Simulations and algorithmic applications of Markov chains 19.02.2020

BONUS task, encryption

Paweł Lorek

This document describes a way some message was encrypted. First person who will decrypt it *using Markov chains*, and follows the instruction given in the beginning of the message, will get **bonus points**.

Define a set of characters (lower and upper case letters plus space ' ')

$$\begin{aligned}\Sigma = \{&\text{' ', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u',}\\ &\text{'v', 'w','x', 'y', 'z', 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O',}\\ &\text{'P', 'Q','R', 'S','T', 'U', 'V', 'W', 'X', 'Y', 'Z'}\}\end{aligned}$$

Note that $|\Sigma| = 53$.

Substitution cipher. Assume we have a message which consists only of characters from Σ (i.e., all other characters should be removed first). Denote the message by (each $m_i \in \Sigma$)

$$\mathbf{m} = (m_1, m_2, \ldots, m_L).$$

Consider a 1-1 mapping on Σ, i.e., a permutation

$$\sigma : \Sigma \to \Sigma.$$

We encrypt the message substituting a letter $\chi \in \Sigma$ with $\sigma(\chi)$ obtaining a cryptogram

$$\mathbf{c} = (c_1, \ldots, c_L) = (\sigma(m_1), \sigma(m_2), \ldots, \sigma(m_L)).$$

After choosing a random permutation σ a message of length 10015 was encrypted in the above way.

- The encrypted message is available at

  `www.math.uni.wroc.pl/~lorek/teaching/pliki/2020sim_alg_mc_encrypted_message.txt`

- The set of characters Σ is available at

  `www.math.uni.wroc.pl/~lorek/teaching/pliki/2020sim_alg_mc_characters.txt`

Task. Decrypt the message. Find "optimal" σ, or some "close" σ_0, such that when applied to cryptogram $\mathbf{c}$ you will get

$$\mathbf{m}' = (\sigma_0^{-1}(c_1), \ldots, \sigma_0^{-1}(c_L))$$

you will be able to read the message.
Note that there are $53! \approx 4.27 \times 10^{69}$ all possible permutations.