Experiment with the stylo package in R.

## **Installation**:

install.packages("stylo")

## **Sample datasets**:

Pull in the data from GitHub to the DH23 folder: in GitBash / Terminal, go to the DH23 folder on the Desktop, and write the command “git pull”.

=> the new data will be in a new “session\_8\_stylometry” folder:

* stylo\_nahda\_novels : a collection of 65 Arabic novels from the 19th C, by 6 authors
* stylo\_nahda\_novels\_3 : a smaller collection of 21 Arabic novels from the 19th C, by 3 authors. Includes one mystery novel. Who is likely to have written it?
* translated\_novels: a collection of 58 novels translated into Arabic. Contains two sub-folders: one in which file names start with the name of the author, another in which file names start with the name of the translator.
* RdlR\_for\_rolling\_classify: a dataset to test rolling stylometry

## **Experiments:**

### 1. Nahda novels:

The session\_8\_stylometry folder contains a folder called "stylo\_nahda\_novels", which contains a collection of 65 Arabic novels from the 19th C, by 6 authors.

This dataset was used by Maxim Romanov to test which parameters work best with texts in classical Arabic.

Take a look at the folder structure in your file explorer:

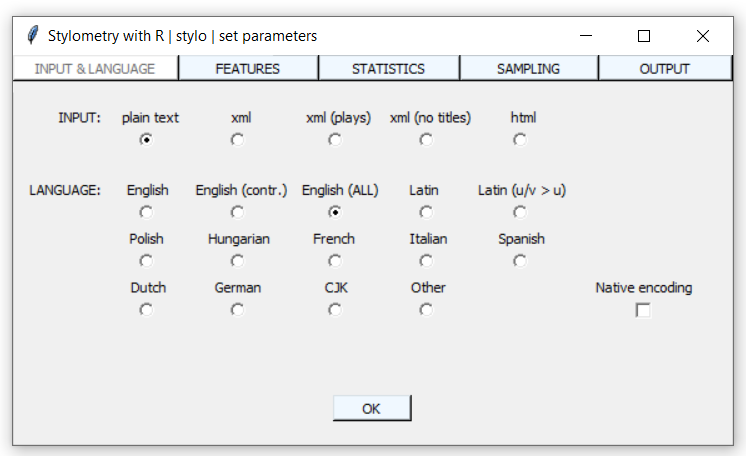
* the texts are inside a subfolder called “corpus”: stylo will automatically find your texts if they are in a subfolder with this name, inside the curent working directory. If not, you will have to provide the path to the folder manually.
* The file names have a similar structure: <date+author>\_<title>.txt. The first element of the file name (before the (first) underscore) will be used by stylo to assign a color to each group of texts. It therefore makes sense to use the author’s name as the first element if you are interested in the style of each author

**Question**: if you’d be interested in the gender characteristics in your corpus, how would you name your files? Use as an example Mary Shelley’s *Frankenstein* (she’s a female author) and Sir Walter Scott’s *Ivanhoe* (he’s male):

…  
…

Set the working directory in RStudio to the “stylo\_nahda\_novels” folder.

Now load stylo from your package library by running the command “library(“stylo”)”, and run stylo by calling the `stylo()` function. A popup should appear that looks like this:



**Troubleshooting:**

If this popup does not show up, check if you can see an icon with a feather among your open programs at the bottom of your screen, and click it:



If you are on a Mac and RStudio prints an error message "GUI could not be launched", it is likely because you have not installed XQuartz prior to installing R. If you hadn't installed XQuartz, do it now, and re-install R, RStudio and stylo (instructions for installing XQuartz and R here: <https://openiti.org/2021IslamicateWorldCourse/week-8-introducing-r-in-gui-through-stylometry-1.html#on-mac>; and for RStudio here: <https://openiti.org/2021IslamicateWorldCourse/week-8-introducing-r-in-gui-through-stylometry-1.html#installing-rstudio>). **Restart your computer before trying to start up stylo**.

If you are still experiencing trouble, see https://github.com/computationalstylistics/stylo#installation-issues for more possible solutions – and ask the instructors for help.

In the first tab (“INPUT & LANGUAGE”), don’t forget to set the language setting to “Other” – this will make sure Arabic texts is tokenized correctly.

Click OK to run the analysis a first time, with default parameters. It will take a minute or so.

Once it finished running (a “>” prompt will appear in your console), go to a file explorer and look at the files that were created by the script in the “stylo\_nahda\_novels” folder (look at the time stamps to check which files were created when you ran the function). Open each of the files and describe what they contain:

* … (file name) : … (description)
* … : …
* … : …
* … : …

NB: deleting the “stylo\_config.txt” will reset all parameters to their defaults. **Delete this file now.**

Run the stylo() function again; in the “INPUT & LANGUAGE” tab, set the “LANGUAGE” option to “Other” again.

Open the “OUTPUT” tab in the popup; here you can adapt the output created by the function. By default, the graphs stylo produces are only shown “onscreen” in RStudio’s Plots panel. Check the “PNG” box to save each graph as a PNG image file. Each time you run stylo now, it will create a PNG image file of that graph in the “stylo\_nahda\_novels” folder.

Click the OK button; stylo will run the analysis again. Once it is finished, insert the output PNG file here:

<insert file here>

NB: the labels in the graphs are the file names of the texts; they are coloured by the first element of the filename (the element before the first underscore, in our case, the author’s name). This colour is only meant to make it visually easier to see whether texts from the same “class” – in our case, texts by the same author – cluster together in the analysis; the information from the file name is not at all used by stylo in the analysis.

To test this**, go to the “corpus” folder** inside the “stylo\_nahda\_novels” folder, and replace the author’s name in one of the filenames by “ANONYMOUS”; in the next graphs we will produce, this text will now have a different colour, and you can check easily whether it still clusters with the other texts by the same author.

Run the stylo() function again; in the “INPUT & LANGUAGE” tab, make sure the “LANGUAGE” option is set to “Other” (do this every time; this is the last time we will say this explicitly!).

Now open the “FEATURES” tab; this tab describes which and how many linguistic features stylo will use in its analysis.

NB: ngrams are sequences of a fixed number of words or characters. For example, the sentence “This is a sample string containing 8 words” can be split into the following ngrams:

* word 1-grams (“unigrams”): “This”, “is”, “a”, “sample”, “string”, …
* word 2-grams (“bigrams”): “This is”, “is a”, “a sample”, “sample string”, …
* word 3-grams (“trigrams”): “This is a”, “is a sample”, “a sample string”, …
* …
* character 1-grams: “T”, “h”, “i”, “s”, “ ”, “i”, “s”, “ ”, “a”, “s”, “a”, “m”, “p”, …
* character 2-grams: “Th”, “hi”, “is”, “s ”, “i “, “is”, “s ”, “ a”, “a ”, “ s”, “sa”, …
* character 3-grams: “Thi”, “his”, “is ”, “s i”, “ is”, “is “, “s a”, “ a ”, “a s, …

Set the “FEATURES” to “chars” (that is, characters), and the ngram size to 4.

Click “OK” to run the script again. Once it finished, open the “wordlist.txt” file in the “stylo\_nahda\_novels” folder. What has changed in this file since you last opened it?

…

What are the most frequent character 4-grams in the corpus?

…

Insert the output PNG file here:

<insert file here>

Compare it to the previous output, which was produced using the default word 1-grams. Describe the difference between the two images.

…

Which of the two seems to have worked best? (criterium: texts of the same author should be clustered together)

…

Run the stylo() function again.

In the “FEATURES” tab, check the “Existing frequencies” checkbox in the “VARIOUS” section. Now stylo will not load all the texts again but reuse the frequencies it has calculated before and stored in the “table\_with\_frequencies.txt” file. You’ll notice that this speeds up the process enormously (results are now produced in a second or less).

The second row of settings in the “FEATURES” tab is “MFW settings”; MFW stands for “Most Frequent Words”. It defines how many of the most frequent features – in our case, character 4-grams – stylo will use in its analysis.

Keeping the “FEATURES” set to character 4-grams, change the MFW settings to see whether the results improve or get worse. Please make sure you enter the same value for “Minimum” and “Maximum” for now – we’ll vary those later.

Do this at least four times (it should be really fast, thanks to the “Existing frequencies” function), with different values, and insert the output PNG files here:

<insert file here>

Describe the changes:

…

<insert file here>

Describe the changes:

…

<insert file here>

Describe the changes:

…

<insert file here>

Describe the changes:

…

Instead of manually changing the MFW parameters, and choosing which ones work best, we can ask stylo to run the analysis with a lot of different parameters and build a consensus graph based on the different results.

Run stylo again.

In the “STATISTICS” tab, select “Consensus Tree” instead of “Cluster Analysis”.

Now open the FEATURES tab. Make sure “Existing frequencies” is still checked. We are now going to set a range of MFW settings, instead of just one. To do this, set the “Minimum” value to, for example, 10, and the “Maximum” value to, for example, “1000”. Then set the “Increment” value: this will decide the number of steps between the Minimum and Maximum value: if you set “Increment” to 1, stylo will run the analysis 990 times (once for 10 MFW, once for 11, …); if you set it to 10, it will run it 90 times (once for 10 MFW, once for 20 MFW, …). I’d set it to 10.

Insert the output png file here:

<insert file here>

Describe what you see:

…

### 2. Authorship – Translation signals

**This exercise is optional – no grades to be won here!**

The “session\_8\_stylometry” folder also contains a “translated\_novels” folder. This contains two subfolders, “author\_first”, and “transl\_first”, each of which contains the same collection of texts, 58 English novels translated into Arabic, by three different authors (Doyle, the author of the Sherlock Holmes series; and two science fiction authors, Wells and Mitchell); and eight different translators. The only difference between the two folders is the file names:

* In the “corpus” subfolder of “author\_first”, file names are <author>\_<translator>\_<title>.txt;
* In the “corpus” subfolder of “author\_first”, file names are <translator>\_<author>\_<title>.txt;

The aim of the experiment is to see whether stylo picks up the style of the author or the style of the translator – or both.

What do you expect? Will texts cluster by author, or by translator? Please write down what you think now, before doing the experiment – there are no wrong answers!

…

Set the working directory to the “author\_first” folder, and run stylo with the default settings, but make sure it produces a PNG output file (and the language is set to “Other”, of course!).

Insert the PNG file here:

<insert file here>

Describe what you see: do all/most/some texts by the same author cluster together?

…

Set the working directory to the “transl\_first” folder, and run stylo with the default settings.

Insert the PNG file here:

<insert file here>

Describe what you see: do all/most/some texts by the same translator cluster together?

…

Check the “Existing frequencies” checkbox. Now vary the MFW parameter (Minimum and Maximum the same value) a couple of times, and try to see if by changing the parameters, it is possible to make the text cluster more by translator?

<insert file here>

<insert file here>

<insert file here>

<insert file here>

Describe what you see:

…

Take a good look at the first 30 words in “wordlist.txt” in the “transl\_first” folder (it will be identical to the one in the “author\_first” folder).

Do you see a strange word among the first 30 words? What do you think the effect of this word could be on the analysis?

…

In order to deal with such words that are present only in a subset of the text, we can set yet another parameter in the “FEATURES” tab: “CULLING”. This will take words from the word list into account if they appear in minimum X percent of the texts in the corpus. If you set the culling value to 100 (both Minimum and Maximum), only words that appear in all of the texts in the corpus will be taken into account. If you set it to 75, only words that appear in at least 75 percent of the texts in the corpus will be taken into account.

Does setting the Culling parameter to 75% change the clustering of texts by the same translator?

…

What is your conclusion about this experiment?

…