Assignment number 1
Probabilities and entropy of natural language

Deadline: March, 22

Let `txt` be a character string. Let X and Y be the pair of random variables whose values are two consecutive letters of `txt` taken randomly. Compute the probability distributions of the pair (joint), of each variable (marginal), and the conditional probabilities of the random variables $Y|x_i$ for each letter x_i. Compare the distributions of X and Y.

Creation of new text. The first objective is to create "artificial" text with the same distribution of probabilities of letters than a given text `txt`. You will need to import functions from the `random` package. Implement the following functions:

- `random_text(txt)` outputs a random text with the same letter probabilities of the input text `txt`.
- `random_text_joint(txt)` outputs a random text with the same joint (and conditional) probabilities than the input text `txt`, with respect to pairs of consecutive letters.

With a large text (millions of letters) belonging to natural language do the following: save the original text, and also the two artificial random texts you created with the same probabilities distributions, in three different files. Compress the three files using a standard compressor (for example Zip or 7-Zip). Compare the results.

Computation of entropies. Implement the following functions for computing information-theoretic properties of the string `txt`:

- `entropy(txt)` computes the entropy $H(X)$ of the variable X.
- `joint_entropy(txt)` computes the joint entropy $H(X, Y)$ of the pair X, Y.
- `conditional_entropy1(txt,ltr)` computes the entropy $H(Y|x_i)$ of the random variable $Y|x_i$ whose value is a random letter of `txt` after the given letter $x_i =$ `ltr`. Here `ltr` denotes one of the letters of `txt`.
- `conditional_entropy(txt)` computes the conditional entropy $H(Y|X)$.

Use these functions to experiment with texts in natural language. In particular, compute and compare the several entropies for different texts (same author, different author but same language, different languages, etc.), and also the entropy of the first letter (resp. the last letter) of a word. Which are the letters x_i where $H(Y|x_i)$ reaches its maximum and minimum values?

Check your functions using the identity

$$\sum_{i=1}^{m} p(x_i)h(Y|x_i) = H(Y|X) = H(X,Y) - H(X).$$

Delivering. Deliver a single .py file including your implementations of the six functions. Remember that:

- The names of the functions must be exactly the given ones.
- The parameters of the functions are:
  - `txt`: a character string of arbitrary length;
  - `ltr`: a character string of length one (a letter).
- The functions return:
  - a character string of the same length of the input the two random text functions.
  - a single floating-point number the four entropy functions;
- The .py file you deliver does not include data reading from or writing to files.