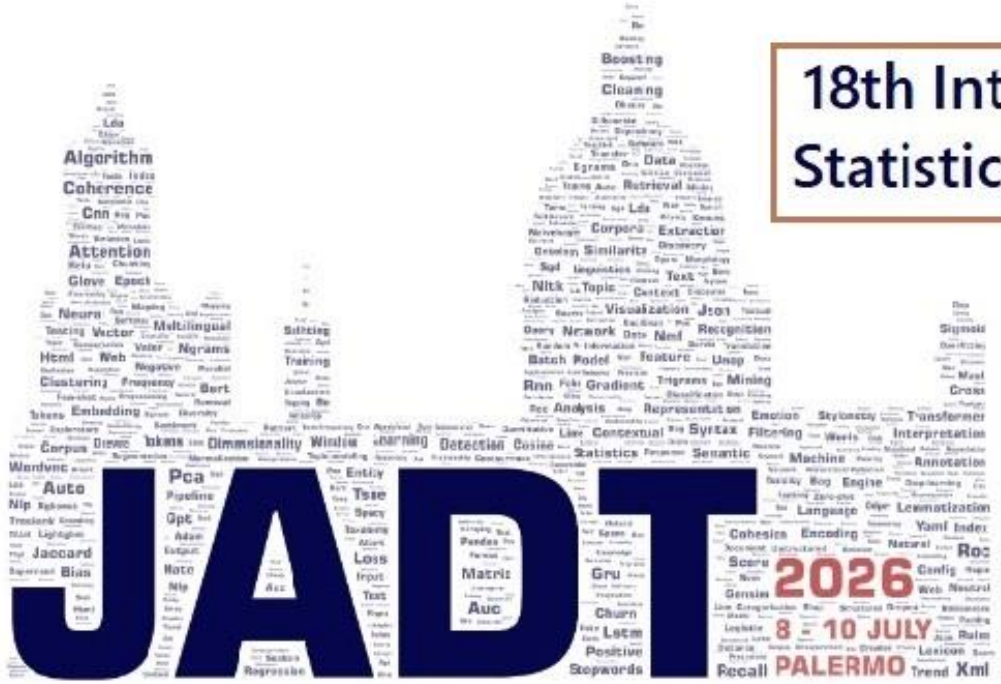


18th International Conference on Statistical Analysis of Textual Data

StatPanormus

UNIVERSITÀ DEGLI STUDI DI PALERMO

dSEAS
dipartimento di scienze economiche aziendali e statistiche
department of economics business and statistics

Validation of Simultaneous Additive Trees in Textual Data Visualization

Ludovic Lebart

CNRS (R), France
ludovic@lebart.org

Validation of Simultaneous Additive Trees in Textual Data Visualization

1. Introduction: Combining principal axes and clustering

We show that in the context of **Simultaneous Additive Trees (SAT)**, non-parametric confidence zones for point positions can be plotted.

This technique complement **Correspondence Analysis (CA)**, not replace it. **CA** remains an essential step when dealing with lexical tables but does not provide a comprehensive overview. In contrast, **AT** allow for a planar reconstruction of distances computed within spaces of high dimensions.

A method for the simultaneous representation of texts and words in **SAT** and for the positioning of supplementary variables is summarized in **Section 2**.

A validation procedure based on the **bootstrap** is presented in **Section 3**.

Section 4 then describes the corpus (**Shakespeare Sonnets**) used as an illustration, while **Section 5** presents the application itself.

Finally, beyond the conclusion (**section 6**) an annex (**section 7**) provides some pieces of information about the self-contained **Python** program chain.

Additive trees (AT): the phylogenetic solutions (*History)*

These trees were originally proposed by Buneman (1971), then studied by Sattah and Tverski (1977).

An AT tree can be drawn with the objects as nodes (vertices), such as the real distance between two objects is the length of the path joining these two objects on the tree.

Saitou and Nei (1987) proposed an algorithm called **Neighbor Joining** which allows to roughly reduce the search of the additive tree to a classical ascending classification procedure.

This heuristic had a huge impact on the rapidly expanding world of phylogenetic research. *(Up to 2026 July 1st, Saitou and Nei's article has been cited more than 77 630 times since its publication).*

Theoretical justifications for the algorithm's efficiency were presented by Mihaescu et al. (2009).

2.1 On the options for drawing additive trees (*reminder*).

For a complete review of graph visualizations, one may consult Di Battista et al. (1999), and more particularly on methods using the so-called “force-directed drawing algorithms”, the article by Kobourov (2013) which analyzes more than 60 publications corresponding to several dozen algorithms.

Originally, the graph drawing algorithm of **Tutte (1963)** is one of the first drawing methods based on algorithms of this type.

The methods proposed by Eades (1984) and the algorithm of Fruchterman and Reingold (1991) are both based on repulsive forces between all the nodes of the graph, but also attractive forces between the nodes which are adjacent (the edges are assimilated to springs, and it is about finding a balance between all the tensions, hence the name: ***force-directed drawings***).

The algorithm of Kamada and Kawai (1989) uses these “spring forces” proportional to these distances calculated on the graph.

This tracing algorithm is therefore the most compatible with the properties of additive trees, and therefore with basic lexical distances.

Simultaneous Additive Trees (*reminder*).

A pragmatic simultaneous representation procedure for texts and words has been proposed (Lebart, 2024) which makes it possible to combine:

- the **advantages of some principal axis methods such as CA:**
  (simultaneous planar representation of rows and columns of a table)

- and the **advantages of clustering techniques**
  (better approximation of the distances in **full** space).

More generally, this procedure applies to the simultaneous representation of columns and rows of any contingency table.

Sequence of computation steps for Simultaneous AT (SAT)

1) **Preliminary correspondence analysis** (of the contingency lexical table).

2) **Choice of the dimension *nx*** of the space deemed significant (generally through bootstrap validation) (12 axes for example). The distances will be calculated from the first *nx* main axes of the CA.

3) **Computation of the additive tree** (Neighbors-Joining method) on the matrix of distances between the coordinates of the columns (texts) on the first *nx* axes.

4) **Drawing of the tree** (Kamada-Kawai procedure).

5) **Barycentric positioning of the rows** (words - forms, lemmas) from the coordinates of the column points (texts) (vertices of the tree) deduced from the procedure (4) and the textual profile of the rows (words/tokens/lemmas)).

6) **Computation**, directly from the lexical table, for each text column, **of the characteristic rows/words** (fixed probabilistic threshold) from the test-values (for test-values, see for example: Lebart et al.; 1998, 2019).

7) **Drawing of supplementary edges** (color and thickness different from those of the edges of the additive tree) joining each column point (text) on the graph to its characteristic lines (words).

3. Validation and bootstrap (*reminder*).

To compute the precision of estimates, the classical analytical approach is unrealistic and complex.

The **bootstrap** (see: Efron and Tibshirani, 1993), on the contrary, makes almost no assumption about the underlying distributions, and gives the possibility *to deal with parameters computed through the most complex algorithms*.

The nonparametric bootstrap consists of drawing with replacement K samples of size n out of n statistical units.

Empirical evidence suggests that 30 is an acceptable value for K in the very specific multidimensional context of Principal Axes Methods.

We have at this stage K samples (the replicates) drawn from a new theoretical population defined by the empirical distribution of the original data set, leading to K estimates of the parameters of interest.

3. Validation and Bootstrap (*reminder*).

In our case, we will use the **Partial Bootstrap** (see Greenacre, 1984; Chateau and Lebart, 1996; Lebart, 2007): We wish to appreciate the stability and the significance of the **supplementary** (*a.k.a.* **external** or **illustrative**) categories (topics in our example).

One replication then consists of a drawing with replacement of the text units (the 154 Shakespeare's sonnets in the case of the example of section 4 below).

This leads to a vector of bootstrap weights, such as 0, 1, 2, 3… the sum of the weight being 154 for each replicate.

A replicated category is then a weighted barycenter of the text units. The process is the same for CA and AT.

The cloud of K replicates for one category allows for the drawing of a confidence ellipse containing 95 % of the replicates.

Examples of such confidence areas are given below for both CA and AT (Fig. 3.1, 3,2 and 4 below).

4. Application. Corpus and context : William Shakespeare : Sonnets

The corpus of **Shakespeare's 154 Sonnets**, will serve as a reference body to present Simultaneous Additive Trees (SAT) including 8 recognized “topics” and the bootstrap validation of the locations of these topics. The corpus is well known, translated into almost every language, deeply studied and commented.

Theme, Topic, Subject, Motif... (Reminder)

The definition of topics in Text Mining is pragmatic and can also cover the concepts of theme and motive (in the literary sense). Usually, a topic is the main subject, an objective explanation of the content of a text, whereas a theme represents a deeper underlying message. A motif is simply a recurrent idea used to reinforce the main topic.

Eight topics inspired by expert comments

Table 1. Series of 8 *a priori* topics and corresponding sonnets numbers

Procreation	1 - 17
YoungMan	20-25, 33-38, 40-42, 46, 47, 49, 53-55, 59-60,62-70, 75-77, 88-106, 108-112, 115-125,
DarkLady	127-136, 139, 140, 143-146, 153,154
Absence	26-32, 39, 43-45, 48, 50-52, 56-58, 61, 113-114
Storm	141,142,147-152
Rivalry	78-87
Death	71-74
Etern_poetry	18, 19, 81

The partition of sonnets given in Table 1 is inspired by the works of Alden (1913) and Paterson (2010) but not explicitly mentioned by these authors.

Table 2. List of characteristics words for the 8 *a priori* topics

(Minimum frequency for words: 10, then, test-values > 1.7)

Procreation	*beauty self world die age bear youth live time make*
YoungMan	*all never heart days time sun ever*
DarkLady	*black heart soul face one let well friend still*
Absence	*thought night day mind till being far woe think like*
Storm	*love eyes hate truth false see know best heart lies*
Rivalry	*praise worth making verse fair muse therefore others*
Death	*world death would life*
Etern_poetry	*men long live world summer death*

Specificity of poetic texts

- The statistical analysis of poetry and song collections is hampered by the specific nature of these texts.

  They may contain *choruses*, *deliberate repetitions*, *stylistic devices* (even coarse slang in the case of Shakespeare) or *versification constraints* that challenge the statistical significance of lexical frequencies.

- A pragmatic solution is to work with **"bags of words"** (the presence or absence of words, tokens, lemmas) which constitute only a tiny fraction of the texts' artistic aspect, but nevertheless prove useful for studying, for example, a work within its temporal context.

- Evidently, this is by no way an in-depth study or an elucidation of poetic texts, but simply a careful examination of lexical traces and patterns, which will ultimately prove to be of interest.

4.3 Insufficiency of Principal Axes

- Word bags (presence-absence coding of words in text units) are created from a certain size of text units (poems themselves in the case of poetic works). In our case, for Shakespeare, the text units will be the 154 Sonnets.

As a consequence, the maximum frequency of a word in the whole corpus cannot exceed the number of text groups. In the following example, we will keep the 259 words appearing in at least 9 sonnets.

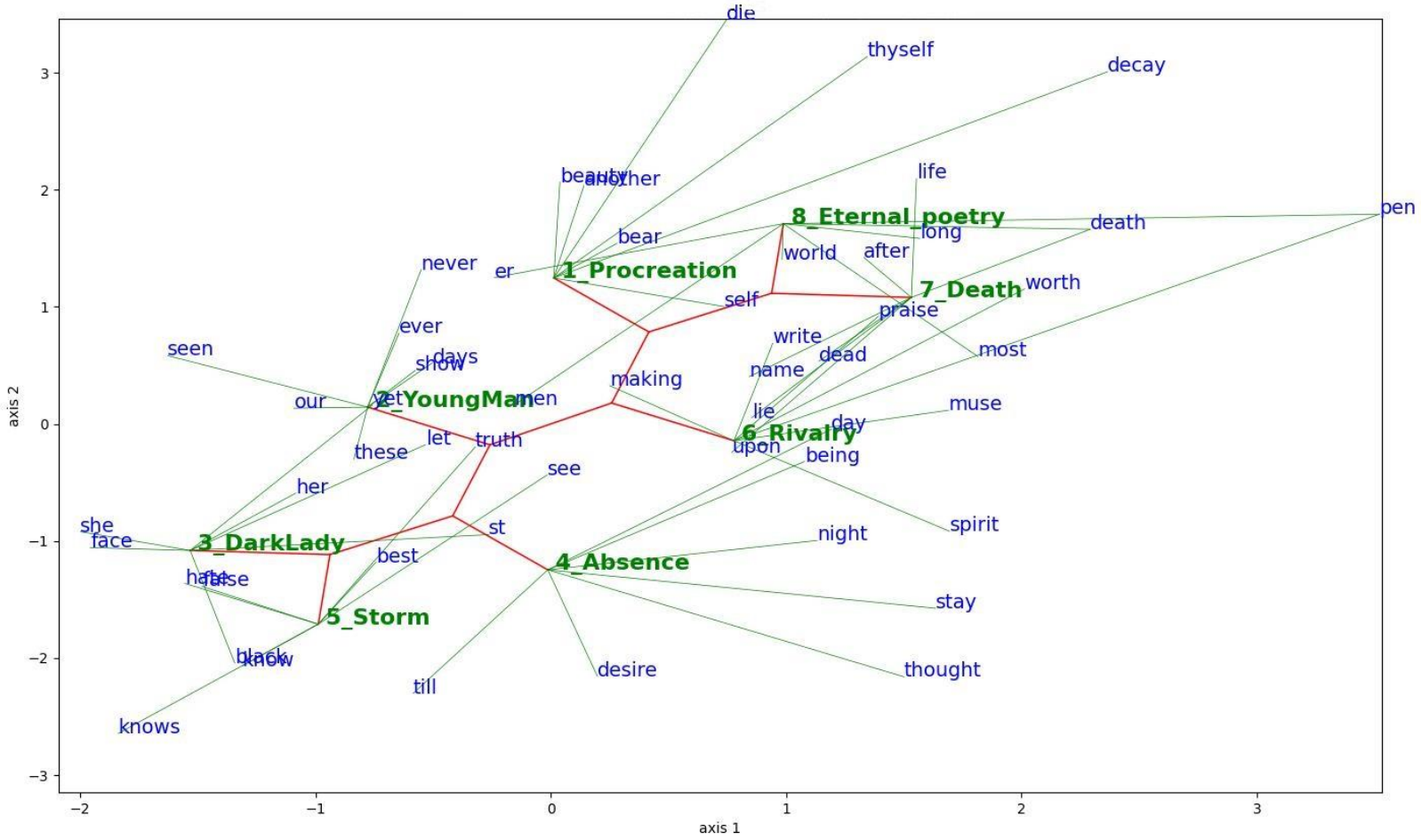
- Under these conditions, the dimensionality and sphericity of the word cloud rarely allow for good visualizations in a low-dimensional space.

In the case of CA, the eigenvalues will be poorly separated, a source of instability for the principal axes.

- The two-dimensional representation of words and texts, an undeniable advantage of CA, becomes insufficient, and additive trees become indispensable.

5. Implementation . Example of visualization: William Shakespeare sonnets (Fig.1)

Figure 1. Aggregated lexical table. The 8 topics are used to construct the lexical table (8 topics x 259 words [word frequency > 9]). Figure 1 shows the SAT linking the 8 topics, terminal elements of the tree (its "leaves"), with the locations of the words (which are centroids of the topics). The words are linked to the topics they most characterize (criterion: test values). We are dealing with *vocabulary* and not *lexical frequency*: each sonnet is reduced to a *bag of words* .

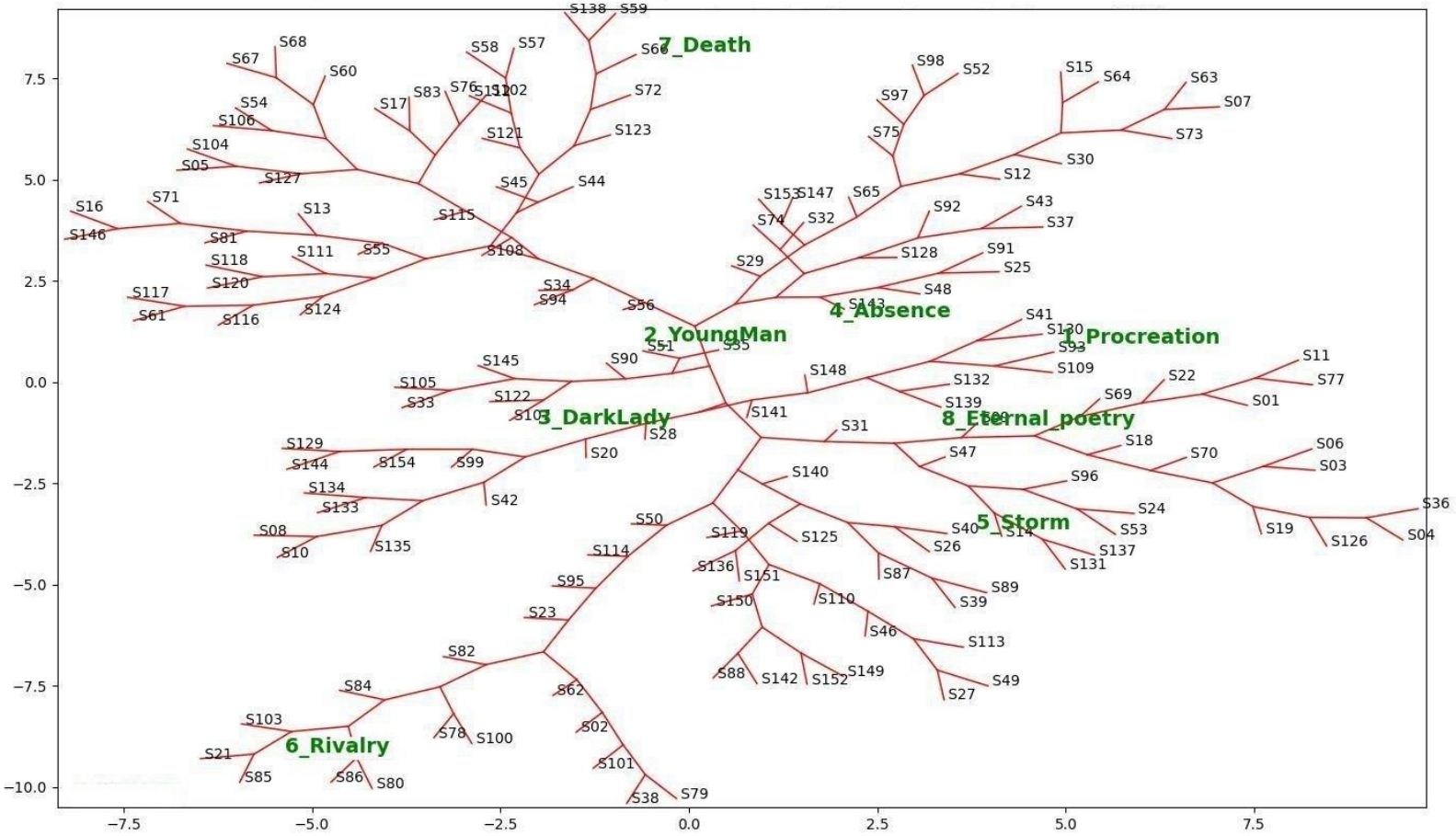


5. Implementation . Example of visualization: William Shakespeare sonnets (Fig. 2)

Figure 2. Direct Simultaneous Additive Tree.

SAT of the lexical table (154 Sonnets x 259 words) (the 154 sonnets being identified from “S01” to “S154”). The topics are now located ***a posteriori***. As in CA, they are centroids of the implied sonnets, these centroids being expanded to equalize the dispersions of the sonnets and of the topics.

For example, we can see that the topic "Death" (at the top) is particularly far removed from the topic "Rivalry" (at the bottom left).



5. Implementation . Example of visualization: William Shakespeare sonnets (Fig. 3.1)

Figure 3.1. Direct CA with bootsrap confidence areas (ellipses).
First principal plane of a CA of the same lexical table as Figure 2. The words are still omitted for readability, but the topics are shown as supplementary elements. We will focus, in the next figure, on the two topics : Death and Rivalry.

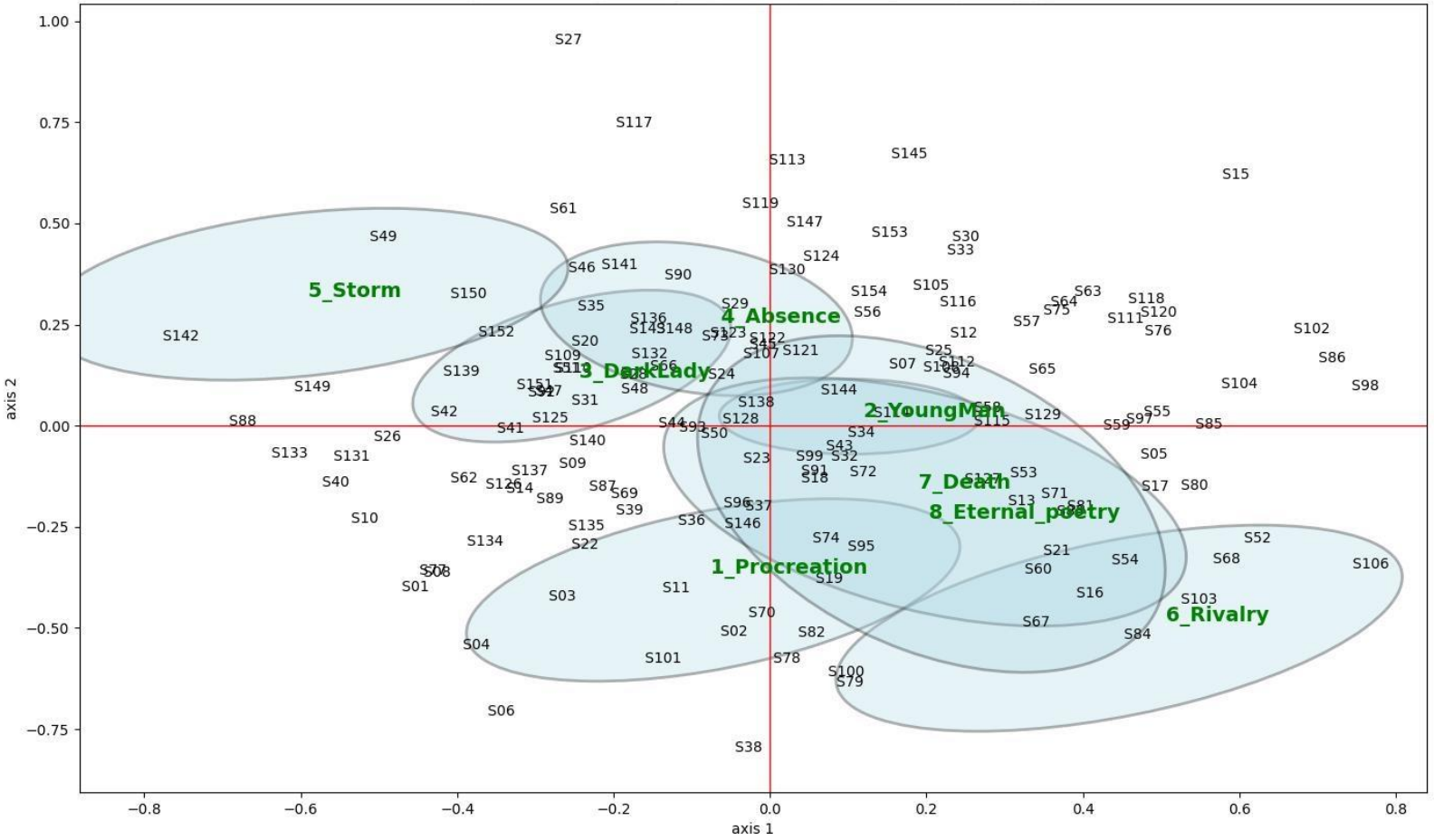
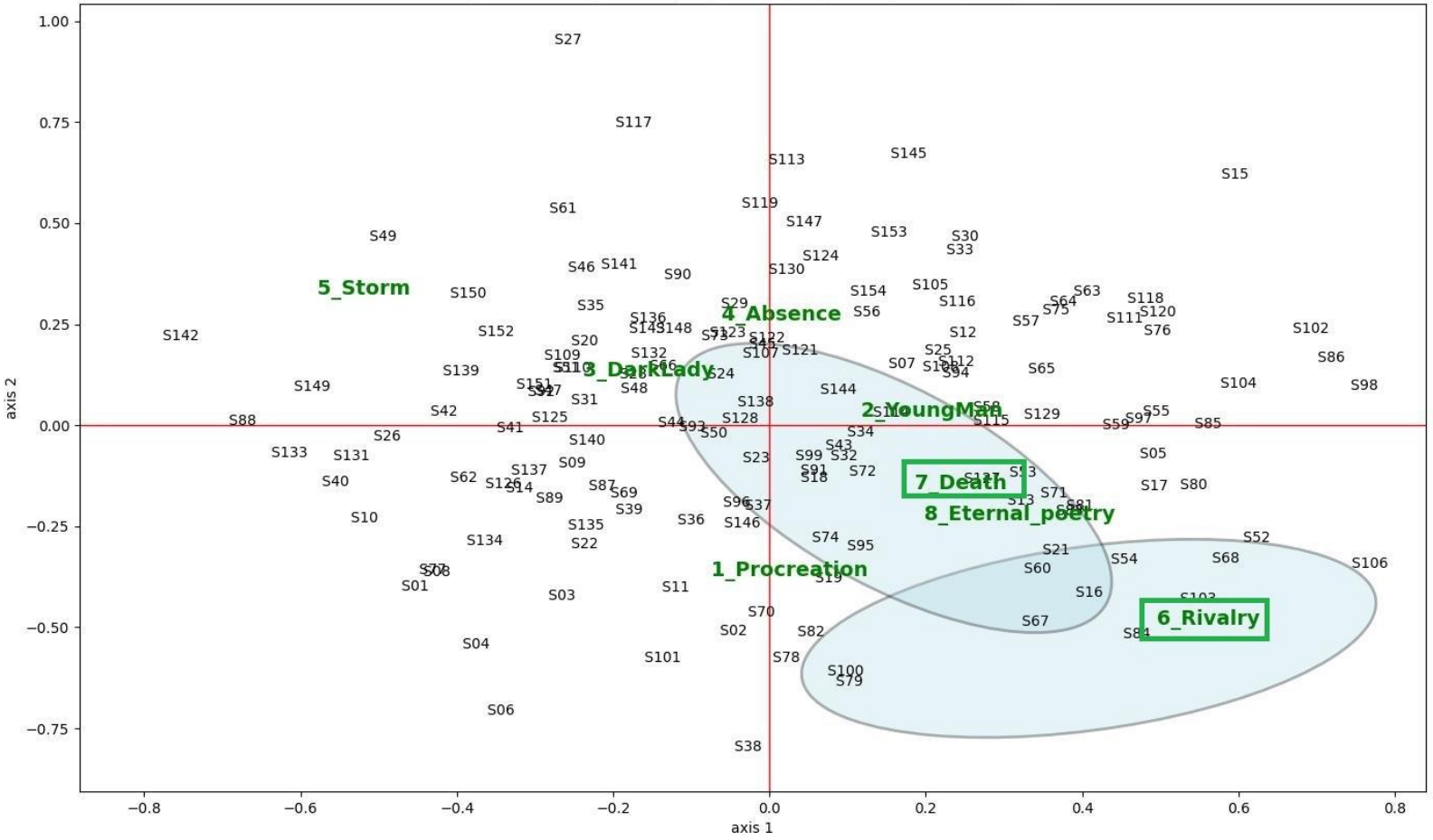


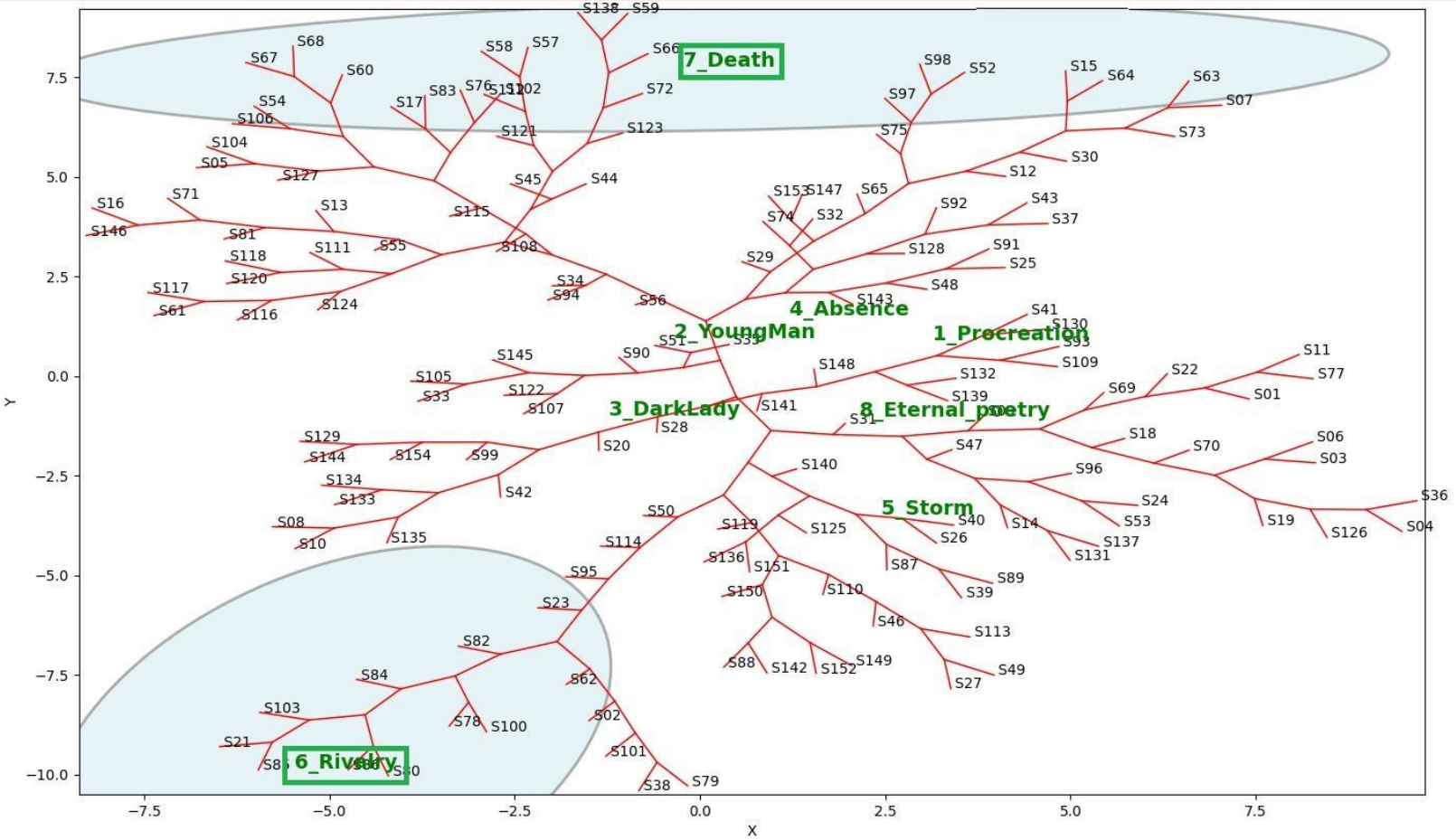
Figure 3.2. Same direct CA with 2 specific bootsrap confidence ellipses.
Same first principal plane as Figure 3,1. Only two bootstrap confidence ellipses for the position of two topics are drawn: Death and Rivalry. We can see that the topics are not as far apart as in Figure 2, but also that their confidence ellipses have a fairly clear intersection, which casts doubt on the significance of their distinction.



5. Implementation . Example of visualization: William Shakespeare sonnets (Fig. 4)

Figure 4. Direct SAT with 2 bootsrap confidence ellipses.

Finally, same SAT as in Figure 2, but now with bootstrap confidence ellipses for the two boxed topics. It is clear that these topics are not only distinct but almost opposite, something the principal plane of the CA, limited to two dimensions, could not detect. [One could argue that it is possible to observe such divergences by carefully analyzing the whole sequence of principal axes beyond axis 2. However, practitioners know that this is both difficult and time consuming].



6. Conclusion (1)

Let us remind the seminal remark by Le Cun *and al.* ... in 2015

"Unsupervised learning had a catalytic effect in reviving interest in deep learning but has since been overshadowed by the successes of purely supervised learning. (…) We expect unsupervised learning to become far more important in the longer term. Human and animal learning is largely unsupervised: we discover the structure of the world by observing it, not by being told the name of every object…."

Le Cun, Bengio & Hinton, Deep Learning, Nature, 2015.
(up to 2026 July 1st, 115,937 quotations…)

Data visualization methods (DVM) certainly use algebraic or algorithmic methods similar to those of Artificial Intelligence. In some respect, CA is also a neuronal method (Lebart, 1997). Despite several similarities with AI, DVM occupy a particular place in a knowledge process.

A visualization is neither a simple task to perform nor a decision to make. We submit our texts to understand and reflect. The approach is mainly unsupervised.

The “argument from disability” makes the claim that “a machine can never do **X**.” As examples of **X**, **Alan Turing** lists the following:

« Be kind, resourceful, beautiful, friendly, have initiative, have a sense of humor, tell right from wrong, make mistakes, fall in love, enjoy strawberries and cream, make someone fall in love with it, learn from experience, use words properly, be the subject of its own thought, have as much diversity of behavior as man, do something really new ».

6. Conclusion (final)

When visualizing textual data, the priority is not systematically "recognition" or "decision" but discovery, description, comparison, understanding, detection of new "statistical facts".

In SAT, as in correspondence analysis with its simultaneous representations, the observable pattern of column points (texts, collections) tells us **how** these points are organized. The presence in the same visualization of row points (words) and additional variables (topics, periods) tells us **why** they are organized in this way: texts are similar because they often use the same words or deal with the same topics. **But those visualizations also need to be assessed.**

It has been said, about simultaneous additive trees, that words and categories dress the skeleton of the additive tree.

But this dressing would be merely a disguise in the absence of validation procedures.

The non-parametric validation proposed here is an attempt to give these visualizations a scientific status.

7. Python Software note (1)

1.

Sketch of the interface

2.

This chain uses the following standard Python modules:
Tkinter (interface, graph framing), *os*, *Lib*, *numpy*, *pandas*, *matplotlib*, *igraph*, *importlib*, *textwrap*, *re* (regular expression), *sklearn*, and *scipy*.

The interface allows the users to select folders, files, and parameters:
The section: ***1) Selection of folder*** looks for the basic folder containing data files and modules.
The section: ***2) Select files***:

Select basic_text:

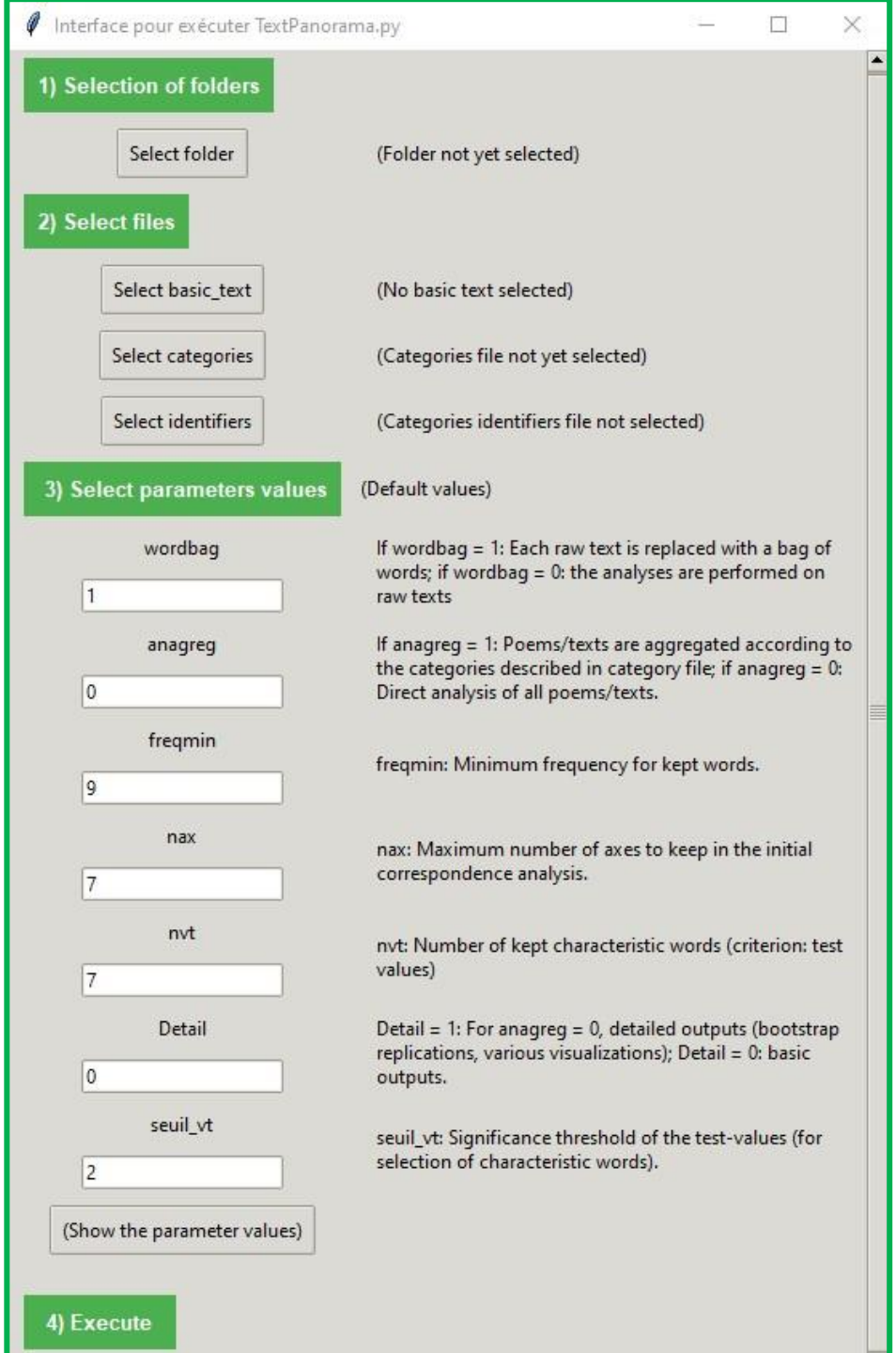
(raw input texts : tokens, lemmas, etc.)

Select categories:

(XL file (.csv) file describing categories of texts)

Select identifiers:

(XL file (.csv) file describing the identifiers of these categories)



7. Python Software note (2)

Sketch of the interface (2)

The section:
3) Select parameters values registers the values of 7 parameters:

wordbag: (If wordbag = 1: Each raw text is replaced with a bag of words; if wordbag = 0: the analyses are performed on raw texts.)

anagreg: (If anagreg = 1: Poems/texts are aggregated according to the categories described in category file; if anagreg = 0: Direct analysis of all poems/texts, leading to validation procedures.)

freqmin: (numerical) (Minimum frequency for kept words.)

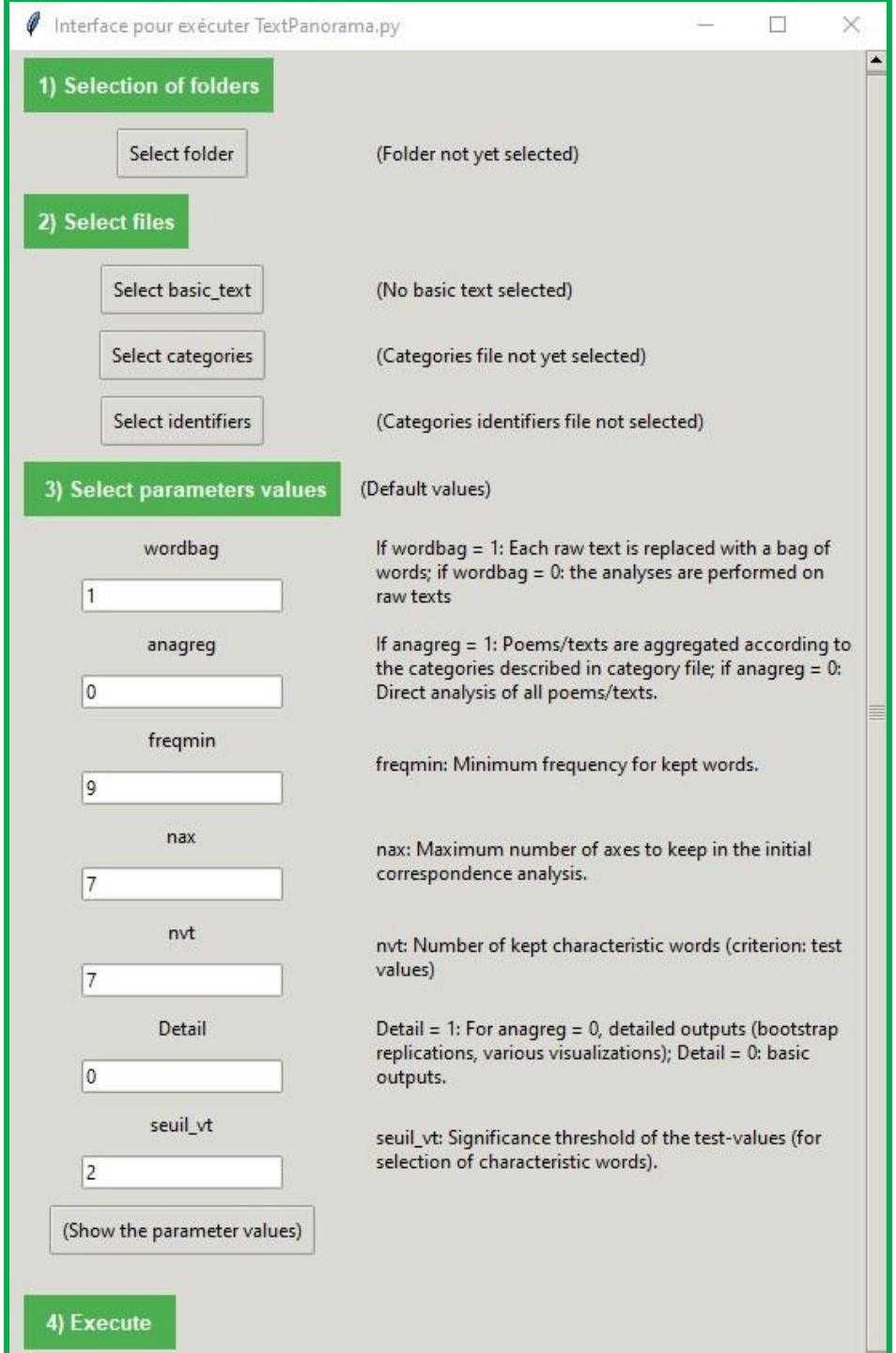
nax: (numerical) (Maximum number of axes to keep in the initial correspondence analysis)

nvt: (numerical) Number of kept characteristic words (criterion: test values)

Detail: Detail = 1: For anagreg = 0, detailed outputs (various visualizations); Detail = 0: basic outputs.

seuil_vt: (numerical) Significance threshold of the test-values (for selection of characteristic words).

3.



8. References (1)

Alden, R. M. (1913). *Sonnets and a Lover's Complaint.* New York: Macmillan.

Barthélémy J.-P. and Guénoche A. (1988). *Les arbres et les représentations de proximité*. Paris: Masson.

Bryant D. (2005). On the uniqueness of the selection criterion in Neighbor-Joining. *Journal of Classification,* vol. (22), 1: 3-16.

Buneman P. (1971). The recovery of trees from measurements of dissimilarity. In: Hodson F. R. D. Kendall G., and Tautu P., (Editors). *aMathemtics in the archeological and historical sciences.* Edinburgh University Press, Edinburgh: 387-395.

Chateau, F. and Lebart, L. (1996). Assessing sample variability in visualization techniques related to principal component analysis: bootstrap and alternative simulation methods. In : A. Prats (Ed): *COMPSTAT96,* Physica Verlag, Heidelberg, 205-210.

Di Battista, G. Eades, P., Tamassia R., and Tollis, I.G. (1999). *Graph Drawing: Algorithms for the Visualization of Graphs*, Englewood Cliffs: Prentice. Hall.

Efron, B. and Tibshirani, R. J. (1993). *An Introduction to the Bootstrap*, Chapman and Hall: New York.

Eades P. (1984). A heuristic for graph drawing. *Congressus Numerantium*, 42:149–160.

Fruchterman T. and Reingold E. (1991). Graph drawing by force-directed placement. *Softw. – Pract*. Exp., 21(11):1129–1164.

Greenacre, M. (1984): *Theory and Applications of Correspondence Analysis*. Academic Press, London.

8. References (2)

Huson D. H. and Bryant D. (2006) Application of Phylogenetic Networks in Evolutionary Studies. *Molecular Biology and Evolution*, 23 (2): 254 - 267. Software available from www.splitstree.org.

Kamada T. and Kawai S. (1989). An algorithm for drawing general undirected graphs. *Inform. Process. Lett.*, 31:7–15.

Kobourov, S. G. (2013). Force-Directed Drawing Algorithms. in: *Handbook on Graph Drawing and Visualization.* Chapman and Hall/CRC.

Lebart L. (1997). Correspondence analysis, discrimination and neural networks. In: *Data Science, Classification and Related Methods.* Hayashi C., Ohsumi N., Yajima K., Tanaka Y., Bock H.- H., Baba Y. (eds). Berlin : Springer, 423-430.

Lebart L. (2007). Which bootstrap for principal axes methods? In: *Selected Contributions in Data Analysis and Classification.* P. Brito *et al.* (eds), Springer: 581-588.

Lebart L. (2024). Des outils pour décrire certains corpus de poèmes et de chansons : Les arbres additifs simultanés. (2024). In: *JADT 2024, "Mots comptés, Textes déchiffrés"* Anne DISTER et Dominique LONGREE (eds) Vol. 2, PUL, (Presses Universitaires de Louvain) : 553- 562.

Lebart L., Pincemin B. and Poudat C. (2019). *Analyse des Données Textuelles.* Presses de l'Université du Québec, Québec, Canada.

Lebart L., Salem A. and Berry L. (1998). *Exploring Textual Data,* Kluwer, Dordrecht.

LeCun Y, Bengio Y. and Hinton G. (2015). Deep learning. *Nature*. May 28;521(7553):436-44.

Luong X. (1988). *Méthodes d'analyse arborée. Algorithmes, applications.* Thèse pour le doctorat ès sciences. Université Paris V.

8. References (3)

Mihaescu R., Levy D. and Pachter L. (2009). Why Neighbor-Joining works? *Algorithmica*, vol. (54): 1-24.

Paterson D. (2010). *Reading Shakespeare Sonnets*. Faber & Faber Ltd. London.

Saitou N. and Nei M. (1987). The neighbor joining method: a new method for reconstructing phylogenetic trees, *Molecular Biology and Evolution*, vol. (4), 4: 406-425.

Sattath S. and Tversky A. (1977). Additive similarity trees. *Psychometrika*, vol. (42), 3: 319-345.

Shakespeare, W. (1901). *Poems and sonnets: Booklover's Edition*. Ed. The University Society and Israel Gollancz. New York: University Society Press. Shakespeare Online. Dec. 2017.

Tutte, W. T. (1963). How to draw a graph. *Proc. London Math. Society*, 13(52):743–768.

More *(and Python software after July 15th)* on: www.dtmvic.com

Thank You

Gracias

Grazie

Domo Arigato

Obrigado

Merci

Danke

Choukrane

Questions ? *ludovic@lebart.org*

28