SlideR01

Table of contents

mlr3

- 教師付き学習の” メタ” パッケージ
- OOP
- インスタンス化を明示的に行う

SetUp

```r
set.seed(111)
library(tidyverse)
library(mlr3verse)

Data <- arrow::read_parquet(
  "Public/Example.parquet"
)
```

Define instance

```r
TreeDepth2 <- lrn("regr.rpart")

TreeDepth2$param_set$values$maxdepth <- 2

TreeDepth2$param_set$values$cp <- 0

TreeDepth2
```

```
<LearnerRegrRpart:regr.rpart>: Regression Tree
* Model: -
* Parameters: xval=0, maxdepth=2, cp=0
* Packages: mlr3, rpart
* Predict Types:  [response]
* Feature Types: logical, integer, numeric, factor, ordered
* Properties: importance, missings, selected_features, weights
```

Define Task

```r
Task <- as_task_regr(
  x = Data,
  target = "Price"
  )

Task
```

```
<TaskRegr:Data> (16786 x 38)
* Target: Price
* Properties: -
* Features (37):
  - dbl (37): After, DistanceStation, District_X13101, District_X13102,
    District_X13103, District_X13104, District_X13105, District_X13106,
    District_X13107, District_X13108, District_X13109, District_X13110,
    District_X13111, District_X13112, District_X13113, District_X13114,
    District_X13115, District_X13116, District_X13117, District_X13118,
    District_X13119, District_X13120, District_X13121, District_X13122,
    District_X13123, Kenpei, Reform, RoomD, RoomL, RoomNum, Size,
    StructureSRC, Tenure, Youseki, ZoneBusiness, ZoneFactory, ZoneHouse
```

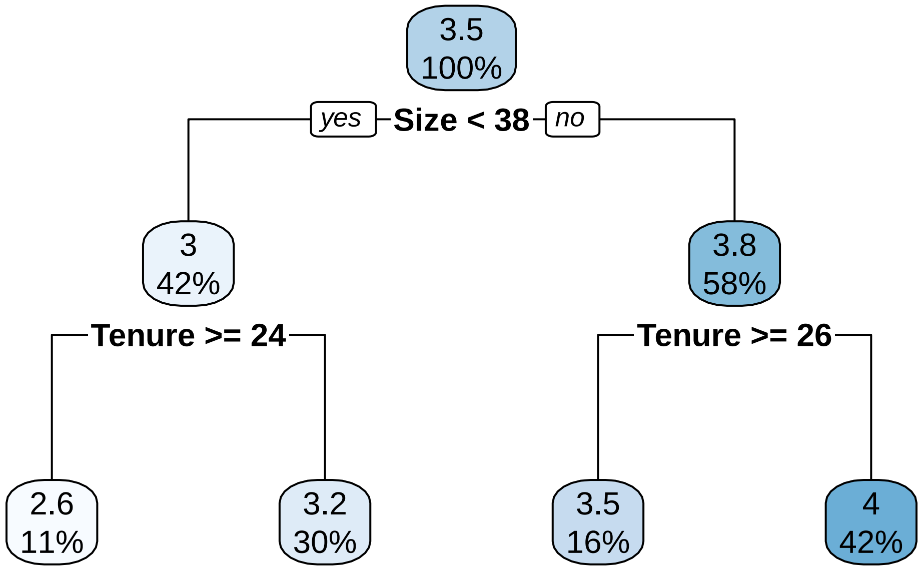
Fitting

```
Group <- partition(
  Task,
  ratio = 0.8
  ) # Split sample

TreeDepth2$train(
  Task,
  Group$train
  ) # Fit tree model
```

Visualization

```
rpart.plot::rpart.plot(TreeDepth2$model)
```



Validation

```
TreeDepth2$predict(
  Task,
  Group$test)$
  score(msr("regr.rsq"))
```

```
 regr.rsq
0.5379693
```

Use benchmark_grid

```
TreeDepth2 <- lrn("regr.rpart", id = "depth 2")

TreeDepth2$param_set$values$maxdepth <- 2

TreeDepth2$param_set$values$cp <- 0

TreeDepth10 <- lrn("regr.rpart", id = "depth 10")

TreeDepth10$param_set$values$maxdepth <- 10

TreeDepth10$param_set$values$cp <- 0
```

Use benchmark_grid

```
Design <- benchmark_grid(
  tasks = Task,
  learners = list(
    TreeDepth2,
    TreeDepth10
  ),
  resamplings = rsmp(
    "holdout",
    ratio = 0.8) # Training(8割)/Test(2割) へ分割
)
```

```
BenchMark <- benchmark(Design)
```

```
INFO  [12:56:04.249] [mlr3] Running benchmark with 2 resampling iterations
INFO  [12:56:04.280] [mlr3] Applying learner 'depth 2' on task 'Data' (iter 1/1)
INFO  [12:56:04.319] [mlr3] Applying learner 'depth 10' on task 'Data' (iter 1/1)
INFO  [12:56:04.480] [mlr3] Finished benchmark
```

Use benchmark_grid

```
BenchMark$aggregate(msr("regr.rsq"))
```

```
   nr task_id learner_id resampling_id iters  regr.rsq
1:  1    Data    depth 2       holdout     1 0.5527818
2:  2    Data   depth 10       holdout     1 0.7886166
Hidden columns: resample_result
```