

# Package ‘signalHsmm’

May 9, 2026

**Type** Package

**Title** Predict Presence of Signal Peptides

**Version** 1.5

**LazyData** true

**Date** 2018-11-11

**Description** Predicts the presence of signal peptides in eukaryotic protein using hidden semi-Markov models. The implemented algorithm can be accessed from both the command line and GUI.

**License** GPL-3

**URL** <https://github.com/michbur/signalhsmm>

**Depends** R (>= 3.0.0)

**Imports** graphics, seqinr, shiny, stats, utils

**LinkingTo** Rcpp

**Suggests** DT, rmarkdown, shinythemes

**NeedsCompilation** yes

**Repository** CRAN

**Encoding** UTF-8

**RoxygenNote** 6.1.0

**Author** Michal Burdukiewicz [cre, aut] (ORCID:  
<<https://orcid.org/0000-0001-8926-582X>>),  
Piotr Sobczyk [aut],  
Jaroslaw Chilimoniuk [ctb] (ORCID:  
<<https://orcid.org/0000-0001-5467-018X>>)

**Maintainer** Michal Burdukiewicz <michalburdukiewicz@gmail.com>

**Date/Publication** 2018-11-15 09:10:03 UTC

## Contents

aaaggregation . . . . .	2
add_k_mer_state . . . . .	3
benchmark_dat . . . . .	4
duration_viterbi . . . . .	4
find_nhc . . . . .	5
gui_signalHsmm . . . . .	6
hsmm_pred . . . . .	6
hsmm_pred_list . . . . .	7
is_protein . . . . .	7
plot.hsmm_pred . . . . .	8
pred2df . . . . .	8
predict.sighsmm_model . . . . .	9
read_txt . . . . .	9
read_uniprot . . . . .	10
run_signalHsmm . . . . .	10
signalHsmm . . . . .	11
summary.hsmm_pred . . . . .	12
summary.hsmm_pred_list . . . . .	13
train_hsmm . . . . .	13
<b>Index</b>	<b>14</b>

---

aaaggregation	<i>Reduced amino acid alphabet</i>
---------------	------------------------------------

---

### Description

Amino acids are grouped together in larger sets based on their physicochemical properties important in the recognition of signal peptide.

### Usage

```
aaaggregation
```

### Format

a list of length four. Each element contains a character vector of amino acid names (one-letter abbreviations).

---

add_k_mer_state	<i>Adds k-mer hidden state to signalHsmm model</i>
-----------------	--

---

**Description**

Changes parameters for Hidden Semi-Markov Model to add k-mer

**Usage**

```
add_k_mer_state(kMer, pipar, tpmpar, od, params, pState, nState, pTrans, d)
```

**Arguments**

kMer	character vector representing k-mer aminoacid sequence.
pipar	Probabilities of initial state in Markov Model.
tpmpar	Matrix with transition probabilities between states.
od	Matrix of response probabilities. Eg. od[1,2] is a probability of signal 2 in state 1.
params	Matrix of probability distribution for duration. Eg. params[10,2] is probability of duration of time 10 in state 2.
pState	number denoting hidden state right before k-mer.
nState	number denoting hidden state right after k-mer.
pTrans	Probability of change from pState to k-mer hidden state.
d	Duration of the state.

**Value**

A list of length four:

- pipar a vector of new probabilities of initial state in Markov Model,
- tpmpar a matrix with new transition probabilities between states,
- od matrix of new response probabilities,
- params matrix of new probability distributions for duration.

**Note**

Currently add only k-mers without distance.

---

benchmark_dat	<i>Benchmark data set for signalHsmm</i>
---------------	--

---

**Description**

Lists eukaryotic proteins added to UniProt database release 2015\_06 between 1.01.2010 and 1.06.2015 (140 proteins with signal peptide and 280 randomly sampled proteins without signal peptide).

**Usage**

```
benchmark_dat
```

**Format**

a list of [SeqFastaAA](#) objects. Slot sig contains the range of signal peptide (if any).

**Source**

[UniProt](#)

**Examples**

```
summary(benchmark_dat)
```

---

duration_viterbi	<i>Compute most probable path with extended Viterbi algorithm.</i>
------------------	--

---

**Description**

Viterbi algorithm for Hidden Markov Model with duration

**Usage**

```
duration_viterbi(aa_sample, pipar, tpmpr, od, params)
```

**Arguments**

aa_sample	character vector representing single aminoacid sequence.
pipar	probabilities of initial state in Markov Model.
tpmpr	matrix of transition probabilities between states.
od	matrix of response probabilities. Eg. od[1,2] is a probability of signal 2 in state 1.
params	matrix of probability distribution for duration. Eg. params[10,2] is probability of duration of time 10 in state 2.

**Value**

A list of length four:

- path a vector of most probable path
- viterbi values of probability in all intermediate points,
- psi matrix that gives for every signal and state the previous state in viterbi path,
- duration matrix that gives for every signal and state gives the duration in that state on viterbi path.

**Note**

All computations are on logarithms of probabilities.

---

find\_nhc

*Localize n-, h- and c-region in signal peptide*

---

**Description**

Finds borders between distinct regions constituting signal peptides using a heuristic algorithm.

**Usage**

```
find_nhc(protein, signal = NULL)
```

**Arguments**

protein            a vector of amino acids or object of class [SeqFastaAA](#).

signal            range of signal peptide. If NULL, the attribute sig of protein will be used.

**Value**

a vector of length 4 containing positions of:

1. start of n-region,
2. start of h-region,
3. start of c-region,
4. cleavage site.

**References**

Henrik Nielsen, Anders Krogh (1998). Prediction of signal peptides and signal anchors by a hidden Markov model. *Proc. Sixth Int. Conf. on Intelligent Systems for Molecular Biology*.

---

gui_signalHsmm	<i>GUI for signalHsmm</i>
----------------	---------------------------

---

**Description**

A graphical user interface for predicting presence of signal peptides.

**Usage**

```
gui_signalHsmm()
```

**Value**

null.

**Note**

Any ad-blocking software may be cause of malfunctions.

**See Also**

[run\\_signalHsmm](#)

---

hsmm_pred	<i>hsmm_pred class</i>
-----------	------------------------

---

**Description**

A single prediction of signalHsmm.

A stochastic model of signal peptide produced by signalHsmm.

**Details**

Always a named list of five elements

1. `sp_probability` is a probability of signal peptide presence.
2. `sp_start` is a start of potential signal peptide (naively 1 aminoacid).
3. `sp_end` is a position of last amino acid of signal peptide.
4. `struc` is numeric vector representing predicted structure of input protein.
5. `prot` is character vector containing input sequence of amino acids.
6. `str_approx` has value bigger than 0 if the predicted signal peptide structure was approximated (usually in case of sequences that have no signal peptides).

Always a named list of five elements

1. aa\_group encoding of amino acids. See [aaaggregation](#) for an example.
2. pipar probabilities of initial state in Markov Model.
3. tpmpr matrix of transition probabilities between states.
4. od matrix of response probabilities. Eg. od[1,2] is a probability of signal 2 in state 1.
5. overall\_probs\_log probabilities of amino acids in mature protein.
6. params matrix of probability distribution for duration. Eg. params[10,2] is probability of duration of time 10 in state 2.

**See Also**

[summary.hsmm\\_pred](#) [plot.hsmm\\_pred](#)  
[train\\_hsmm](#) [predict.sighsmm\\_model](#)

---

<code>hsmm_pred_list</code>	<i>hsmm_pred_list class</i>
-----------------------------	-----------------------------

---

**Description**

A list of prediction(s) generated by [run\\_signalHsmm](#) function.

**Details**

A named list. Each element belongs to the [hsmm\\_pred](#) class.

**See Also**

[summary.hsmm\\_pred\\_list](#), [pred2df](#)

---

<code>is_protein</code>	<i>Protein test</i>
-------------------------	---------------------

---

**Description**

Checks if an object is a protein (contains letters from one-letter amino acid code).

**Usage**

```
is_protein(object)
```

**Arguments**

`object` character vector where each elements represent one amino acid.

**Value**

TRUE or FALSE.

---

plot.hsmm_pred	<i>Plot single signalHsmm prediction</i>
----------------	--

---

**Description**

Plots objects of class [hsmm\\_pred](#).

**Usage**

```
## S3 method for class 'hsmm_pred'  
plot(x, add_legend = TRUE, only_sure = TRUE, ...)
```

**Arguments**

x	object of class <a href="#">hsmm_pred</a> .
add_legend	logical, if TRUE, legend is added to the plot.
only_sure	logical, if FALSE does not draw signal peptide structure when probability is smaller than 0.5.
...	ignored.

**Value**

Nothing.

---

pred2df	<i>Convert list of signalHsmm predictions</i>
---------	---

---

**Description**

Converts objects of class [hsmm\\_pred\\_list](#) to data frame.

**Usage**

```
pred2df(object)
```

**Arguments**

object	of class <a href="#">hsmm_pred_list</a> .
--------	---

**Value**

Data frame which columns contain respectively the probability of signal peptide presence as well as the start and the end of predicted signal peptide.

---

predict.sighsmm\_model *Predict sighsmm\_model object*

---

### Description

Predicts the presence of signal peptides using signalHsmm models.

### Usage

```
## S3 method for class 'sighsmm_model'
predict(object, newdata, ...)
```

### Arguments

object	sighsmm_model object.
newdata	unknown sequence of class character or character. Alternatively, a list of sequences in mentioned formats.
...	further arguments passed to or from other methods.

### Examples

```
#remember to remove it
## Not run:
pos_train_ultrahard <- read_uniprot("pos_ultrahard_data.txt", euk = TRUE)
model1 <- train_hsmm(pos_train_ultrahard, aa_group = aaaggregation)
predict(model1, benchmark_dat[1L:5])

## End(Not run)
```

---

read\_txt *Read sequences from .txt file*

---

### Description

Read sequence data saved in text file.

### Usage

```
read_txt(connection)
```

### Arguments

connection	a <a href="#">connection</a> to the text (.txt) file.
------------	---

### Details

The input file should contain one or more amino acid sequences separated by empty line(s).

**Value**

a list of sequences. Each element has class [SeqFastaAA](#). If connection contains no characters, function prompts warning and returns NULL.

---

read_uniprot	<i>Read data from UniProt database</i>
--------------	--

---

**Description**

Read data saved in UniProt original flat text format.

**Usage**

```
read_uniprot(connection, ft_names, kwds = NULL)
```

**Arguments**

connection	a <a href="#">connection</a> to UniProt data in text format.
ft_names	a character vector of UniProt features to be extracted, for example "signal", "transit", "propep". The case is not matched.
kwds	a NULL or character vector of keywords (not UniProt keywords, but words of interest, that may occur in the protein description).

**Value**

a list of sequences. Each element has a class [SeqFastaAA](#). Attributes OS and OC represents respectively OS and OC fields in the protein description. A value of each feature is preserved as an attribute named after the feature.

---

run_signalHsmm	<i>Predict presence of signal peptide in protein</i>
----------------	--

---

**Description**

Using the hidden semi-Markov model predict presence of signal peptide in eukaryotic proteins.

**Usage**

```
run_signalHsmm(test_data)
```

**Arguments**

test_data	single protein sequence (character vector) or list of sequences. It may be an object of class <a href="#">SeqFastaAA</a> .
-----------	--

**Details**

Function `signalHsmm` returns respectively probability of presence of signal peptide, start of signal peptide and the probable cleavage site localization. If input consists of more than one sequence, result is a data.frame where each column contains above values for different proteins.

**Value**

An object of class `hsmm_pred_list`.

**Note**

Currently start of signal peptide is naively set as 1 amino acid. The prediction of a cleavage site is still an experimental feature, use on your own risk.

**See Also**

[hsmm\\_pred\\_list](#) [hsmm\\_pred](#)

**Examples**

```
#run signalHsmm on one sequence
x1 <- run_signalHsmm(benchmark_dat[[1]])

#run signalHsmm on one sequence, but input is a character vector
x2 <- run_signalHsmm(c("M", "A", "G", "K", "E", "V", "I", "F", "I", "M", "A", "L",
" F", "I", "A", "V", "E", "S", "S", "P", "I", "F", "S", "F", "D",
" D", "L", "V", "C", "P", "S", "V", "T", "S", "L", "R", "V", "N",
" V", "E", "K", "N", "E", "C", "S", "T", "K", "K", "D", "C", "G",
" R", "N", "L", "C", "C", "E", "N", "Q", "N", "K", "I", "N", "V",
" C", "V", "G", "G", "I", "M", "P", "L", "P", "K", "P", "N", "L",
" D", "V", "N", "N", "I", "G", "G", "A", "V", "S", "E", "S", "V",
" K", "Q", "K", "R", "E", "T", "A", "E", "S", "L"))

#run signalHsmm on list of sequences
x3 <- run_signalHsmm(benchmark_dat[1:3])
#see summary of results
summary(x3)
#print results as data frame
pred2df(x3)
#summary one result
summary(x3[[1]])
plot(x3[[1]])
```

---

signalHsmm

*signalHsmm - prediction of signal peptides*

---

**Description**

Using hidden semi-Markov models as a probabilistic framework, `signalHsmm` is new, highly accurate signal peptide predictor for eukaryotic proteins.

## Details

Secretory signal peptides are short (20-30 residues) N-terminal amino acid sequences tagging among others tag among others hormones, immune system proteins, structural proteins, and metabolic enzymes. They direct a protein to the endomembrane system and next to the extracellular localization. All signal peptides possess three distinct domains with variable length and characteristic amino acid composition. Despite their variability, signal peptides are universal enough to direct properly proteins in different secretory systems. For example, artificially introduced bacterial signal peptides can guide proteins in mammals and plants.

The development of signalHsmm was funded by National Science Center (2015/17/N/NZ2/01845).

## Examples

```
few_predictions <- run_signalHsmm(benchmark_dat[1:3])
#see all predictions
pred2df(few_predictions)
#summary one prediction
summary(few_predictions[[1]])
#plot one prediction
plot(few_predictions[[1]])

#have fun with GUI
## Not run:
gui_signalHsmm()

## End(Not run)
```

---

summary.hsmm_pred	<i>Summarize single signalHsmm prediction</i>
-------------------	---

---

## Description

Summarizes objects of class [hsmm\\_pred](#).

## Usage

```
## S3 method for class 'hsmm_pred'
summary(object, only_sure = TRUE,
         double_linebreak = FALSE, ...)
```

## Arguments

object	of class <a href="#">hsmm_pred</a> .
only_sure	logical, if FALSE does not draw signal peptide structure when probability is smaller than 0.5.
double_linebreak	logical, if TRUE adds aestically pleasing (in .Rmd) double linebreaks between the elements of summary.
...	ignored

**Value**

Nothing.

---

```
summary.hsmm_pred_list
```

*Summarize list of signalHsmm predictions*

---

**Description**

Summarizes objects of class [hsmm\\_pred\\_list](#).

**Usage**

```
## S3 method for class 'hsmm_pred_list'
summary(object, ...)
```

**Arguments**

object	of class <a href="#">hsmm_pred_list</a> .
...	ignored

**Value**

nothing.

---

```
train_hsmm
```

*Train sighsmm\_model object*

---

**Description**

Train sighsmm\_model object

**Usage**

```
train_hsmm(train_data, aa_group, max_length = 32,
            region_fun = find_nhc)
```

**Arguments**

train_data	training data.
aa_group	method of aggregating amino acids.
max_length	maximum length of signal peptide.
region_fun	function defining borders of regions (see <a href="#">find_nhc</a> ).

**Value**

object of class `sighsmm_model`.

# Index

- \* **classif**
  - run\_signalHsmm, 10
- \* **datasets**
  - aaaggregation, 2
  - benchmark\_dat, 4
- \* **hplot**
  - plot.hsmm\_pred, 8
- \* **manip**
  - pred2df, 8
  - read\_txt, 9
  - read\_uniprot, 10
  - summary.hsmm\_pred\_list, 13
- \* **methods**
  - plot.hsmm\_pred, 8
  - summary.hsmm\_pred, 12
  - summary.hsmm\_pred\_list, 13
- \* **print**
  - summary.hsmm\_pred, 12
  - summary.hsmm\_pred\_list, 13

aaaggregation, 2, 7

add\_k\_mer\_state, 3

benchmark\_dat, 4

connection, 9, 10

duration\_viterbi, 4

find\_nhc, 5, 13

gui\_signalHsmm, 6

hsmm\_pred, 6, 7, 8, 11, 12

hsmm\_pred\_list, 7, 8, 11, 13

is\_protein, 7

plot.hsmm\_pred, 7, 8

pred2df, 7, 8

predict.sighsmm\_model, 7, 9

read\_txt, 9

read\_uniprot, 10

run\_signalHsmm, 6, 7, 10

SeqFastaAA, 4, 5, 10

signalHsmm, 11

signalHsmm-package (signalHsmm), 11

summary.hsmm\_pred, 7, 12

summary.hsmm\_pred\_list, 7, 13

train\_hsmm, 7, 13