Package iRRAPy

import

It is possible to use all the function described in document by importing the package in the following way:

```
from iRRAPy import *
```

It is of course possible to import all the function separately. The user could also import other utility functions this way, but these are not meant to be used directy by the user.

External libraries.

This package uses the following external library and function:

`Matplotlib.pyplot`

`Matplotlib.colors`

`Scipy.integrate.quad()`

Their copyright and licences are available in the file `licenze.txt,` or on their website.

Display the Region of Interest and compute the Ratio of Relevant Areas

This package allows the user to plot a ROC curve and a Region of Interest (RoI), highlighting the space under the curve, and to compute the Ratio of Relevant Areas (RRA). It is also possible to confront different ROC curves' RRAs and average them out.

Functions:

- rra
- RRAresult
  - plot
  - print
  - getRRA
- rra_average
- rra_test

RRA

This is the main function of this package. It builds the Region of Interest (RoI) given the coordinates of the ROC curve, the number of Actual Positives (AP) and Actual Negatives (AN) and the reference values of the performance metrics.

It returns a `RRAresult` object. It is also possible to print and plot the curve and the RoI directly by this function.

DEFINITION:

```
rra(roc_x, roc_y, AP, AN,
        *,precision = False, c_precision = "pop", p_precision = -1,
        recall = False, c_recall = "pop", p_recall = -1,
        fm = False, c_fm = "pop", p_fm = -1,
        npv = False, c_npv = "pop", p_npv = -1,
        specificity = False, c_specificity = "pop", p_specificity = -1,
        fallout = False, c_fallout = "pop", p_fallout = -1,
        nm = False, c_nm = "pop", p_nm = -1,
        j = False, c_j = -1,
        markedness = False, c_markedness = -1,
        phi = False, c_phi = 0.4,
        ncost = False, c_ncost = "uses_mu", lamb = [-1], mu = 1,
        plot = True, print = True,
        cROI = "grey", cUnder = "c", col="k",
        xlab = "FP/AN", ylab = "TP/AP", main = "ROC curve and RoI",
        xlim = [0,1], ylim = [0,1], asp = 'auto',
        ls = "-", lw = 1, ADD = False, diagonal = True, show = True,
         label = "ROC curve")
```

ARGUMENTS

`roc_x, roc_y`	Two list representing the coordinates of the ROC curve's points
`AP`	The number of actual positives. This value represents the number of positive responses (the "1" values) used to build the ROC curve. It must be greater than 0
`AN`	The number of actual negatives. This value represents the number of negative responses (the "0" values) used to build the ROC curve. It must be greater than or equal to 0
`precision, recall, fm, npv, specificity, fallout, nm, j, phi, markedness, ncost`	Boolean value indicating which metrics the user wants to use as reference values for the generation of the RoI
`c_precision, c_recall, c_fm, c_npv, c_specificity, c_fallout, c_nm`	The reference values. It must be between 0and 1. It can also indicate that the reference value is selected using a uniform random policy ("uni") method (c_... = "uni"). If c_...= "pop" it means that a "uni" method with p(m)=AP/nis used ("Proportion Of Positives" policy)
`p_precision, p_recall, p_fm, p_npv, p_specificity, p_fallout, p_nm`	The unified probabilities used by the "uni" method
`c_j`	The reference valuefor Youden's J. It must be between -1and 1
`c_markedness, c_phi`	The reference values for Phi and Markedness. It must be between 0 and 1
`C_ncost`	The reference valuefor the Normalized Cost. It must be between 0and the NC value selected with the "pop"method. This value will define the cost reduction index (mu). The default value, "uses_mu",

	indicates that the user wants to use the mu value directly
`lamb`	The value of False Negative and False Positive cost ratio as (cFN / (cFN+cFP)). It is a list of one or two elements. They must be between 0 and 1. Indicating two values means that the user wants to consider a range of lambda values
`mu`	The reduction cost index value. It's used only if c_ncost = "uses_mu"
`plot`	If the user wants to plot the RRAresult object. Check the `plot` method for `RRAresult`
`print`	If the user wants to print the RRAresult object. Check the `print` method for `RRAresult`
`cROI, cUnder, col, xlab, ylab, main, xlim, ylim, asp, ls, lw, ADD, diagonal, show, label`	These are arguments used by the `plot` method when `plot = True`. Check the `plot` method for more information

DETAILS

The Region of Interest (RoI) represents the points in the ROC space that have a better performance value than the reference values.

Every performance metrics corresponds to a specific border of the RoI. It is possible to use multiple metrics and different methods, but it is important to keep in mind that some border could be always greater than others within the ROC space. In this case some borders will obscure the others.

Additionally, some special values will be not very significant. For instance, a recall reference value of 1will result in a non-existent RoI. Its RRA will be 0unless the ROC curve is perfect (AUC = 1). On the other hand, if recall is equal to 0 the RoI will correspond to the ROC space, therefore the RRA will be equal to the AUC of the ROC curve.

The default value for phi, 0.4,represents a medium-strong association between a model and actual positiveness.

ERRORS

The function will stop if roc_x and roc_y has different length or have values greater than 1or lesser than 0. It will also stop if the other parameters have invalid values.

NOTE

Note that the precision("uni") border will be `y=x` for every p(m). For this reason, using `c_precision = "uni", p_precision= (0<p<1)` will generate the same border as using `c_precision = "pop"`. This is also true for the NPV border.

RRAresult

The RRAresult class allows the user to save the returned value of the `rra` function, plot them and print the result.

ATTRIBUTES

`self.roc_x`	A list. The x values of the ROC curve's points
`self.roc_y`	A list. The y values of the ROC curve's points
`Self.roi_x`	A list. The x values of the RoI's points
`self.roi_y`	A list. The y values of the RoI's points
`self.roi_under_x`	A list of lists. Each list contains the x values of a portion of the RoI under the curve.
`self.roi_under_y`	A list of lists. Each list contains the y values of a portion of the RoI under the curve.
`self.area_roi`	Value of the area of the Region of Interest
`self.area_under`	Value of the area of the RoI under the curve
`self.metrics`	A list of character. The names of the reference metrics
`self.metrics_values`	A list of character. The values of the performance metrics

SELF.PLOT

It will plot the RRAresult object. It will draw the ROC curve and the RoI, highlighting the part under the curve. It uses the `matplotlib.pyplot` library. Checking its documentation is advised.

DEFINITION

```
plot(self, cROI = "grey", cUnder = "c", col="k",
            xlab = "FP/AN", ylab = "TP/AP", main = "ROC curve and RoI",
            xlim = [0,1], ylim = [0,1], asp = 'auto',
            ls = "-", lw = 1, ADD = False, diagonal = True, show = True,
            label = "ROC curve")
```

ARGUMENTS

`cROI`	The color of the RoI space. By default, it's grey.
`cUnder`	The color of the RoI space under the curve. By default, it's cyan.
`col`	The color of the ROC curve's line. By default, it's black
`xlab`	The name of the x-axis. The x-axis represents the FP/AN values of the ROC curve
`ylab`	The name of the y-axis. The y-axis represents the TP/AP values of the ROC curve
`main`	The title of the plot

`xlim`	The limits of the x-axis. By default, it's [0,1]. A larger limit would be senseless for a ROC curve
`ylim`	The limits of the y-axis. By default, it's [0,1]. A larger limit would be senseless for a ROC curve
`asp`	The aspect ratio of the axes.
`ls`	The line's style of the ROC curve
`lw`	The line's width of the ROC curve
`ADD`	If the user wants to add this curve to the existing plot
`diagonal`	If the user wants to draw the `y=x` line
`show`	If the user wants to directly call the `matplotlib.pyplot.show` function
`label`	The name of the curve.

DETAILS

As mentioned before, this function calls some `matplotlib.pyplot` functions, such as `plot` and `fill`. The arguments `cROI` and `cUnder` are used as color arguments for the `fill` function, while col is the color arguments.

`ls` e `lw` are arguments for the `plot` function, as well as `xlab`, `ylab`, `main`, `xlim`, `ylim` and `asp`. The last ones are only used when `ADD = False`.

IMPORTANT

`ADD = False` will call the `matplotlib.pyplot.close` function, deleting what was drawn before. Be sure to use it carefully.

NOTE

Note that the RoI is plotted as a polygon. It could hide other plotted structures if `ADD= TRUE`.

SELF.PRINT

It will print the information about the reference values generating the RoI's border and the resulting Ratio of Relevant Areas (RRA)

DEFINITION

```
print(self)
```

DETAILS

This function will print the meaning of the Region of Interest selected; it will highlight what performance values are matched by all the points within the RoI space.

If the object represents the AUC of the ROC curve it will just print the resulting Area Under the Curve.

SELF.GETRRA

It returns the RRA value as `self.area_under / self.area_roi`

DEFINITION

```
getRRA(self)
```

RRA_AVERAGE

This function averages the RRA values of multiple `RRAresult` objects. It will return an Average object.

DEFINITION

```
rra_average(results_list, prin = True)
```

ARGUMENTS

`Results_list`	A list of `RRAresult` objects
`Prin`	If the user wants to print the resulting Average object

DETAILS

The Average object has three attributes: `self.average` (The average value), `self.std_deviation` (the Standard deviation) and `self.median` (the median value(s)).

If `prin = True` it will print the Average object

RRA_TEST

This function tests a list of ROC curves and check which one has the best RRA value. It can get as input either a list of RRAresult objects or two lists representing the ROC curves' points based on the same data set (AN and AP are the same for all the curves). It uses the `matplotlib.pyplot` library. Checking its documentation is advised.

DEFINITION

```
rra_test(list1, list2 = [], AP=0, AN=0, plot = True, ADD = False,
            show = True, col_best = "k", c_under_best = "c",
            croi_best="grey", lwd_best = 2, ls_best = "-",
            lwd_other = 1, ls_other = "--",
```

```
add_under_others = False, legend = False, **args)
```

ARGUMENTS

`list1`	A list of RRAresult objects or a list of lists of x values of the ROC curves
`list2`	A list of lists of y values of the ROC curves.
`AP`	The number of actual positives. This value represents the number of positive responses (the "1" values) used to build the ROC curve. It must be greater than 0
`AN`	The number of actual negatives. This value represents the number of negative responses (the "0" values) used to build the ROC curve. It must be greater than or equal to 0
`plot`	If the user wants to plot the confrontation
`ADD`	If the user wants to add the confrontation to the already existing plot
`show`	If the user wants to directly call the `matplotlib.pyplot.show` function
`col_best`	The color of the ROC curve with the highest RRA value
`C_under_best`	The color of the RoI space under the curve of the ROC curve with the highest RRA value
`croi_best`	The color of the RoI space of the ROC curve with the highest RRA value
`lwd_best`	The width of the ROC curve with the highest RRA value
`ls_best`	The style of the ROC curve with the highest RRA value
`lwd_other`	The width of the other ROC curves
`ls_other`	The style of the other ROC curves
`add_under_others`	If the user wants to draw the RoI under the curve of the other ROC curves
`legend`	If the user wants to add a legend
`**args`	Other arguments for the `rra` function and the `plot` function

DETAILS

This function will also return a list of the indices of the ROC curves ordered by the best RRA value to the worse.

It's possible to test any `RRAresult` object, but when two curves are generated by different data sets (AN and AP values have different values for the two curves), or when the RoI is computed by different performance metrics, the user will be warned.

The color of the other curves in is randomly selected by a list of colors.

By default, the RoI under the curve of the other curves will not be drawn to make the plot easier to understand. Consider that, if the `RRAresult` object considered have different RoI and RoI under the curve (diferent data-set or different reference metrics), one may want to set `AddcUnderOthers=TRUE`

All the arguments regarding the style or the color of the curves and the polygons work exactly like the arguments of the `RRAresult.plot` function. Check its documentation for more details.

`**args` allows the user to specify all the reference metrics values he needs when he confronts multiple ROC curves. Additionally, he can specify all the arguments from `RRAresult.plot` he needs to build the axis and set the labels, such as `main, xlim` and `ylim.` These arguments can be also set when testing multiple `RRAresult` objects.

IMPORTANT

`ADD = False` will call the `matplotlib.pyplot.close` function, deleting what was drawn before. Be sure to use it carefully.