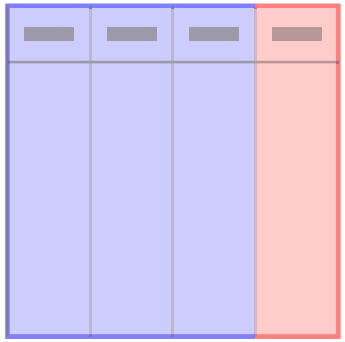
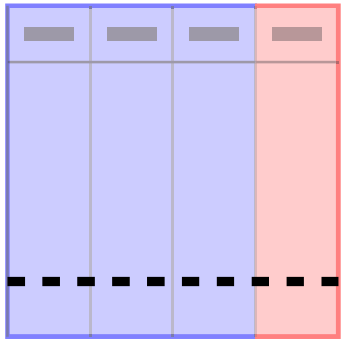
Resampling

RESAMPLING



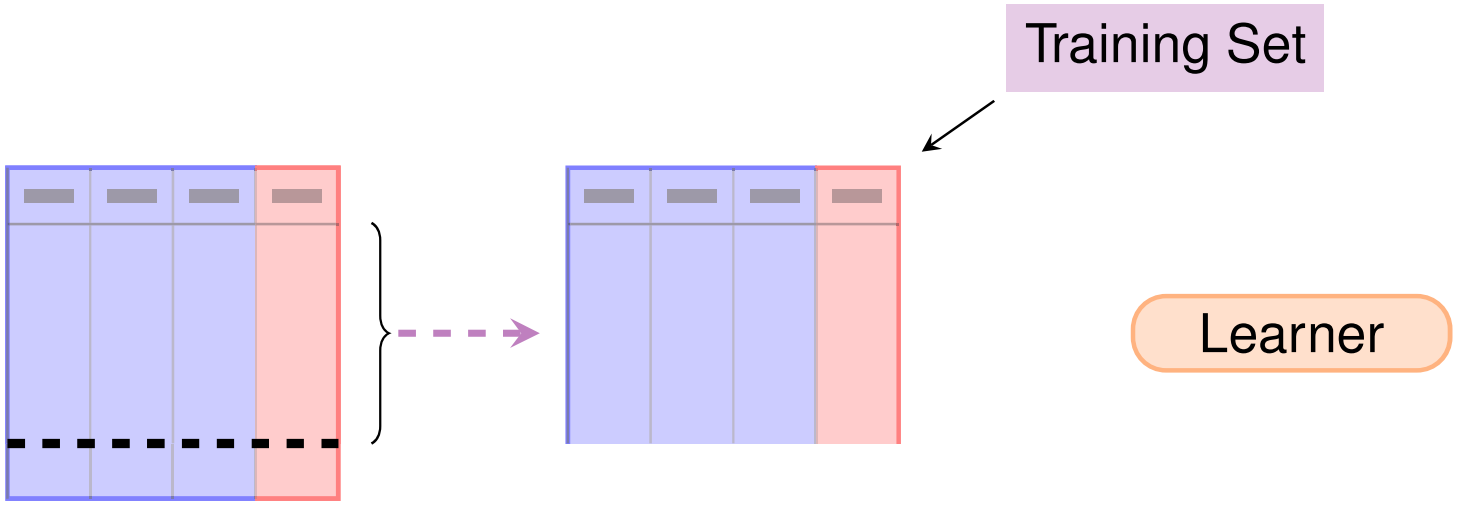
Learner

RESAMPLING

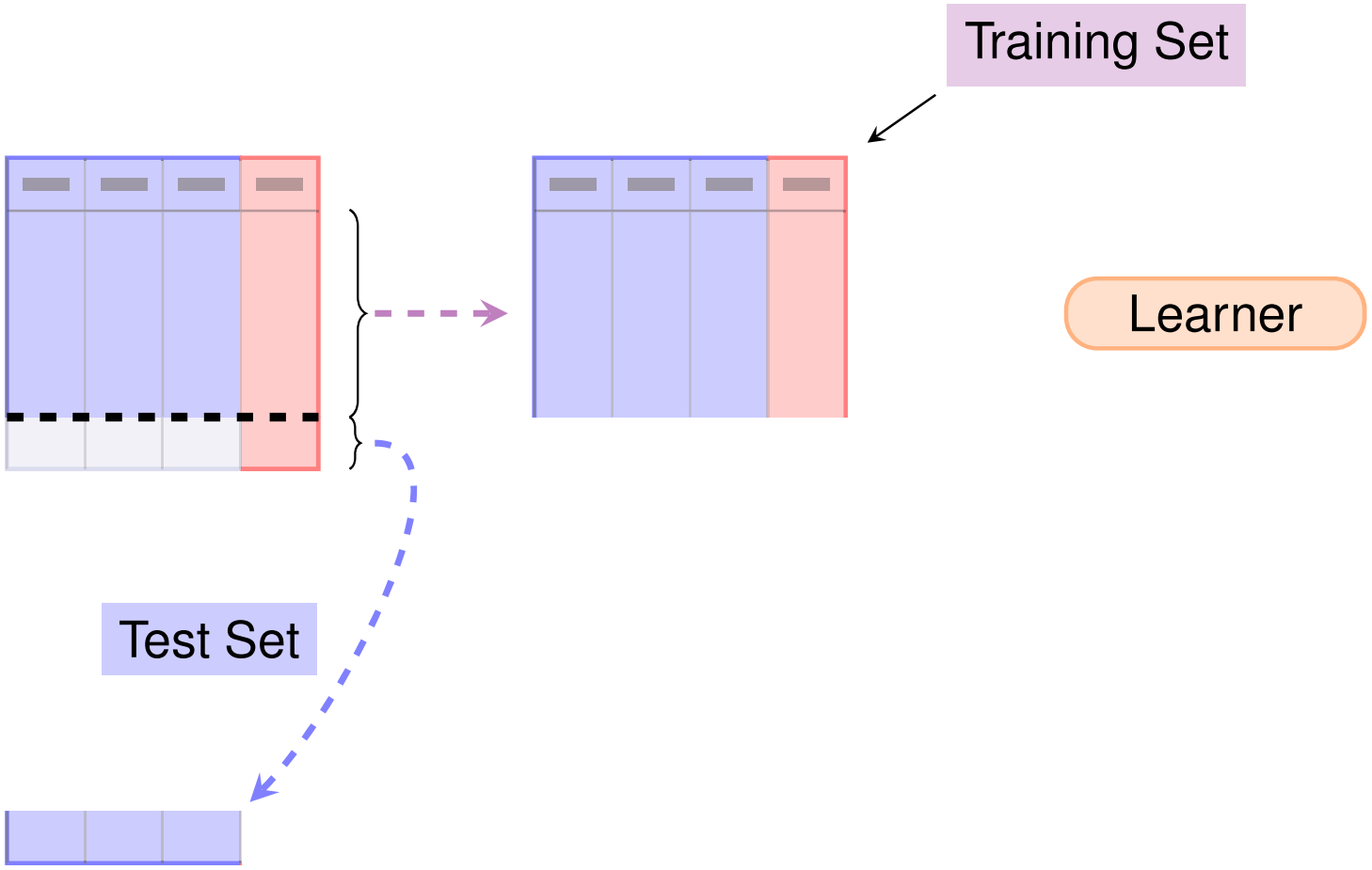


Learner

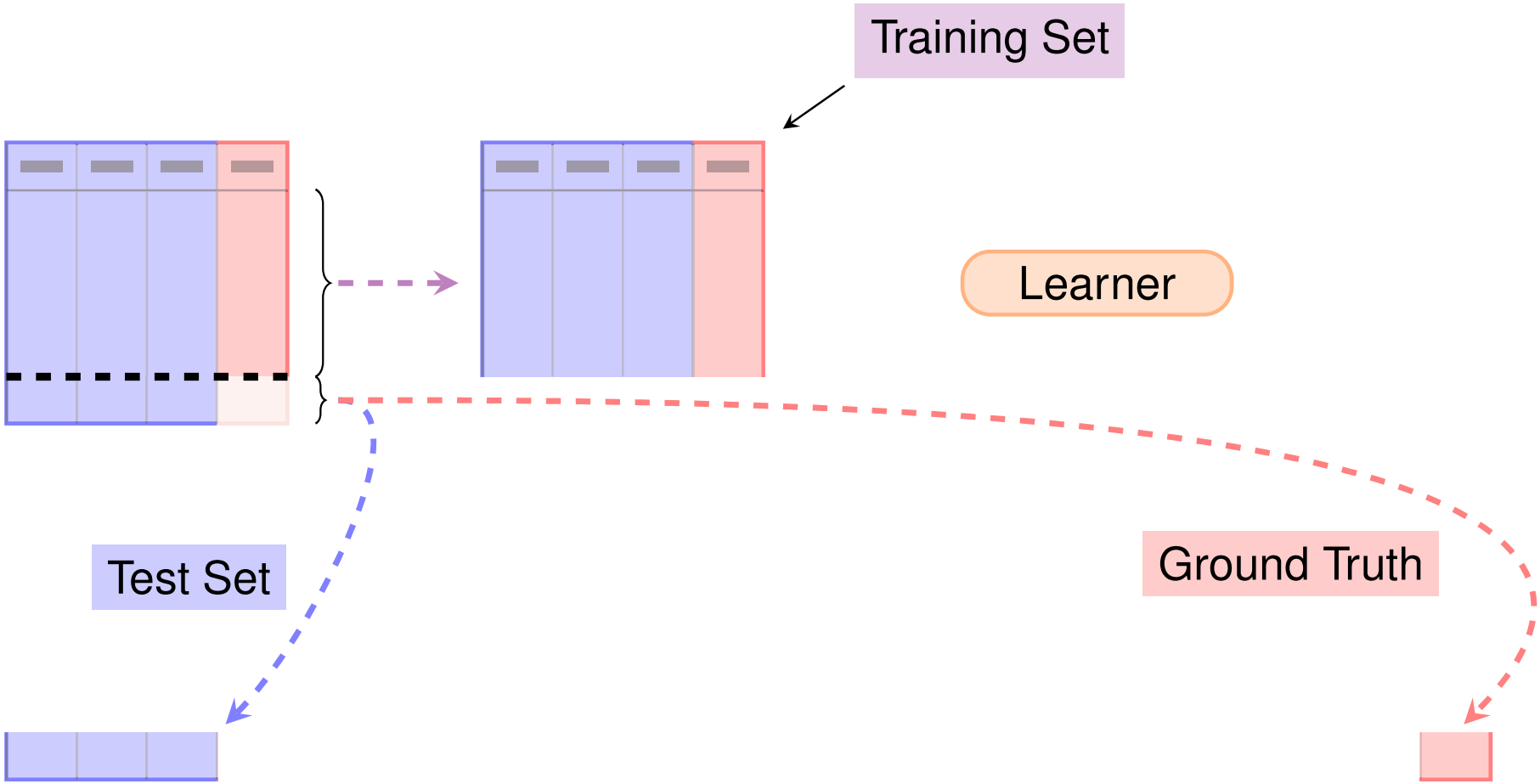
RESAMPLING



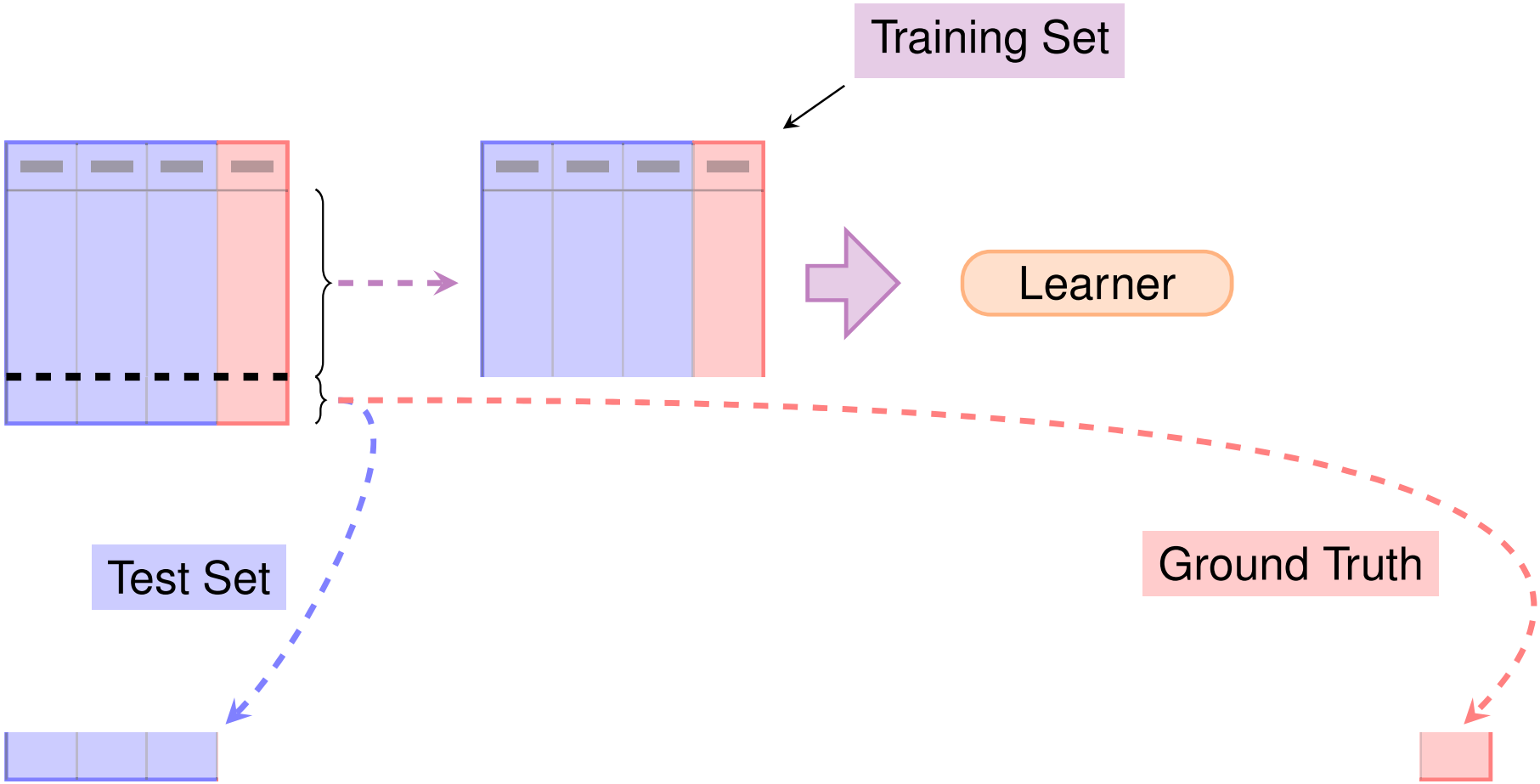
RESAMPLING



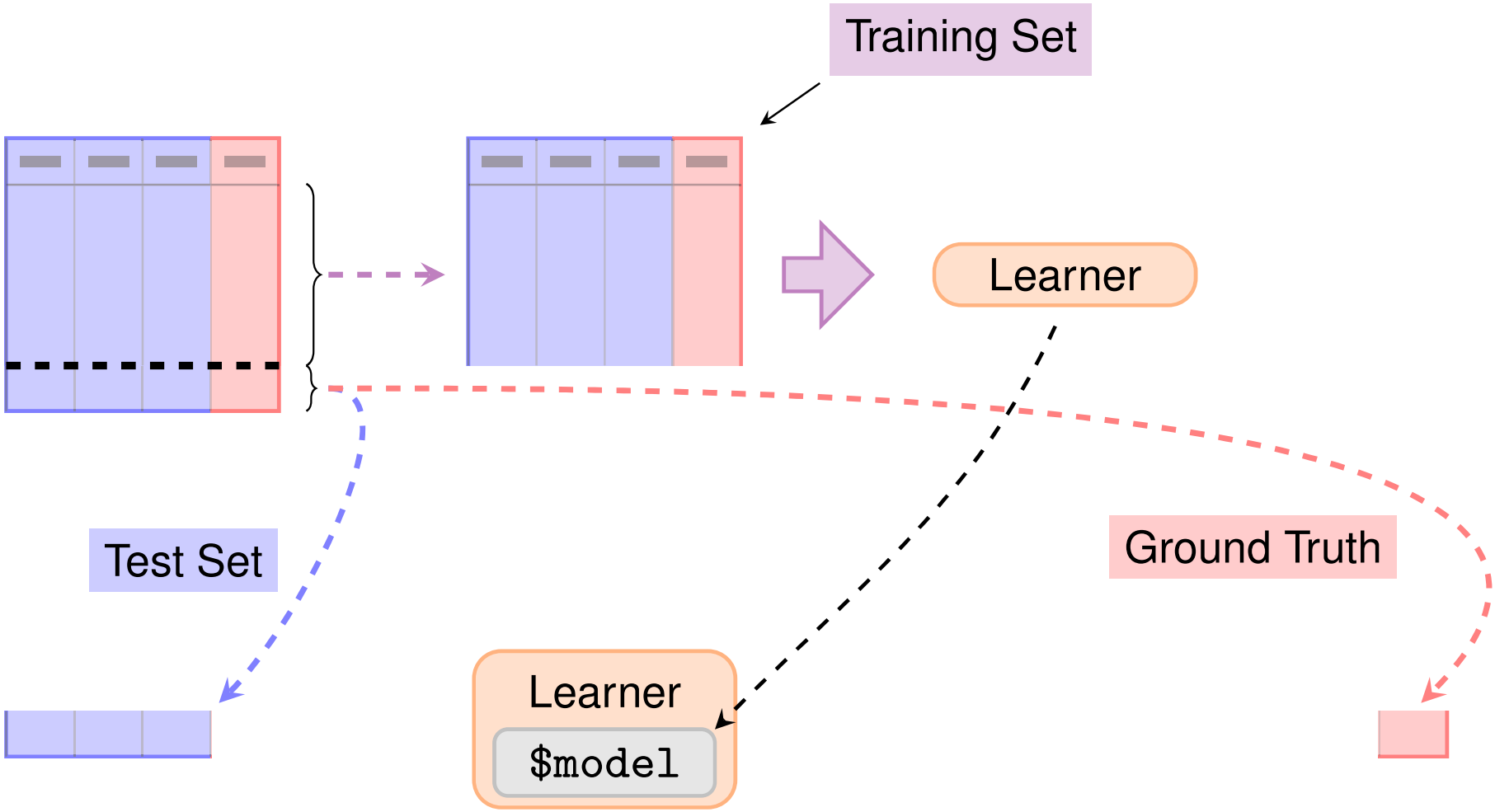
RESAMPLING



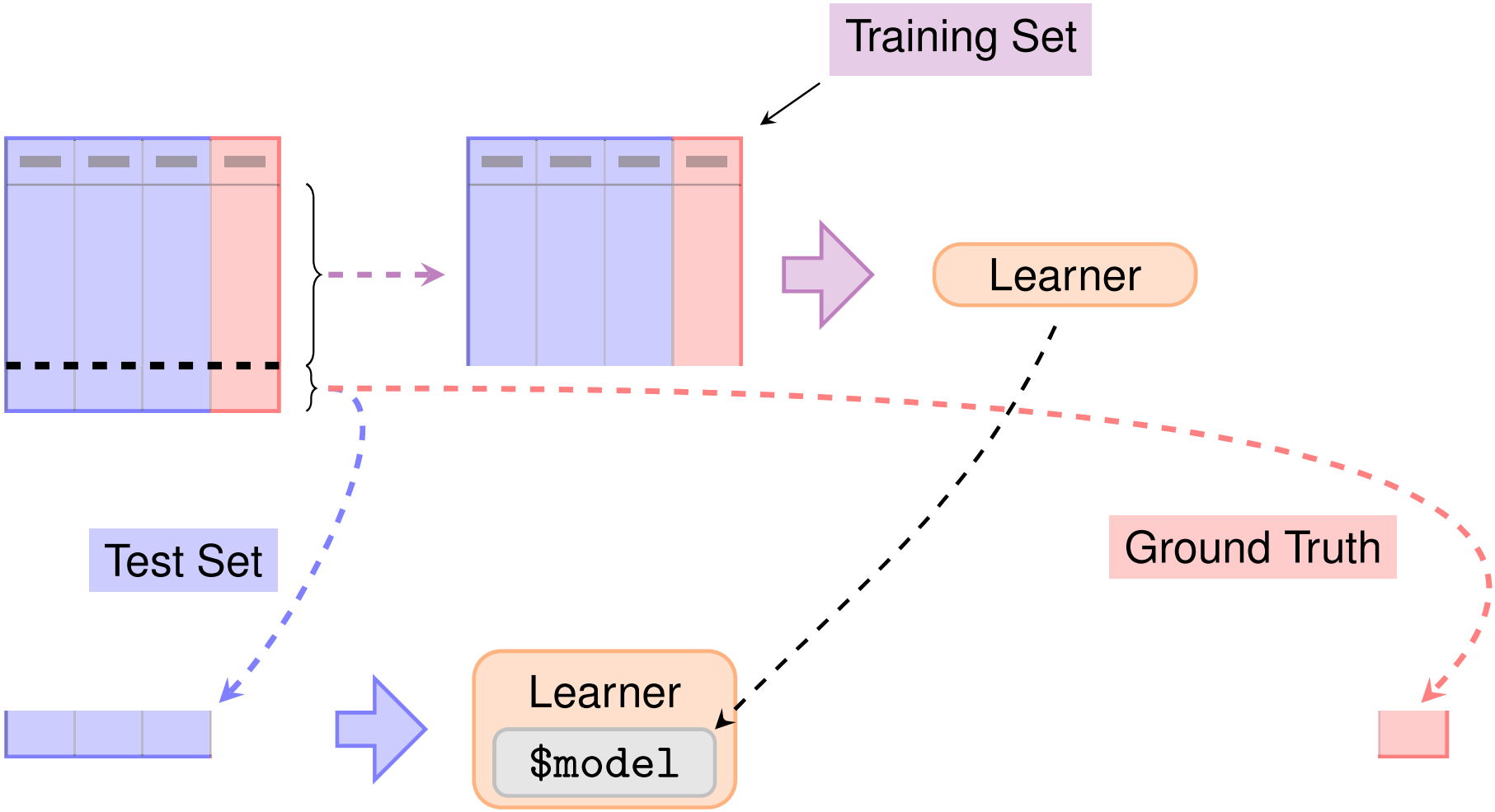
RESAMPLING



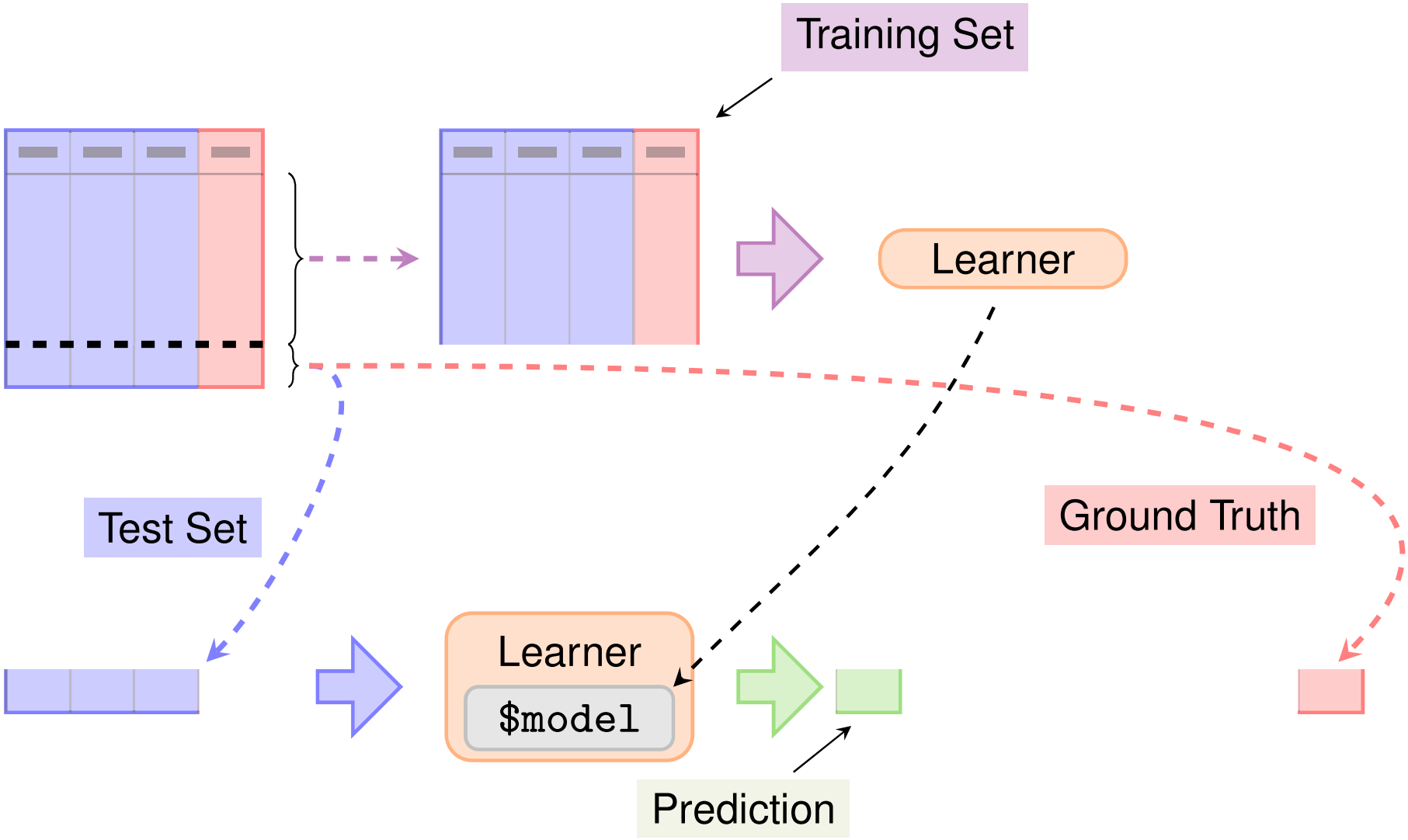
RESAMPLING



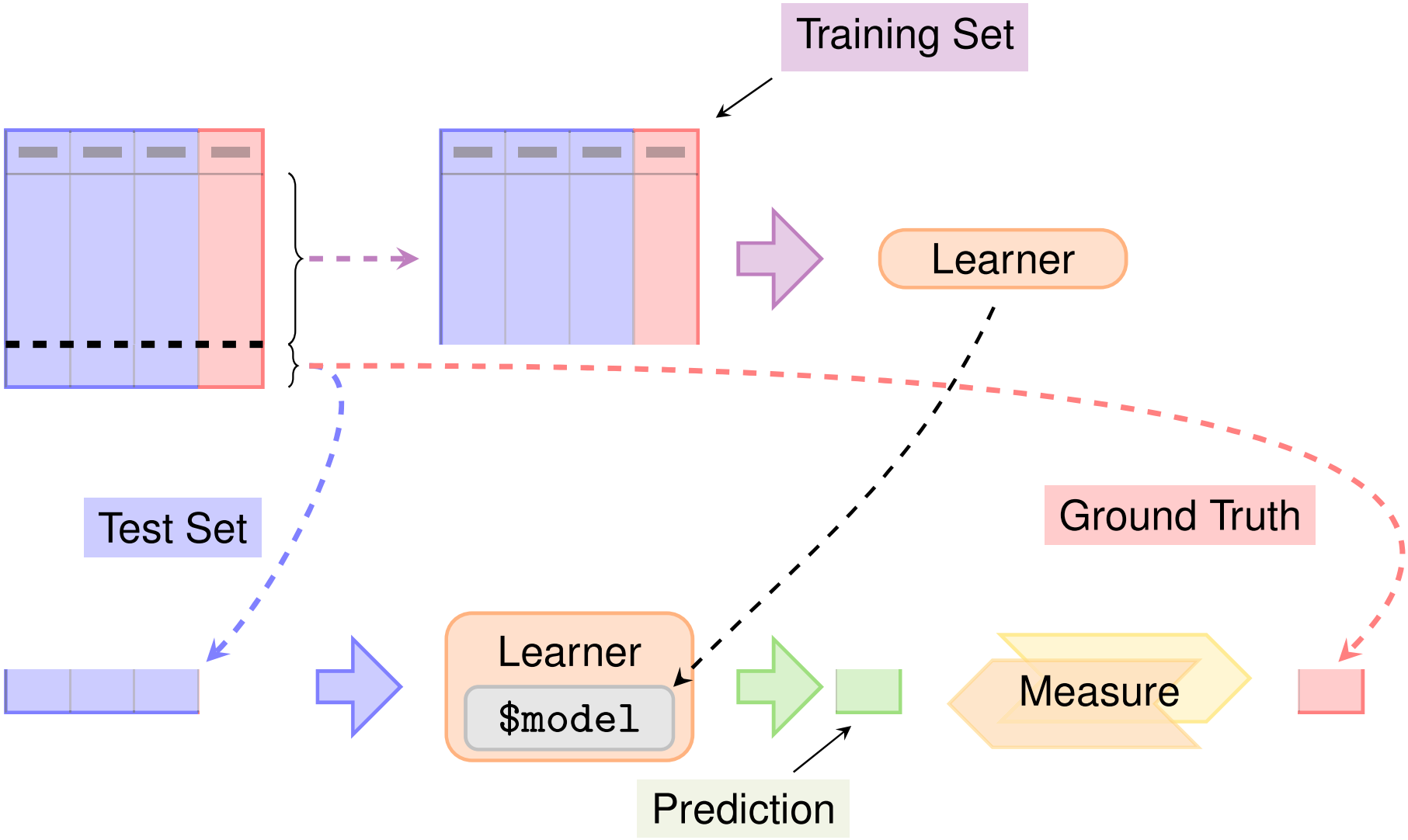
RESAMPLING



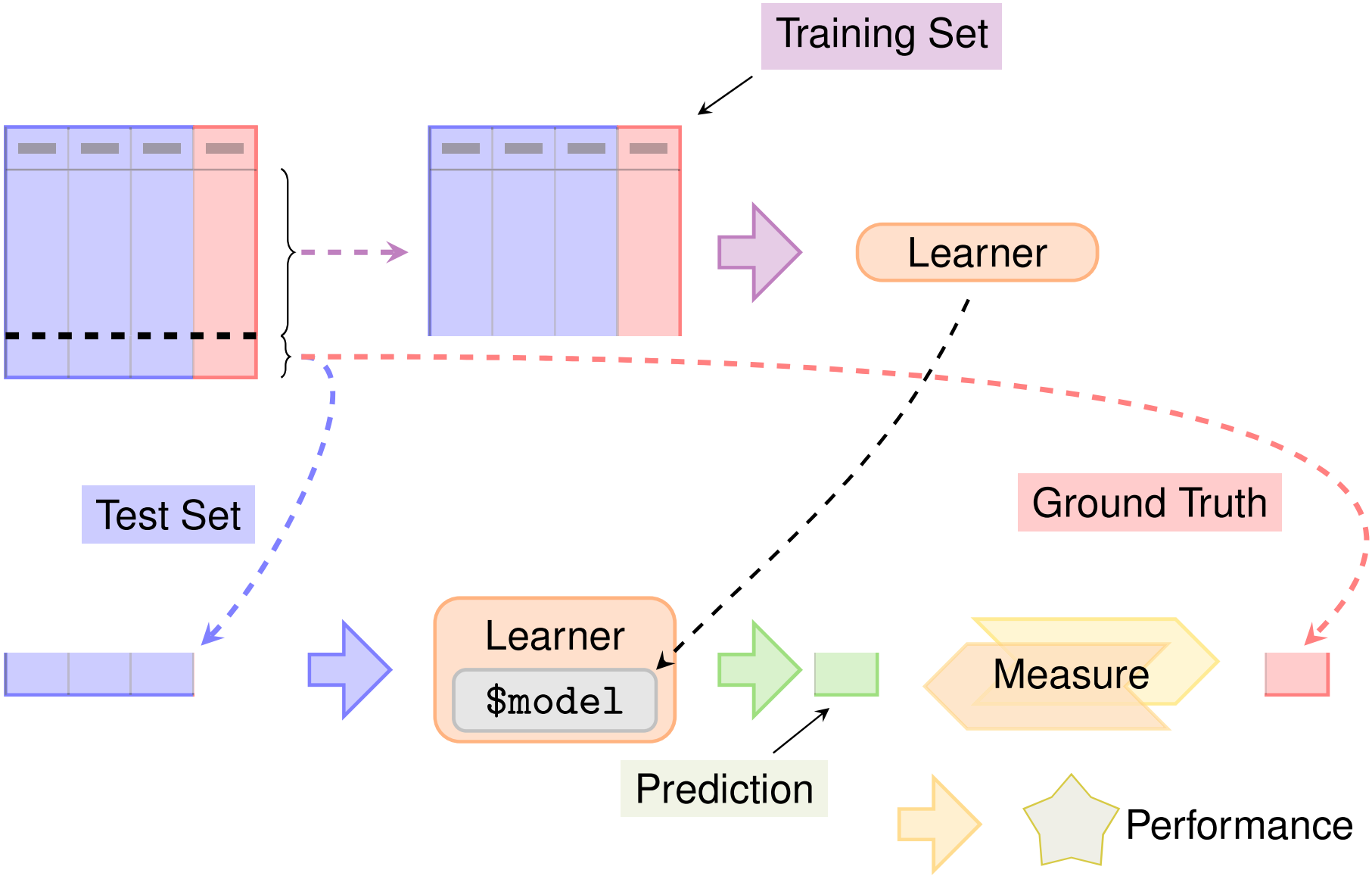
RESAMPLING


Training Set
Learner
Test Set
Ground Truth
Learner
$model
Prediction

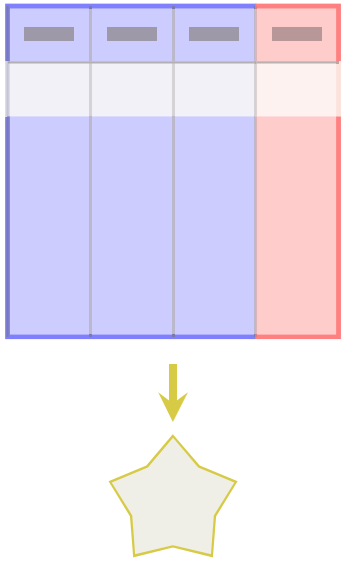
RESAMPLING


Training Set
Learner
Test Set
Ground Truth
Learner
$model
Measure
Prediction

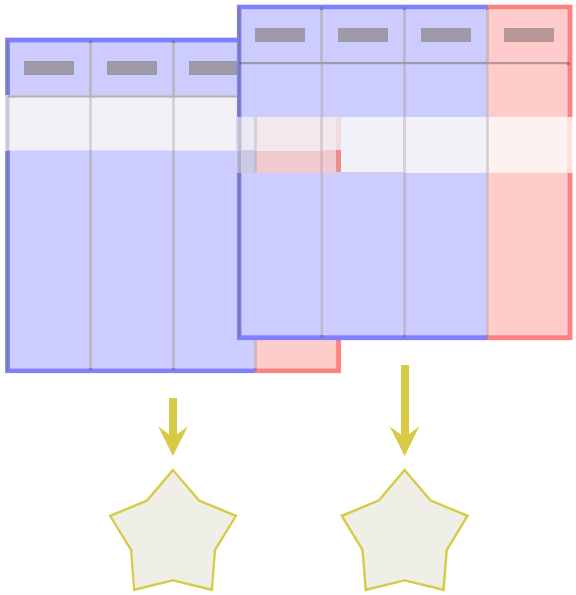
RESAMPLING


Training Set
Learner
Test Set
Ground Truth
Learner
$model
Measure
Prediction
Performance

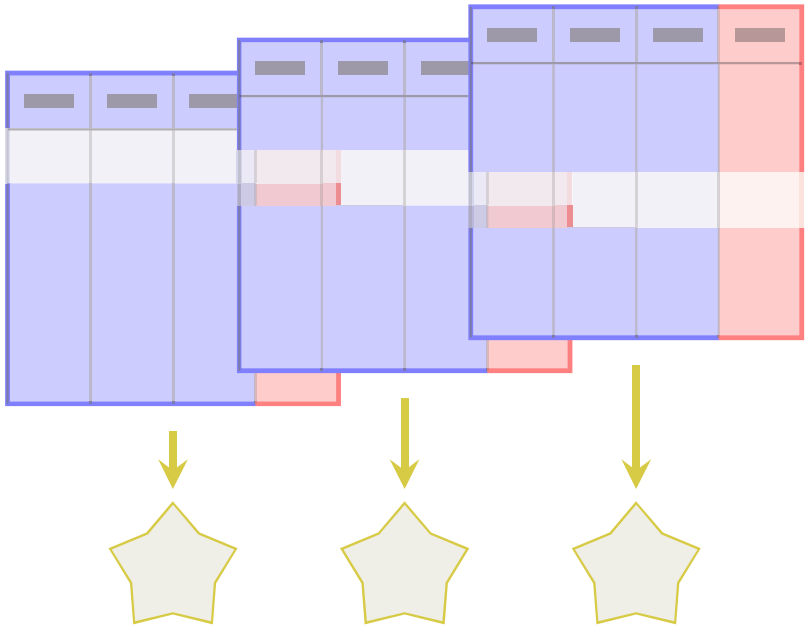
RESAMPLING



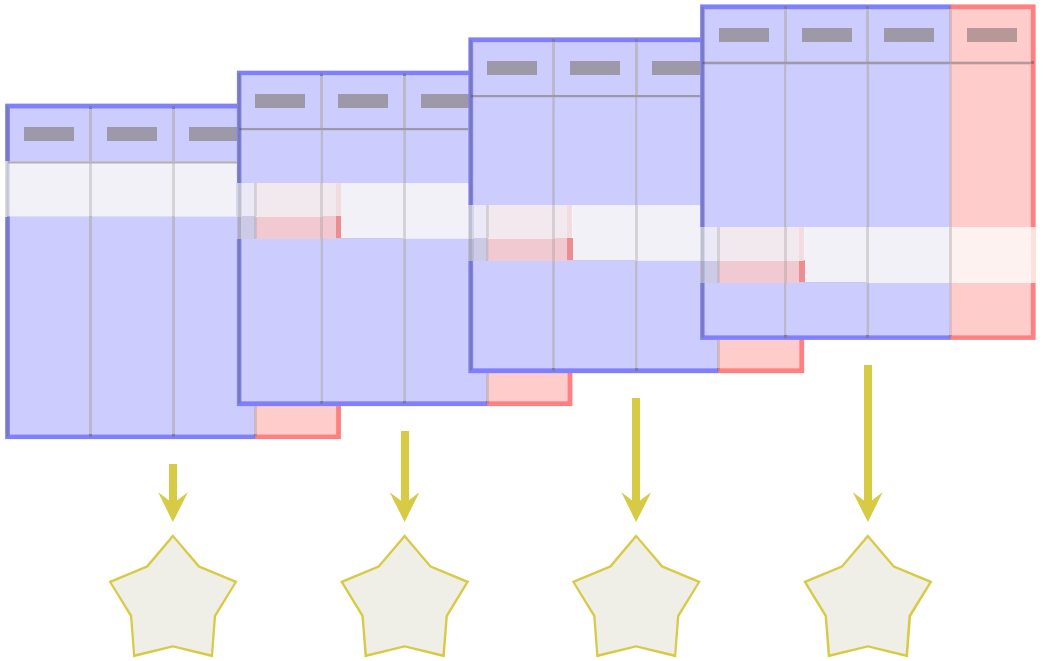
RESAMPLING



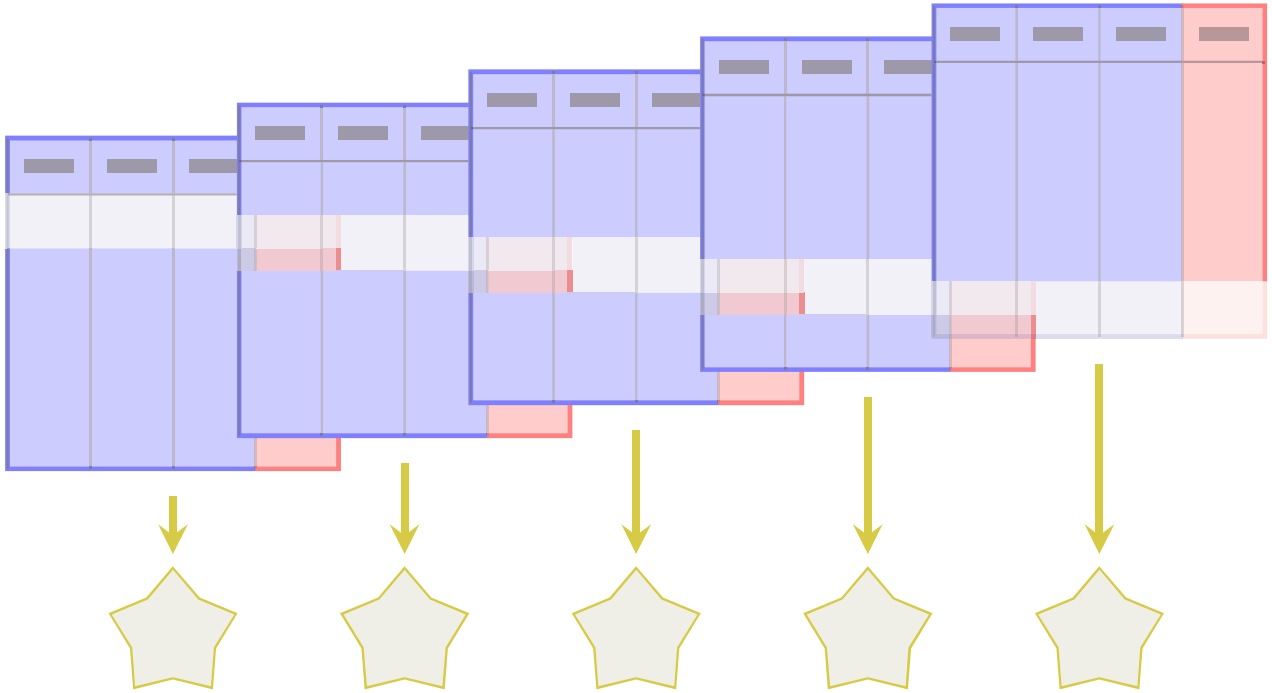
RESAMPLING



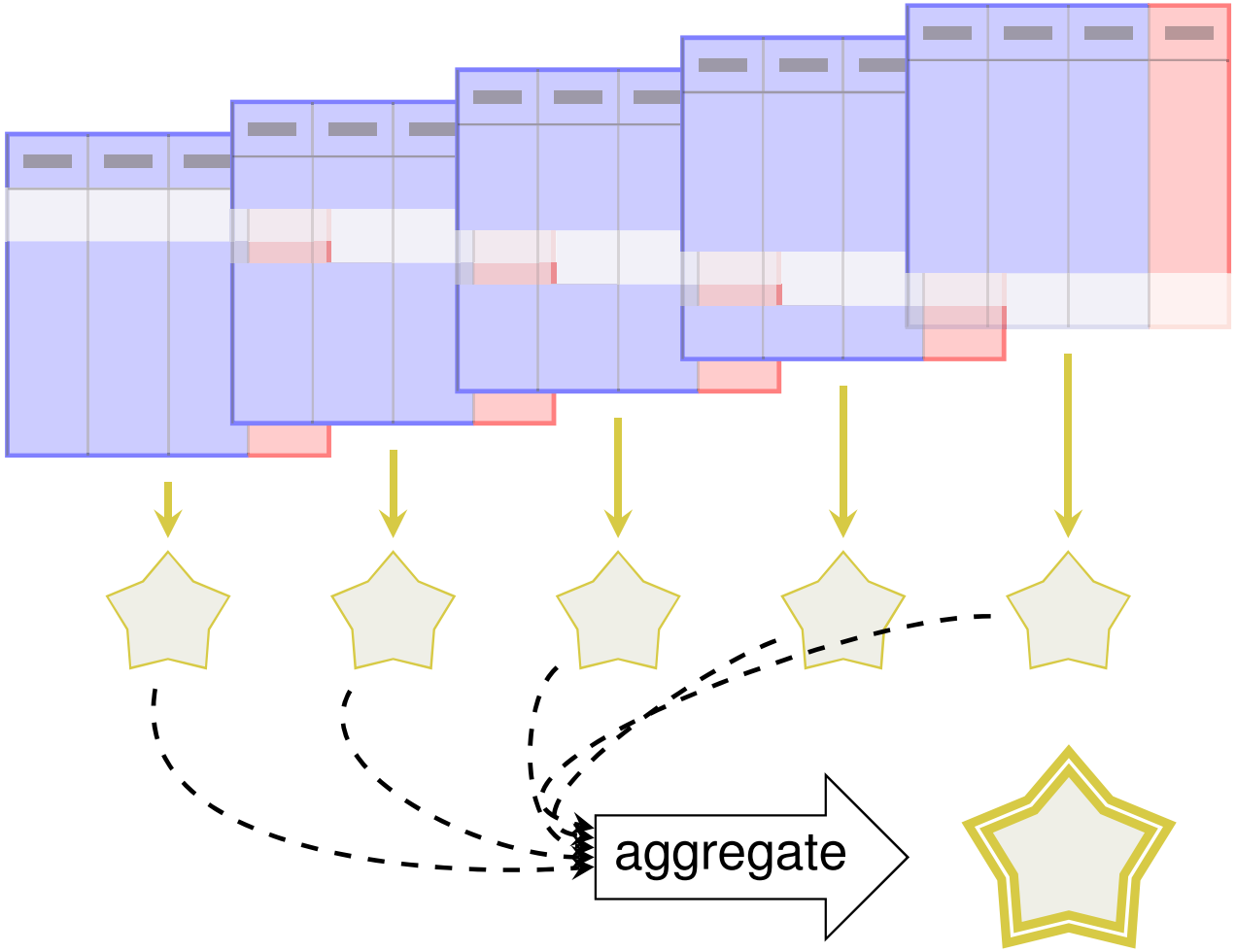
RESAMPLING



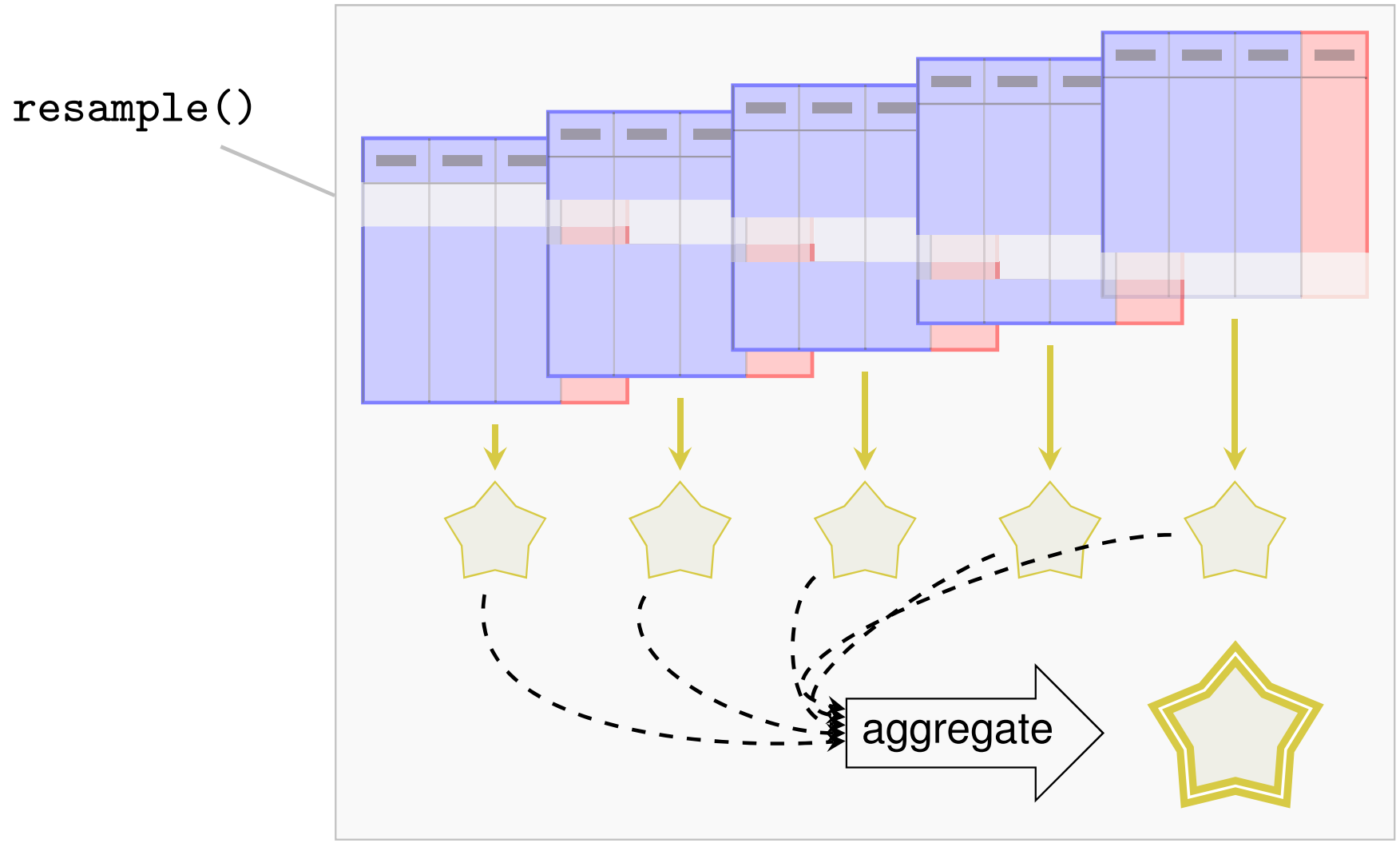
RESAMPLING



RESAMPLING



RESAMPLING



RESAMPLING

- Resample description: How to split the data

```
cv5 = rsmp("cv", folds = 5)
```

RESAMPLING

- Resample description: How to split the data

```
cv5 = rsmp("cv", folds = 5)
```

- Use the `resample()` function for resampling:

```
rr = resample(task, learner, cv5)
```

RESAMPLING

- Resample description: How to split the data

```
cv5 = rsmp("cv", folds = 5)
```

- Use the `resample()` function for resampling:

```
rr = resample(task, learner, cv5)
```

- We get a `ResamplingResult` object:

```
print(rr)
#> <ResampleResult> of 5 iterations
#> * Task: iris
#> * Learner: classif.rpart
#> * Warnings: 0 in 0 iterations
#> * Errors: 0 in 0 iterations
```

RESAMPLING RESULTS

What exactly is a `ResamplingResult` object?

RESAMPLING RESULTS

What exactly is a `ResamplingResult` object?
Remember `Prediction`:

RESAMPLING RESULTS

What exactly is a `ResamplingResult` object?
Remember `Prediction`:

- Get a table representation using `as.data.table()`

```
rr_table = as.data.table(rr)

print(rr_table)
#                  task                     learner         resampling i...
# 1: <TaskClassif[44]> <LearnerClassifRpart[32]> <ResamplingCV[19]>  ...
# 2: <TaskClassif[44]> <LearnerClassifRpart[32]> <ResamplingCV[19]>  ...
# 3: <TaskClassif[44]> <LearnerClassifRpart[32]> <ResamplingCV[19]>  ...
# 4: <TaskClassif[44]> <LearnerClassifRpart[32]> <ResamplingCV[19]>  ...
# 5: <TaskClassif[44]> <LearnerClassifRpart[32]> <ResamplingCV[19]>  ...
```

RESAMPLING RESULTS

What exactly is a `ResamplingResult` object?
Remember `Prediction`:

- Get a table representation using `as.data.table()`

```
rr_table = as.data.table(rr)

print(rr_table)
#                  task                      learner         resampling i...
# 1: <TaskClassif[44]> <LearnerClassifRpart[32]> <ResamplingCV[19]>  ...
# 2: <TaskClassif[44]> <LearnerClassifRpart[32]> <ResamplingCV[19]>  ...
# 3: <TaskClassif[44]> <LearnerClassifRpart[32]> <ResamplingCV[19]>  ...
# 4: <TaskClassif[44]> <LearnerClassifRpart[32]> <ResamplingCV[19]>  ...
# 5: <TaskClassif[44]> <LearnerClassifRpart[32]> <ResamplingCV[19]>  ...
```

- Active bindings and functions that make information easily accessible

RESAMPLING RESULTS

- Calculate performance:

```
rr$aggregate(msr("classif.ce"))
#> classif.ce
#>      0.073
```

RESAMPLING RESULTS

- Calculate performance:

```
rr$aggregate(msr("classif.ce"))
#> classif.ce
#>      0.073
```

- Get predictions

```
rr$prediction()
#> <PredictionClassif> for 150 observations:
#>     row_id     truth  response
#>          5    setosa    setosa
#>         14    setosa    setosa
#>         18    setosa    setosa
#> ---
#>        139 virginica virginica
#>        145 virginica virginica
#>        146 virginica virginica
```

RESAMPLING

- Predictions of individual folds

```
predictions = rr$predictions()
predictions[[1]]
```

```
#> <PredictionClassif> for 30 observations:
#>     row_id     truth  response
#>          5    setosa    setosa
#>         14    setosa    setosa
#>         18    setosa    setosa
#> ---
#>        132 virginica virginica
#>        137 virginica virginica
#>        147 virginica virginica
```

RESAMPLING

- Predictions of individual folds

```
predictions = rr$predictions()
predictions[[1]]
```

```
#> <PredictionClassif> for 30 observations:
#>     row_id     truth  response
#>          5    setosa    setosa
#>         14    setosa    setosa
#>         18    setosa    setosa
#> ---
#>        132 virginica virginica
#>        137 virginica virginica
#>        147 virginica virginica
```

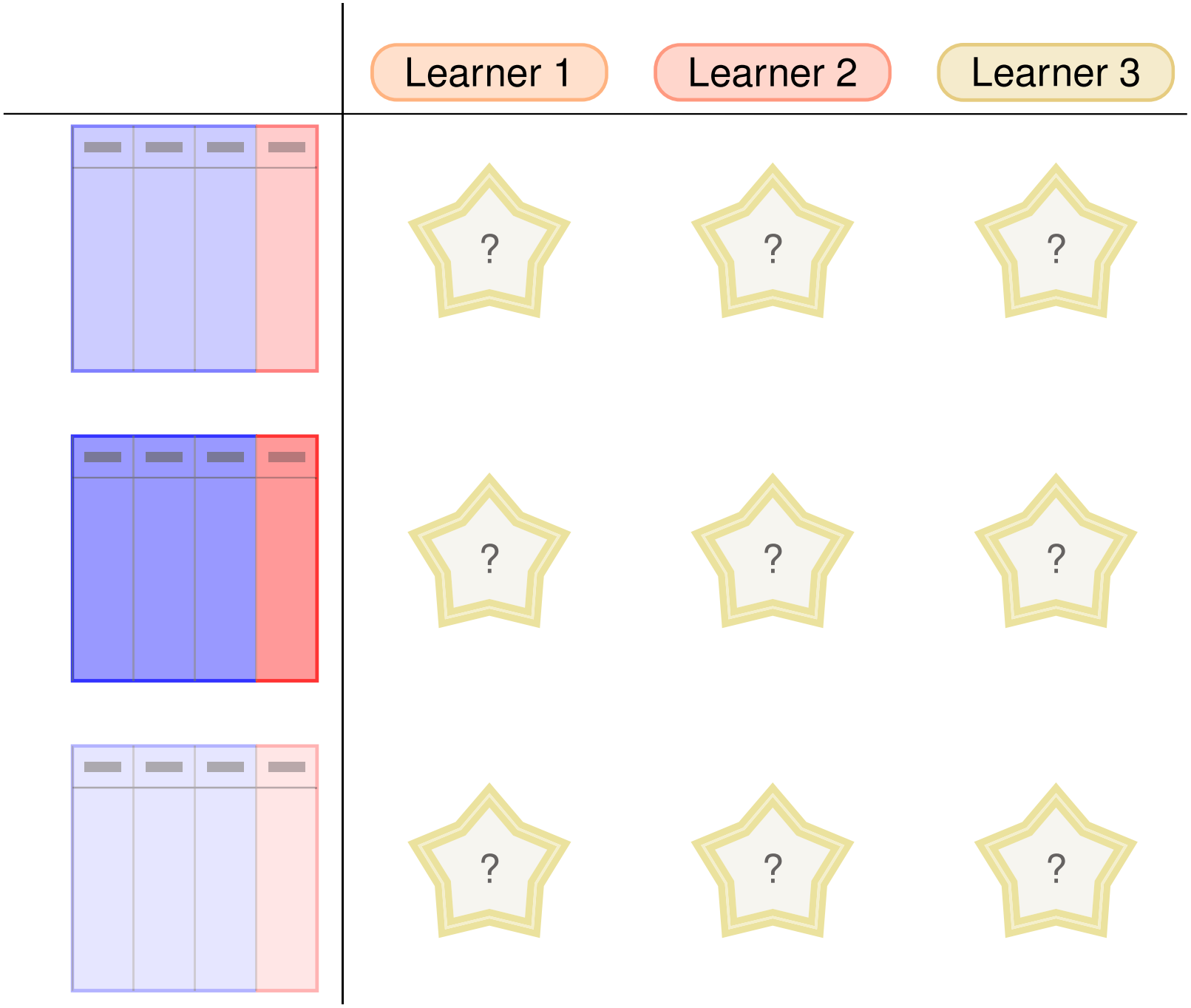
- Score of individual folds

```
scores = rr$score()
scores[1:3, c("iteration", "classif.ce")]
```

```
#>    iteration classif.ce
#> 1:         1      0.033
#> 2:         2      0.033
#> 3:         3      0.100
```

Benchmark

PERFORMANCE COMPARISON



PERFORMANCE COMPARISON

- Multiple Learners, multiple Tasks:

```
library("mlr3learners")
learners = list(lrn("classif.rpart"), lrn("classif.kknn"))
tasks = list(tsk("iris"), tsk("sonar"), tsk("wine"))
```

PERFORMANCE COMPARISON

- Multiple Learners, multiple Tasks:

```
library("mlr3learners")
learners = list(lrn("classif.rpart"), lrn("classif.kknn"))
tasks = list(tsk("iris"), tsk("sonar"), tsk("wine"))
```

- Set up the *design* and execute benchmark:

```
design = benchmark_grid(tasks, learners, cv5)
bmr = benchmark(design)
```

PERFORMANCE COMPARISON

- Multiple Learners, multiple Tasks:

```
library("mlr3learners")
learners = list(lrn("classif.rpart"), lrn("classif.kknn"))
tasks = list(tsk("iris"), tsk("sonar"), tsk("wine"))
```

- Set up the *design* and execute benchmark:

```
design = benchmark_grid(tasks, learners, cv5)
bmr = benchmark(design)
```

- We get a `BenchmarkResult` object which shows that `kknn` outperforms `rpart`:

```
bmr_ag = bmr$aggregate()
bmr_ag[, c("task_id", "learner_id", "classif.ce")]
#>    task_id    learner_id classif.ce
#> 1:    iris classif.rpart      0.060
#> 2:    iris  classif.kknn      0.060
#> 3:   sonar classif.rpart      0.279
#> 4:   sonar  classif.kknn      0.168
#> 5:    wine classif.rpart      0.101
#> 6:    wine  classif.kknn      0.051
```

BENCHMARK RESULT

What exactly is a `BenchmarkResult` object?

BENCHMARK RESULT

What exactly is a `BenchmarkResult` object?
Just like `Prediction` and `ResamplingResult`!

BENCHMARK RESULT

What exactly is a `BenchmarkResult` object?
Just like `Prediction` and `ResamplingResult`!

- Table representation using `as.data.table()`

BENCHMARK RESULT

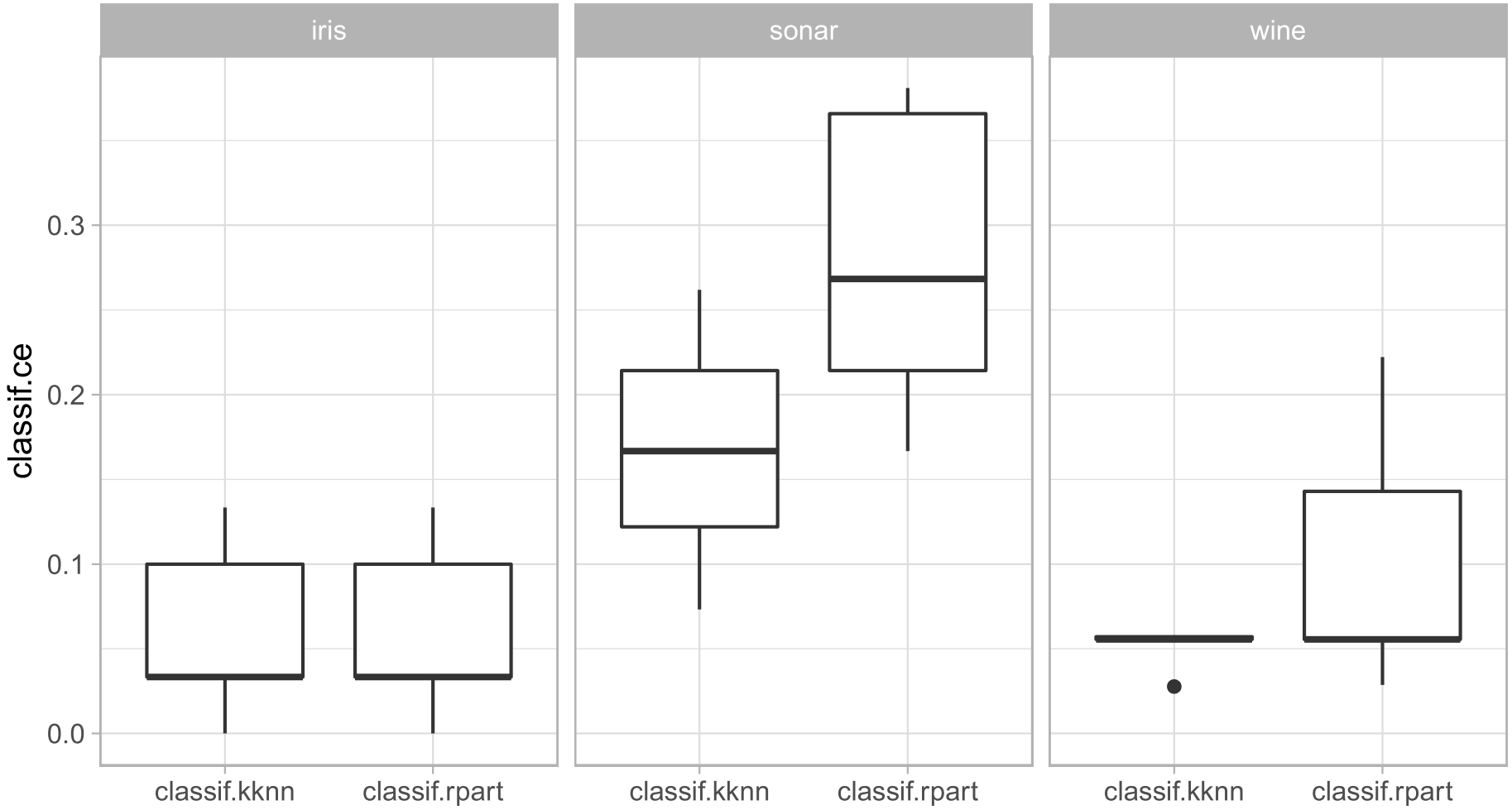
What exactly is a `BenchmarkResult` object?
Just like `Prediction` and `ResamplingResult`!

- Table representation using `as.data.table()`
- Active bindings and functions that make information easily accessible

BENCHMARK RESULT

The `mlr3viz` package contains `autoplot()` functions for many mlr3 objects

```
library(mlr3viz)
autoplot(bmr)
```



Control of Execution

CONTROL OF EXECUTION

Parallelization

```
future::plan("multicore")
```

- runs each resampling iteration as a job
- also allows nested resampling (although not needed here)

Encapsulation

```
learner$encapsulate = c(train = "callr", predict = "callr")
```

- Spawns a separate R process to train the learner
- Learner may segfault without tearing down the session
- Logs are captured
- Possibilty to have a fallback to create predictions

How to get Help

HOW TO GET HELP

- Where to start?
  - Check these slides
  - **Check the mlr3book https://mlr3book.mlr-org.com**

HOW TO GET HELP

- Where to start?
  - Check these slides
  - **Check the mlr3book https://mlr3book.mlr-org.com**
- Get help for R6 objects?
  1. Find out what kind of R6 object you have:

```
class(bmr)
#> [1] "BenchmarkResult" "R6"
```

  2. Go to the corresponding help page:

```
?BenchmarkResult
```

  New: open the corresponding man page with

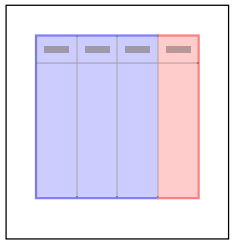
```
learner$help()
```

Outro

OVERVIEW

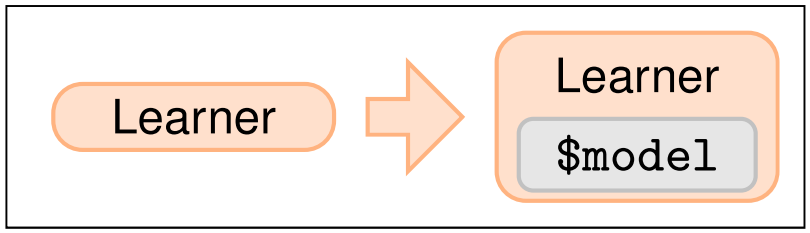
Ingredients:

Data



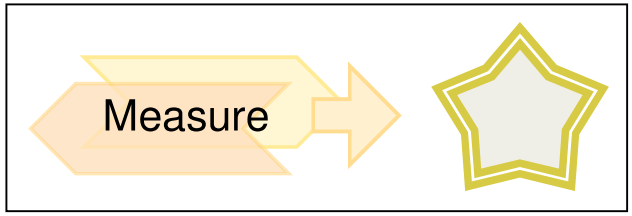
`TaskClassif`,
`TaskRegr`,
`tsk()`

Learning Algorithms



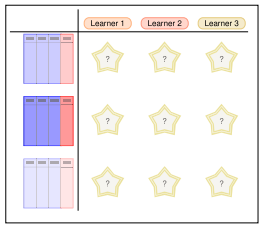
`lrn()` ⇒ `Learner`,
`$train()`,
`$predict()` ⇒ `Prediction`

Performance Evaluation

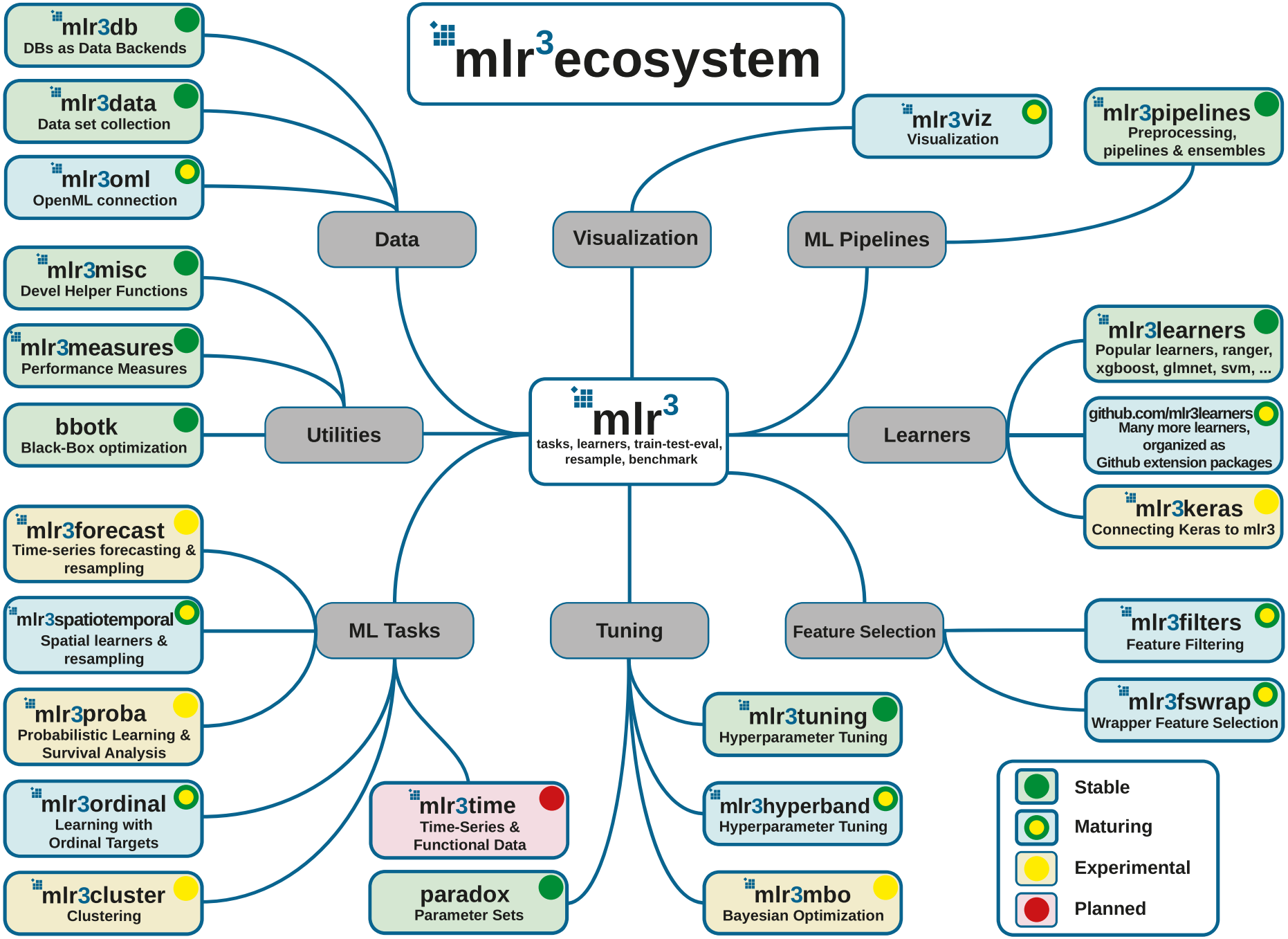


`rsmp()` ⇒ `Resampling`,
`msr()` ⇒ `Measure`,
`resample()` ⇒ `ResamplingResult`,
`$aggregate()`

Performance Comparison



`benchmark_grid()`,
`benchmark()` ⇒ `BenchmarkResult`


mlr3ecosystem
mlr3db
DBs as Data Backends
mlr3data
Data set collection
mlr3oml
OpenML connection
mlr3misc
Devel Helper Functions
mlr3measures
Performance Measures
bbotk
Black-Box optimization
mlr3forecast
Time-series forecasting & resampling
mlr3spatiotemporal
Spatial learners & resampling
mlr3proba
Probabilistic Learning & Survival Analysis
mlr3ordinal
Learning with Ordinal Targets
mlr3cluster
Clustering
Data
Utilities
ML Tasks
Visualization
Tuning
ML Pipelines
Learners
Feature Selection
mlr3
tasks, learners, train-test-eval, resample, benchmark
mlr3viz
Visualization
mlr3pipelines
Preprocessing, pipelines & ensembles
mlr3learners
Popular learners, ranger, xgboost, glmnet, svm, ...
github.com/mlr3learners
Many more learners, organized as Github extension packages
mlr3keras
Connecting Keras to mlr3
mlr3filters
Feature Filtering
mlr3fswrap
Wrapper Feature Selection
mlr3time
Time-Series & Functional Data
paradox
Parameter Sets
mlr3tuning
Hyperparameter Tuning
mlr3hyperband
Hyperparameter Tuning
mlr3mbo
Bayesian Optimization
Stable
Maturing
Experimental
Planned