

# Package ‘RNAmoDR.ML’

December 24, 2024

**Type** Package

**Title** Detecting patterns of post-transcriptional modifications using machine learning

**Version** 1.20.0

**Date** 2021-11-21

**Description** RNAmoDR.ML extend the functionality of the RNAmoDR package and classical detection strategies towards detection through machine learning models. RNAmoDR.ML provides classes, functions and an example workflow to establish a detection strategy, which can be packaged.

**biocViews** Software, Infrastructure, WorkflowStep, Visualization, Sequencing

**License** Artistic-2.0

**Encoding** UTF-8

**LazyData** false

**Depends** R (>= 3.6), RNAmoDR

**Imports** methods, BiocGenerics, S4Vectors, IRanges, GenomicRanges, stats, ranger

**Suggests** BiocStyle, knitr, rmarkdown, testthat, RNAmoDR.Data, RNAmoDR.AlkAnilineSeq, GenomicFeatures, Rsamtools, rtracklayer, keras

**Collate** 'RNAmoDR.ML.R' 'AllGenerics.R' 'ModifierMLModel-class.R' 'ModifierML-class.R' 'ModifierML-data.R' 'ModifierMLkeras-class.R' 'ModifierMLranger-class.R' 'RNAmoDR.ML.example.R' 'utils.R'

**RoxygenNote** 7.1.1

**VignetteBuilder** knitr

**BugReports** <https://github.com/FelixErnst/RNAmoDR.ML/issues>

**URL** <https://github.com/FelixErnst/RNAmoDR.ML>

**git\_url** <https://git.bioconductor.org/packages/RNAmoDR.ML>

**git\_branch** RELEASE\_3\_20

**git\_last\_commit** ebd785f

**git\_last\_commit\_date** 2024-10-29

**Repository** Bioconductor 3.20

**Date/Publication** 2024-12-23

**Author** Felix G.M. Ernst [aut, cre] (<<https://orcid.org/0000-0001-5064-0928>>),  
Denis L.J. Lafontaine [ctb]

**Maintainer** Felix G.M. Ernst <[felix.gm.ernst@outlook.com](mailto:felix.gm.ernst@outlook.com)>

## Contents

ModifierMLkeras-class . . . . .	2
ModifierMLranger-class . . . . .	3
RNAmodR.ML . . . . .	4
RNAmodR.ML-datasets . . . . .	4
RNAmodR.ML-example . . . . .	5
setMLModel<- . . . . .	5
trainingData . . . . .	7
useModel . . . . .	8

**Index** **10**

---

ModifierMLkeras-class *ModifierMLkeras class*

---

## Description

The ModifierMLkeras class extends the virtual class ModifierMLModel and unifies the access to Keras machine learning models used in the detection of post-transcriptional modifications in RNA sequencing data. The ModifierMLkeras class is virtual itself and must be extended from for each individual machine learning model.

The ModifierMLkeras class extends the virtual class ModifierMLModel and unifies the access to Keras machine learning models used in the detection of post-transcriptional modifications in RNA sequencing data. The ModifierMLkeras class is virtual itself and must be extended from for each individual machine learning model.

Since a stored model needs to be loaded from file, the additional slot modelFile is used and can be accessed through a function of the same name. Upon creation of a ModifierMLkeras object, the model is loaded from file, if modelFile is not an empty character value and a valid file name.

## Usage

```
## S4 method for signature 'ModifierMLkeras,ModifierML'
useModel(x, y)
```

## Arguments

x                    a ModifierMLkeras object  
y                    a ModifierML object

## Value

a ModifierMLkeras object

**Slots**

modelFile a character vector of length == 1L, which describes a model to load via [load\\_model\\_hdf5](#).  
The model is then stored in the model slot.

**See Also**

[ModifierMLModel](#)

---

ModifierMLranger-class

*ModifierMLranger class*

---

**Description**

The ModifierMLranger class extends the virtual class ModifierMLModel and unifies the access to [ranger](#) machine learning models used in the detection of post-transcriptional modifications in RNA sequencing data. The ModifierMLranger class is virtual itself and must be extended from for each individual machine learning model.

**Usage**

```
## S4 method for signature 'ModifierMLranger,ModifierML'  
useModel(x, y)
```

**Arguments**

x a ModifierMLranger object  
y a ModifierML object

**Value**

a ModifierMLranger object

**See Also**

[ModifierMLModel](#)

**Examples**

```
# example class derived from the virtual ModifierMLranger class  
setClass("ModifierMExample",  
        contains = c("ModifierMLranger"))  
ModifierMExample <- function(...){  
  new("ModifierMExample")  
}  
mlmodel <- ModifierMExample()
```

RNAmodR.ML

*RNAmodR.ML*

---

**Description**

To extend the functionality of the RNAmodR package and classical detection strategies towards detection through machine learning models, RNAmodR.ML provides classes and an example workflow.

**Author(s)**

Felix G M Ernst [aut]

**See Also**

[RNAmodR](#) package.

---

RNAmodR.ML-datasets*Example data in the RNAmodR.ML package*

---

**Description**

This contains an example ModifierSet object

**Usage**

```
data(dmod)
```

```
data(mod7)
```

```
data(me)
```

```
data(model)
```

**Format**

a GRanges object containing D positions

a GRanges object containing m7G positions

a ModMLExample object for examples

a ranger object for examples

---

RNAmodR.ML-example	<i>RNAmodR.ML functions for example</i>
--------------------	---

---

### Description

The exported functions here are used in the vignette as examples. If you want to reuse them, please implement them yourself. This will allow for fine tuning on your side and does not create a dependency to example functions, which could change

### Usage

```
aggregate_example(x)

find_mod_example(x, minCoverage)
```

### Arguments

x	a ModifierML object
minCoverage	the minimum coverage for finding modifications

### Value

- `aggregate_example` a `SplitDataFrameList` object containing the aggregated data for a `Modifier` object
- `find_mod_example` a `GRanges` object with coordinates of found modifications

### Examples

```
# no examples runnable. See vignette for explanation on how to use these
# example functions
aggregate_example
find_mod_example
```

---

<code>setMLModel&lt;-</code>	<i>The ModifierML class</i>
------------------------------	-----------------------------

---

### Description

The `ModifierML` class is a virtual class, which provides the central functionality for searching with a machine learning models for patterns of post-transcriptional RNA modifications in high throughput sequencing data.

It extends the virtual `Modifier` class from the `RNAmodR` package and add the `useMLModel` function. If not called directly for a `ModifierML` class, the `useMLModel` will be called from the `aggregate` function.

The slot `m1Model` is added and serves a dual purpose. If `m1Model` is a character, a class of the type `ModifierMLModel` is created upon creation of a `ModifierML` object. However, for developing purposes the slot can also remain empty and a `ModifierMLModel` object can be set using the `setMLModel` function and retrieved using the `getMLModel`. If the `m1Model` slot is empty, the `findMod` setting will be set to `FALSE` and the `ModifierML` object will be returned just with the aggregate data. Such an object can then be used to train a machine learning model. The data can be accessed using [trainingData](#).

**Usage**

```

setMLModel(x) <- value

getMLModel(x)

hasMLModel(x)

useMLModel(x)

## S4 replacement method for signature 'ModifierML'
setMLModel(x) <- value

## S4 method for signature 'ModifierML'
getMLModel(x)

## S4 method for signature 'ModifierML'
hasMLModel(x)

## S4 method for signature 'ModifierML'
aggregate(x, force = FALSE)

## S4 method for signature 'ModifierML'
useMLModel(x)

## S4 method for signature 'ModifierML'
modify(x, force = FALSE)

```

**Arguments**

x	a ModifierML object.
value	a ModifierMLModel object
force	whether to recreate the aggregated data, if it is already stored inside the Modifier object.

**Value**

a ModifierML object

**Slots**

mlModel a character describing a class name for creating a ModifierMLModel object or a ModifierMLModel object itself. If mlModel is a character, the class will tried to be create by calling a function of the same name.

**See Also**

[Modifier](#)

**Examples**

```

# an example implementation of a ModifierML object
setClass("ModMExample",
        contains = c("RNAModifierML"),

```

```

prototype = list(mod = c("D"),
                 score = "score",
                 dataType = c("PileupSequenceData",
                              "CoverageSequenceData"),
                 mlModel = character(0))
# constructor function for ModMLExample
ModMLExample <- function(x, annotation = NA, sequences = NA, seqinfo = NA,
                        ...){
  RNAmodR:::Modifier("ModMLExample", x = x, annotation = annotation,
                    sequences = sequences, seqinfo = seqinfo, ...)
}

```

---

trainingData

*Assemble training data from aggregate sequence data*


---

## Description

trainingData subsets the aggregate data of a ModifierML object to positions as defined by coord. Positions with an entry in the mod column are labeled TRUE.

## Usage

```

trainingData(x, coord, ...)

## S4 method for signature 'ModifierML,GRanges'
trainingData(x, coord, ...)

## S4 method for signature 'ModifierML,GRangesList'
trainingData(x, coord, ...)

```

## Arguments

x	a ModifierML object
coord	a GRanges or a GRangesList object
...	See <a href="#">subsetByCoord</a> for more details. type is hard coded to FALSE to disregard subsetting by type. In addition, merge is set to FALSE by default, but it can be set to TRUE.

## Value

a CompressedSplitDataFrameList with aggregate data and an addition label column.

## See Also

For more details have a look at [subsetByCoord](#).

## Examples

```

data("dmod",package = "RNAmodR.ML")
setClass("ModMLExample",
  contains = c("RNAmodifierML"),
  prototype = list(mod = c("D"),
    score = "score",
    dataType = c("PileupSequenceData",
      "CoverageSequenceData"),
    mlModel = character(0)))
data("me",package = "RNAmodR.ML")
nextUPos <- function(gr){
  nextU <- lapply(seq.int(1L,2L),
    function(i){
      subseq <- subseq(RNAmodR::sequences(me)[dmod$Parent], start(dmod)+3L)
      pos <- start(dmod) + 2L +
        vapply(strsplit(as.character(subseq),""),
          function(y){which(y == "U")[i]},integer(1))
      ans <- dmod
      ranges(ans) <- IRanges(start = pos, width = 1L)
      ans
    })
  nextU <- do.call(c,nextU)
  nextU$mod <- NULL
  unique(nextU)
}
nondmod <- nextUPos(dmod)
nondmod <- nondmod[!(nondmod %in% dmod)]
coord <- unique(c(dmod,nondmod))
coord <- coord[order(as.integer(coord$Parent))]
trainingData(me,coord)

```

---

useModel

*ModifierMLModel virtual class*

---

## Description

The `ModifierMLModel` is a virtual class and is used for representing different types of machine learning models used in the detection of post transcriptional modifications in RNA sequencing data.

The next class inheriting from here should only implement a certain type of model and also be virtual. The grand child of the `ModifierMLModel` class should then implement a specific model for detecting certain types of modifications.

## Usage

```
useModel(x, y)
```

```
## S4 method for signature 'ModifierMLModel,ANY'
useModel(x, y)
```

## Arguments

`x` a `ModifierMLModel` object  
`y` See [ModifierMLranger](#) for an example



**Value**

a ModifierMLModel object

**Slots**

model a machine learning object of any type

**See Also**

[ModifierMLranger](#) [ModifierMLkeras](#)

[ModifierMLranger](#) for an example implementation

**Examples**

```
# an example implementation of a ModifierMLModel object using the
# ModifierMLModel derived class ModifierMLranger
setClass("ModifierMExample",
        contains = c("ModifierMLranger"))
ModifierMExample <- function(...){
  new("ModifierMExample")
}
```

# Index

- \* **datasets**
  - RNAmodR.ML-datasets, 4
- aggregate, ModifierML-method
  - (setMLModel<-), 5
- aggregate\_example (RNAmodR.ML-example), 5
- calculate\_correct\_base\_score
  - (RNAmodR.ML-example), 5
- dmod (RNAmodR.ML-datasets), 4
- DNAModifierML-class (setMLModel<-), 5
- find\_mod\_example (RNAmodR.ML-example), 5
- getMLModel (setMLModel<-), 5
- getMLModel, ModifierML-method
  - (setMLModel<-), 5
- hasMLModel (setMLModel<-), 5
- hasMLModel, ModifierML-method
  - (setMLModel<-), 5
- load\_model\_hdf5, 3
- me (RNAmodR.ML-datasets), 4
- mod7 (RNAmodR.ML-datasets), 4
- model (RNAmodR.ML-datasets), 4
- Modifier, 6
- ModifierML (setMLModel<-), 5
- ModifierML-class (setMLModel<-), 5
- ModifierMLkeras, 9
- ModifierMLkeras
  - (ModifierMLkeras-class), 2
- ModifierMLkeras-class, 2
- ModifierMLModel, 3
- ModifierMLModel (useModel), 8
- ModifierMLModel-class (useModel), 8
- ModifierMLranger, 8, 9
- ModifierMLranger
  - (ModifierMLranger-class), 3
- ModifierMLranger-class, 3
- modify, ModifierML-method
  - (setMLModel<-), 5
- ranger, 3
- RNAModifierML-class (setMLModel<-), 5
- RNAmodR, 4
- RNAmodR.ML, 4
- RNAmodR.ML-datasets, 4
- RNAmodR.ML-example, 5
- RNAmodR.ML.example
  - (RNAmodR.ML-example), 5
- setMLModel<- , 5
- setMLModel<- , ModifierML-method
  - (setMLModel<-), 5
- subsetByCoord, 7
- trainingData, 5, 7
- trainingData, ModifierML, GRanges-method
  - (trainingData), 7
- trainingData, ModifierML, GRangesList-method
  - (trainingData), 7
- useMLModel (setMLModel<-), 5
- useMLModel, ModifierML-method
  - (setMLModel<-), 5
- useModel, 8
- useModel, ModifierMLkeras, ModifierML-method
  - (ModifierMLkeras-class), 2
- useModel, ModifierMLModel, ANY-method
  - (useModel), 8
- useModel, ModifierMLranger, ModifierML-method
  - (ModifierMLranger-class), 3