Supplementary material for VISROC 2.0: Updated Software for the Visualization of the significance of Receiver Operating Characteristics based on confidence ellipses

1. The Receiver Operating Characteristics method

The Receiver Operating Characteristics (ROC) curve is a technique used[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12] to estimate the predictability of various complex systems. Briefly, the concept behind the method is as follows[13]:

In the case of binary predictions, there are two classes of events: Positive **p** or Negative **n**, where an event is considered as **p** if its magnitude exceeds a given threshold M_t, otherwise it is classified as **n**. Before the occurrence of each event, a predictor ϵ is assigned and a class **P** or **N** for the impending event is decided based on whether ϵ is larger or smaller than, respectively, the predictor threshold value ϵ_t. If this hypothesized class is **P** and the event is **p** we have a successful prediction called True Positive (TP). If the hypothesized class is **N** and the event is **n** we also have a successful prediction, this time called True Negative (TN). However, if the class is **P** and the event is **n** we have a False Positive (FP) unsuccessful prediction, while if the class is **N** and the event is **p** we have a False Negative (FN) unsuccessful prediction.

Assuming that we examine P events belonging to the **p** class and Q events belonging to the **n** class, one defines[13] the hit rate (H) as:

$$H \equiv \frac{|TP|}{P} = \frac{|TP|}{|TP| + |FN|}, \tag{1}$$

and the false alarm rate (F) as:

$$F \equiv \frac{|FP|}{Q} = \frac{|FP|}{|FP| + |TN|}. \tag{2}$$

The ROC curve is then obtained by plotting H as a function of F while we vary the predictor threshold value, ϵ_t (see, for example, Fig. 1 of Ref. [14]).

The statistical significance of a ROC curve will depend on its deviation from the diagonal where $H = F$ which corresponds on average to totally random predictions.

When P and Q are sufficiently large[15], we are led, for random predictions, to a two dimensional Gaussian distribution for H and F[14]:

$$f(F, H) = \frac{\sqrt{PQ}}{2\pi\epsilon_t(1-\epsilon_t)} \exp\left[-\frac{Q(F-\epsilon_t)^2 + P(H-\epsilon_t)^2}{2\epsilon_t(1-\epsilon_t)}\right]. \quad (3)$$

The confidence regions in this case[16] are the confidence ellipses[14]:

$$Q(F-\epsilon_t)^2 + P(H-\epsilon_t)^2 = k\epsilon_t(1-\epsilon_t), \quad (4)$$

with center $(F, H) = (\epsilon_t, \epsilon_t)$ on the diagonal of the ROC space. k is a positive parameter signifying the confidence level with $p_0 = \exp(-k/2)$. Varying ϵ_t within the region $[0, 1]$, the center of the ellipse moves along the diagonal. Thus, we can obtain different ellipses for a given p_0-value. The envelope of these ellipses can be determined by maximizing H under the condition (4) for given values of F and k. It turns out that this envelope is also an ellipse. The family of these envelope ellipses is called[14] k-ellipses family and it covers the whole ROC space. The k-value corresponding to an arbitrary ROC point (F, H) is given by[14]:

$$\begin{aligned} k(F,H) &= 2P(H^2-H) + 2Q(F^2-F) \\ &+ 2\sqrt{[P(H^2-H) + Q(F^2-F)]^2 + PQ(F-H)^2}. \end{aligned} \quad (5)$$

Mason and Graham[17] have shown that the Area Under the Curve (AUC), labelled as A, in the ROC plane can be calculated as:

$$A = 1 - \frac{U}{PQ}, \quad (6)$$

where U follows the Mann-Whitney U-statistics[18]. This means that U equals the sum of the number of cases a number $u_k, k = 1, 2, \ldots, P$ is larger than a number $u'_l, l = 1, 2, \ldots, Q$ when both $\{u_k\}$ and $\{u_l\}$ originate from the same continuous distribution. For large values of P and Q, i.e., when P or Q are greater or equal to 30 and $P + Q \geq 40$, the U-statistics can be approximated[17] by a Gaussian distribution.

By calculating the AUC corresponding to a specific k-ellipse, we can estimate its significance level p. It has been shown[14] that the AUC can be calculated using the following equation:

$$A(k) = \left(1 - \frac{x_1}{2}\right) + \left(\frac{Q}{Q+k}\right)\frac{x_1}{2}(x_1 - 1)$$

$$+\frac{1}{2}\left(\frac{1}{Q+k}\right)\sqrt{\frac{k(Q+k+P)}{P}}\left\{\sqrt{Q}\left[\left(x_1 - \frac{1}{2}\right)\sqrt{\frac{Q+k}{4Q} - \left(x_1 - \frac{1}{2}\right)^2}\right.\right.$$

$$\left. + \left(\frac{Q+k}{4Q}\right)\arcsin\left(2\left(x_1 - \frac{1}{2}\right)\sqrt{\frac{Q}{Q+k}}\right)\right]$$

$$\left. + \frac{1}{4\sqrt{Q}}\left[\sqrt{kQ} + (Q+k)\arcsin\left(\sqrt{\frac{Q}{Q+k}}\right)\right]\right\} \tag{7}$$

where

$$x_1 = \frac{1}{2} + \frac{PQ - k\sqrt{Q(k+Q+P)}}{2Q(k+P)} \tag{8}$$

The previous version[14] published in 2014 provided a *FORTRAN* code called VISROC in order to estimate the statistical significance p in the ROC plane using either Mann-Whitney U-statistics or its Gaussian approximation, if applicable. To do so, for each point (F, H), the authors estimated the k value (Eq. 5) of the k-ellipse that passes through this point. Then, they calculated the AUC $A(k)$ using Eq.(7), as well as three k-ellipses with p-values equal to 10%, 5% and 1%, using Eq. (11) of Ref. [14]. The statistical significance p of a user defined area under the ROC curve could also be estimated by the code.

In this paper, we present the updated code for the visualization of the confidence intervals in the ROC plane. The new code has two versions, in *R* and in *Python*, and implements a GUI that was not available before. We will discuss each version's characteristics as well as the steps taken to overcome the *FORTRAN* version's limitations in the subsequent Section.

2. VISROC Updated Version

The most important new feature in VISROC 2.0 is the addition of a GUI, rendering the software more user-friendly by allowing the user to perform ROC analysis without the knowledge of a specific programming language. Additionally, a user defined ROC can be inserted through a file containing (F, H) points, a feature that was not available before.

Furthermore, the original *FORTRAN* code VISROC had an unusual feature during the Gaussian approximation in the extreme case that either P or Q was unity (and the other was greater than 38). In such cases, the final column of the output file outCL.dat (see Subsection 2.1) appeared as NaN, reflecting the fact that the k-ellipse with p-value $p = 1\%$ could not be drawn since all the points of the ROC unit square corresponded to k-ellipses with larger p-values. To surpass this, we introduce in VISROC 2.0 an upper limit for the AUC of $p = 1\%$, where if the calculated AUC is larger than unity, then it will default to the number $1 - 10^{-15} = 0.999999999999999$. Obviously when this upper limit is invoked the resulting k-ellipse is very close to the upper horizontal axis of the ROC diagram and hence indiscernible, causing no confusion.

The code for VISROC 2.0 was written in two programming languages: *R* and *Python*. In Subsections 2.1 and 2.2, we will describe how each version works in more detail.

2.1. R Version

The *R* version of VISROC 2.0 can either be run locally using the *R* language environment installed in the user's computer, or be launched as an online application, opening in a browser.

The software tests for package dependencies that need to be installed before running the application and installs them automatically in the required version.

The GUI is shown in Fig. 1. One can see that the user can input the number of positive events P, the number of negative events Q, and the number N of segments in which the interval $[0, 1]$ is divided on a square lattice in the ROC diagram by using the *resolution* field. Furthermore, the value of a user defined AUC for which they would like to find the p-value can also be defined, as well as the values F_1 and H_1 ($H_1 > F_1$) of a user defined point (F_1, H_1) on the ROC diagram, for which the user would like to find the p-value of the k-ellipse passing through it. In addition, by using the button "Browse", the user can insert a file containing (F, H) points.

The main window in Fig. 1 showcases a general graph interpretation, with an added button that helps the user refer back to it after running an example. Finally, a "Help" button offers instructions regarding the program's input parameters, as well as the output files and results, the different menus, and the potential error outputs.

In Figure 2, we showcase an example by inserting a file called example.csv containing some test data. The input files we use are $P = 15$, $Q = 35$, $N = 100$, User Defined AUC= 0.51, and $(F_1, H_1) = (0.65, 0.75)$. The resulted p-value for the User Defined AUC is approximately 0.46. For the $(0.65, 0.75)$ point the p-value is approximately 0.17 with an AUC approximately 0.58. Finally, for the example.csv input file the p-value is approximately 0.01 with AUC=0.70. The "No fault" indication signifies that we had correct execution, with reasonable input parameters.

In addition, a Plot(F1, H1) button shows the plot for the $(F_1, H_1) = (0.65, 0.75)$ point and the corresponding k-ellipse. Finally, the program allows to change the ROC diagram's axis labels and use either Hit Rate vs. False Alarm Rate, Sensitivity vs. 1-Specificity, or True Positive Rate vs. False Positive Rate.

The results after the execution of the program can be saved in various files, apart from briefly summarized on the workspace (see Fig. 2). In particular, the saved files are:

i) ROC_plot.pdf: It contains the ROC graph.

ii) F1H1_plot.pdf: It contains the ROC graph for the particular point (F_1, H_1).

iii) out_F1H1.csv: One line text file, comma separated, containing the p-value of the k-ellipse passing through the point (F_1, H_1), the values F_1 and H_1, the value of "ifault", which is an error code that should be zero for correct execution and becomes equal to one if some input parameter is unreasonable, the number of positive and negative events P and Q, and the AUC of the k-ellipse.

iv) out_k_F1H1.csv: The coordinates of the k-ellipse that passes through the point (F_1, H_1) in the first two columns together with the corresponding k value in the third column. These coordiantes can be outside the $[0, 1]$ interval. For calculating the AUC and the p-value, we truncate this set of coordinates.

v) outCL.csv: 4-column text file, comma separated, containing the F coordinate of the points of the k-ellipses with p-values equal to 10%, 5% and 1% in the first column, and their H coordinates in the second, third and fourth column, respectively.

vi) outfield.csv: 3-column text file, comma separated, containing the coordinates of the lattice points in the ROC diagram for which calculation has been made together with the corresponding p-values.

vii) out_p.csv: One line data file containing the p-value of the user defined AUC, the input value of the user defined AUC, the number of positive and negative events P and Q, and the value of the “ifault” error code.

The aforementioned files are saved in a single compressed file titled output.zip.

2.2. Python Version

The VISROC 2.0 written in *Python* can be run on windows (.exe file), linux (.elf file) or macOS (.app) file. The GUI is shown in Fig. 3 and it is essentially the same with the *R* version regarding the fields for the input values and the menus.

In Figure 4, we showcase the same example as in Subsection 2.1. The results regarding the p-values and AUCs are identical with those produced after executing the *R* version. However, the GUI in this case has a series of control buttons that help the user optimise the size of the ROC graph (see Fig. 4). A clickable reference to the publication of the original VISROC program is also added.

Additionally, using the *Python* version of VISROC 2.0 the user has the option to see the p-value of any point on the ROC graph by simply passing the cursor over it. An example of this is shown in Fig. 5.

Finally, the software produces the same output files as in Subsection 2.1. However, in this case, the user has the additional option to save the individual files separately apart from the single compressed file in the *R* case. In addition, the default format for the ROC_plot and F1H1_plot is .png files instead of .pdf, while there are additional options for .eps, .jpeg, .pdf, .pgf, .ps, .raw, .svg, .tif file formats. We also note that there are some known issues regarding the execution of VISROC 2.0 which, however, do not affect the operation of the tool. The solution to these issues is described in detail in the README.md file.

3. Summary

Here, we presented an updated version of the program VISROC for the visualization of the confidence intervals in the ROC plane with a Graphical User Interface. Code in *R* and *Python* executables for windows, linux, and macOS operating systems are available. These provide an estimate of the statistical significance p for each point on the ROC plane based on the family of k-ellipses. The k-ellipses with p-values 10%, 5% and 1% which also may be of interest for researchers using ROC curves in various fields are also provided. The statistical significance p of a user defined area under the ROC curve can be estimated as well as the p-value for a user defined ROC.

References

[1] D. Hosmer, S. Lemeshow, Applied logistic regression. John Wiley & Sons, Ltd, New York, NY (2000).

[2] A. Garber, S. Hallerberg, H. Kantz, Predicting extreme avalanches in self-organized critical sandpiles, Phys. Rev. E 80 (2009) 026124.

[3] N. V. Sarlis, E. S. Skordas, P. A. Varotsos, Order parameter fluctuations of seismicity in natural time before and after mainshocks, EPL 91 (2010) 59001.

[4] P. A. Varotsos, N. V. Sarlis, E. S. Skordas, Natural Time Analysis: The new view of time. Precursory Seismic Electric Signals, Earthquakes and other Complex Time-Series, Springer-Verlag, Berlin Heidelberg, 2011.

[5] F. Caruso, H. Kantz, Prediction of extreme events in the OFC model on a small world network, Eur. Phys. J. B 79 (2011) 7.

[6] N. Sarlis, E. Skordas, P. Varotsos, The change of the entropy in natural time under time-reversal in the Olami-Feder-Christensen earthquake model, Tectonophysics 513 (2011) 49 – 53.

[7] N. V. Sarlis, S.-R. G. Christopoulos, Predictability of the coherent-noise model and its applications, Phys. Rev. E 85 (2012) 051136.

[8] P. A. Varotsos, N. V. Sarlis, E. S. Skordas, Order parameter fluctuations in natural time and b-value variation before large earthquakes, Natural Hazards and Earth System Science 12 (2012) 3473–3481.

[9] N. Sarlis, S.-R. Christopoulos, E. Skordas, Minima of the fluctuations of the order parameter of global seismicity, Chaos: An Interdisciplinary Journal of Nonlinear Science 25 (6) (2015) 063110.

[10] K. Feng, H. Hong, K. Tang, J. Wang, Properties of ROC curves (February 25, 2022), Available at SSRN: https://ssrn.com/abstract=3382962 or http://dx.doi.org/10.2139/ssrn.3382962 (2022).

[11] N. V. Sarlis, E. S. Skordas, S.-R. G. Christopoulos, P. A. Varotsos, Natural time analysis: The area under the receiver operating characteristic curve of the order parameter fluctuations minima preceding major earthquakes, Entropy 22 (5) (2020) 583.

[12] E. Raffinetti, P. Giudici, A generalised ROC curve (July 9, 2021), Available at SSRN: https://ssrn.com/abstract=3883422 or http://dx.doi.org/10.2139/ssrn.3883422 (2021).

[13] T. Fawcett, An introduction to ROC analysis, Pattern Recogn. Lett. 27 (8) (2006) 861–874.

[14] N. V. Sarlis, S.-R. G. Christopoulos, Visualization of the significance of receiver operating characteristics based on confidence ellipses, Computer Physics Communications 185 (3) (2014) 1172–1176.

[15] W. Feller, An Introduction to Probability Theory and Its Applications, Vol. I, Wiley, New York, 1968.

[16] J. Beringer, *et al.*, Particle Data Group, Review of particle physics, Phys. Rev. D 86 (2012) 010001.

[17] S. J. Mason, N. E. Graham, Areas beneath the relative operating characteristics (ROC) and relative operating levels (ROL) curves: Statistical signififcance and interpretation, Quart. J. R. Meteor. Soc. 128 (2002) 2145–2166.

[18] H. B. Mann, D. R. Whitney, On a test of whether one of two random variables is stochastically larger than the other, Ann. Math. Statist. 18 (1947) 50–60.

Figure 1: VISROC 2.0 GUI for the R version of the software.

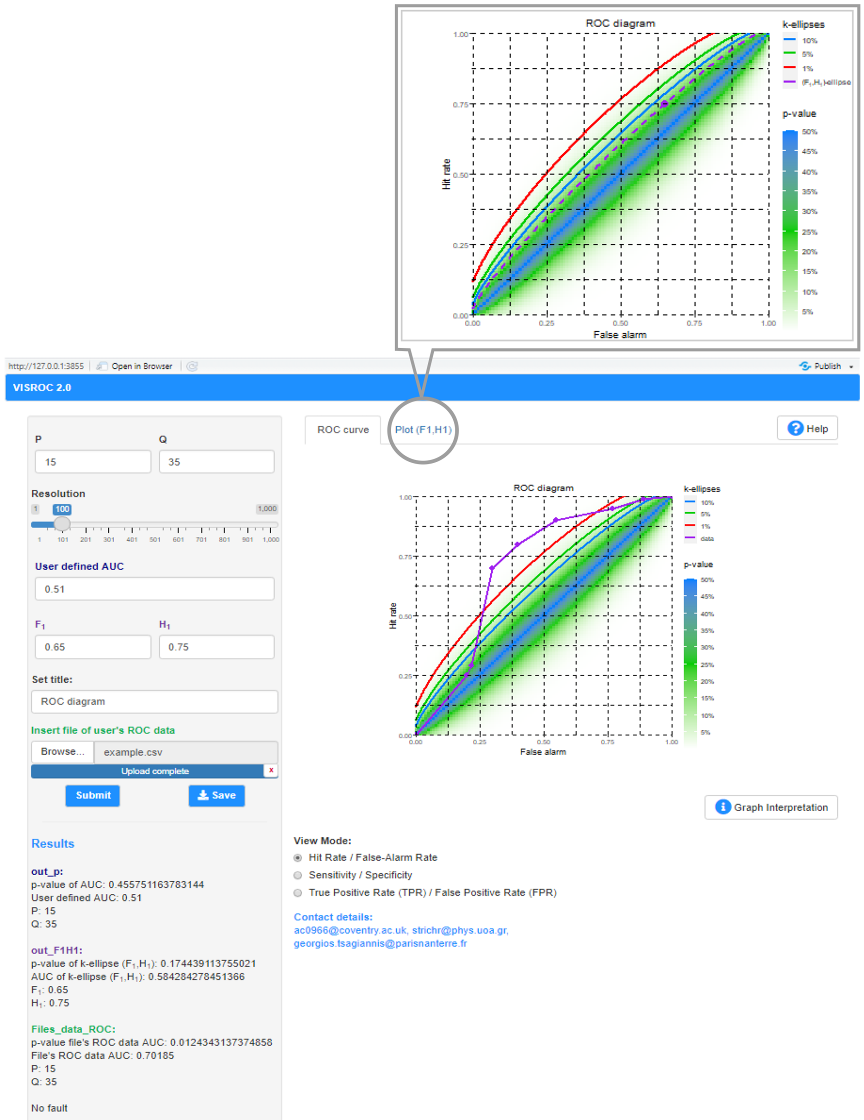

Figure 2: An example run using the *R* version of VISROC 2.0. By clicking on the circled selection, the window changes to the picture on the top.

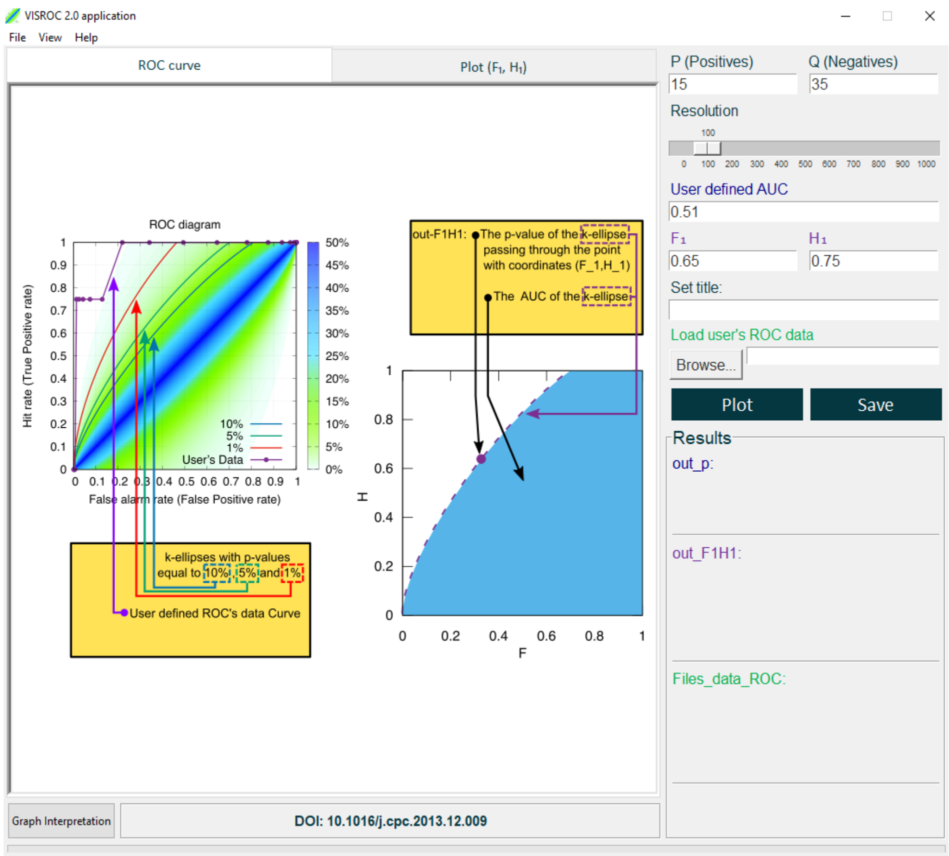

Figure 3: VISROC 2.0 GUI for the *Python* version of the software.

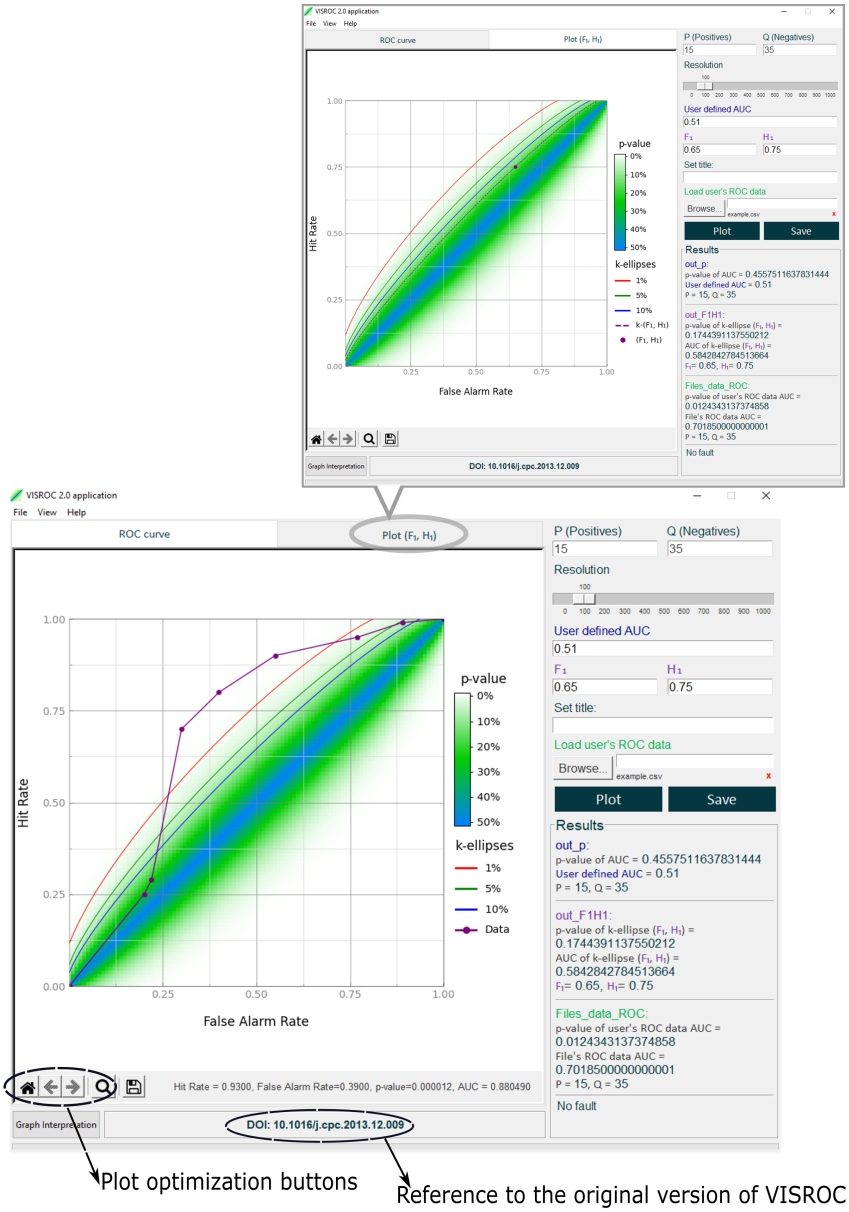

Figure 4: An example run using the *Python* version of VISROC 2.0. By clicking on the circled selection, the window changes to the picture on the top.

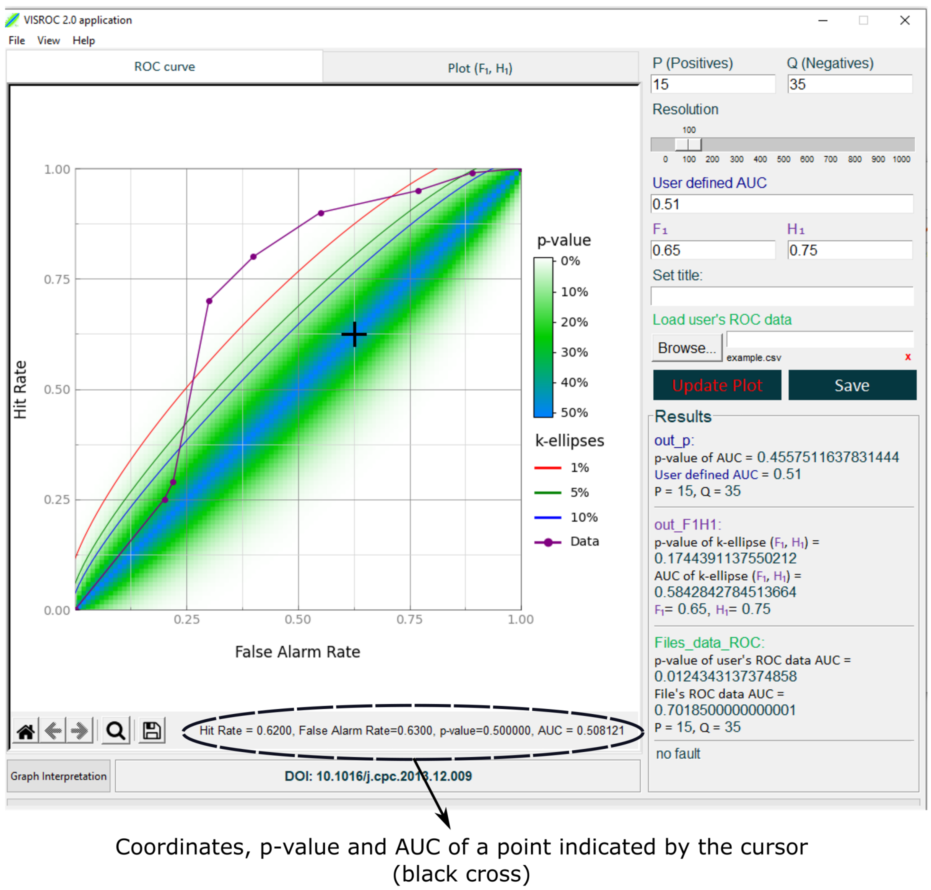

Figure 5: By passing the cursor over a specific point in the *Python* version of VISROC 2.0, the user can see its coordinates, p-value and AUC.