files needed to recreate the model
model_fit %>% save_h2o_model(path = "/dir_h2o_automl_model/")

# Loads the model
load_h2o_model(path = "/dir_h2o_automl_model/")

# Shutdown H2O when Finished.
# Make sure to save any work before.
h2o.shutdown(prompt = FALSE)
```

```
## End(Not run)
```

Index