

binspec

June 17, 2016

<code>accuracy_matrix</code>	<i>Generates matrix of accuracy values</i>
------------------------------	--

Description

Returns a matrix of accuracy values where the columns represent the classifier type, and the rows represent the `min_peak_count.neighbor` values used for the classifier

Usage

```
accuracy_matrix(rf, svm)
```

Arguments

<code>rf</code>	A random forest object
<code>svm</code>	A support vector machine object

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

`classifier_accuracies`*Classifier Accuracies*

Description

Find the best classifier using leave-one-out cross validation (svm) and out-of-bag error (random forests). Returns a list of classifier results

Usage

```
classifier_accuracies(peaks, labels, training, min_peak_percentage)
```

Arguments

<code>peaks</code>	Boolean matrix of mass spectra rows with m/z columns, indicating if an m/z value corresponds to a peak.
<code>labels</code>	The correct classifications of the peaks.
<code>training</code>	The rows to actually use to train
<code>minpeaks</code>	How many "true" values must show up for a given m/z value for it to be considered a feature.

`combine_peaks`*Combine peak vectors*

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

<code>list_mz_peaks</code>	List of m/z peak vectors
----------------------------	--------------------------

```
naive_feature_importance
```

Rank importance of features

Description

Given a matrix of binary peaks and each row's corresponding labels, this function takes returns the absolute difference between the proportion of times an m/z value was labeled as a peak within each of the two classes.

Usage

```
naive_feature_importance(peaks, labels)
```

Arguments

peaks	A matrix of peaks
labels	A factor vector of labels whose length is equal to the number of rows of peaks

```
peak_frequencies
```

Get frequency of peaks

Description

Return frequency of peaks within each label

Usage

```
peak_frequencies(peaks, labels)
```

```
plot_naive_importance
```

Plot importance of naive importance vector

Description

Plot importance of each m/z value according to the naive ranking vector. Returns a ggplot object

Usage

```
plot_naive_importance(naive_importance, count = 10)
```

Arguments

naive_importance	A vector returned from naive_feature_importance()
------------------	---

`plot_rf_differences`*Plot important peaks on each day*

Description

Returns a ggplot object of the most important peak frequencies

Usage

```
plot_rf_differences(peak_freqs, rf, count)
```

Arguments

<code>rf</code>	A random forest object, to be used for feature importance
<code>count</code>	Number of peaks to select
<code>peaks_freqs</code>	The frequency of each peak within each label

`plot_rf_importance` *Plot importance of randomForest*

Description

Plot importance of each m/z value according to a randomForest object. Returns a ggplot object

Usage

```
plot_rf_importance(rf, count)
```

Arguments

<code>rf</code>	A randomForest object
-----------------	-----------------------

`round_df` *Round data frame*

Description

Round all m/z and intensity values to integers.

Usage

```
round_df(df)
```

Arguments

<code>df</code>	Data frame
-----------------	------------

svm_rf*SVM and RF Accuracies*

Description

Given a vector of neighbor values and a vector of the minimum number of peaks to be considered, this function finds the peak m/z values for a data set by running `binary_peaks` using each of the neighbor vector values, runs SVM and RF on the peaks for each of the `min_peak_count` values, and returns the accuracies of each test in a table. The table's rows are the number of neighbors, and the columns are the `min_peak_count` values.

Usage

```
svm_rf(list_of_dfs, labels, neighbors, min_peaks_percentage,  
       error_window = 0.005, training = 1:length(list_of_dfs),  
       multiple_cores = 1)
```

Arguments

<code>list_of_dfs</code>	The first data frame of m/z values and frequencies
<code>labels</code>	The labels of the two states the first data frame's values could be classified as
<code>neighbors</code>	A vector of the number of neighbors to be considered in the <code>binary_peaks</code> function
<code>min_peaks_percentage</code>	A vector of the minimum percent of times an m/z must be a peak to be considered in the <code>classifier_accuracies</code> function
<code>training</code>	Vector of the data frames to be used for training
<code>multiple_cores</code>	Number of cores to use
<code>errow_window</code>	A vector of percentage of nearby peaks that should be also labeled as peaks when one is found

Index

accuracy_matrix, [1](#)
binary_peaks, [1](#)
classifier_accuracies, [2](#)
combine_peaks, [2](#)
naive_feature_importance, [3](#)
peak_frequencies, [3](#)
plot_naive_importance, [3](#)
plot_rf_differences, [4](#)
plot_rf_importance, [4](#)
round_df, [4](#)
svm_rf, [5](#)