

# binspec

May 18, 2016

---

<code>ann_classify</code>	<i>Artificial Neural Network</i>
---------------------------	----------------------------------

---

## Description

Build a ANN classifier and test it.

Build a SVM classifier and test it.

## Usage

```
ann_classify(training_set, training_labels, test_set, test_labels)
```

```
ann_classify(training_set, training_labels, test_set, test_labels)
```

---

<code>binary_peaks</code>	<i>Find binary peaks</i>
---------------------------	--------------------------

---

## Description

Find peaks in window of size  $2 \times \text{neighbors} + 1$  and label m/z integers within the error as peaks.  
Returns vector of peak m/z integers.

## Usage

```
binary_peaks(df, neighbors, error = 0)
```

## Arguments

<code>df</code>	Data frame of m/z and intensities
<code>neighbors</code>	Number of neighboring m/z values to compare on right and left
<code>error</code>	m/z Decimal error value

---

combine_peaks	<i>Combine peak vectors</i>
---------------	-----------------------------

---

**Description**

Create a binary matrix, each column represents an m/z value, and each row represents a mass spectra. The value indicates whether or not the m/z of this spectra is a peak.

**Usage**

```
combine_peaks(list_mz_peaks)
```

**Arguments**

list_mz_peaks	List of m/z peak vectors
---------------	--------------------------

---

random_forest_classify	<i>Random Forest</i>
------------------------	----------------------

---

**Description**

Build a random forest classifier and test it. No need for test set because out-of-bag error measurement.

**Usage**

```
random_forest_classify(training_set, training_labels)
```

---

round_df	<i>Round data frame</i>
----------	-------------------------

---

**Description**

Round all m/z and intensity values to integers.

**Usage**

```
round_df(df)
```

**Arguments**

df	Data frame
----	------------

# Index

`ann_classify`, [1](#)  
`binary_peaks`, [1](#)  
`combine_peaks`, [2](#)  
`random_forest_classify`, [2](#)  
`round_df`, [2](#)