Supplementary materials to "LZW-Kernel: fast kernel utilizing variable length code blocks from LZW compressors for protein sequence classification"

Gleb Filatov[1], Bruno Bauwens[2], and Attila Kertész-Farkas*[1]

[1]School of Data Analysis and Artificial Intelligence, Faculty of Computer Science, National Research University Higher School of Economics (HSE)
[2]Big Data and Information Retrieval School, Faculty of Computer Science, National Research University Higher School of Economics (HSE)

April 18, 2018

1 The pseudo code of the LZW compression algorithm

Algorithm 1 Calculation the length of compressed string The input is a finite string over a finite alphabet. This procedure returns only the length of the compressed string.

```
1:  procedure LZW-Compression(string s)
2:      Len=0;                                  ▷ Count the length of the compressed string
3:      Code=λ                                  ▷ Code initialized as an empty string
4:      Dictionary=set of ASCII characters      ▷ A Dictionary of codes
5:      while data to be read do
6:          ch = read next character from s
7:          if Code + ch ∈ Dictionary then
8:              Code = Code + ch
9:          else
10:             Dictionary = Dictionary ∪ Code+ch    ▷ Add code C to Dictionary
11:             Len = Len + 1
12:             Code = ch
13:         end if
14:     end while
15:     return Len                              ▷ Return the length of the compressed string
16: end procedure
```

2 Sequence clustering on the Alfree

We tested LAK, MMK, and our LZW-Kernel method on the Alfree benchmark dataset [1], which was constructed based on the ASTRAL v2.06 dataset [2] from 6569 protein sequences that share less than 40% identity to each other. Protein sequences were compartmentalized at four levels, yielding 513 family groups, 282 superfamilies, 219 folds, and 4 classes. Alfree contains a python package called Alfpy that has 38 (mainly alignment-free) sequence comparison measures and it is freely available at http://150.254.122.234:8000.

*Correspondence with akerteszfarkas@hse.ru

Alfree uses ROC analysis to measure the clustering ability of the sequence comparison measures in the following way: a protein sequence pair is considered "positive" if both proteins belong to the same class, fold, superfamily, and family (respectively); otherwise, the pair is considered "negative". Therefore, the resulting AUC shows how the distribution of similarity/distance scores of protein sequences from the same group is distinct from the distribution of similarity/distance scores of proteins belonging to different groups.

The ROC evaluation of the performance of unnormalized sequence similarity measures (SW, LAK, LZW-Kernel, MMK, and NGD) shows a ranking different from the one we obtained in protein classification. The Table 1 shows the results. Surprisingly, NGD, SW, and LZW-NCD outperformed the unnormalized kernel functions MMK, LAK, and LZW-Kernel by large margins; moreover, the kernel methods were hardly better than a random scoring function on the class level. This was unanticipated because kernel functions perform well on remote protein homology detection and protein classification tasks at higher levels of the SCOP and the CATH hierarchy in the PCB dataset, as shown in Table 3 of our manuscript. Furthermore, the NGD outperformed all other methods on the Alfree dataset but fell behind these methods on the gold-standard dataset SCOP1.53 in terms of protein classification.

We attribute this dramatic change to the ROC evaluation procedure rather than to differences in the versions of the SCOP datasets. The protein groups are known to be heterogeneous in many different characteristics. For instance, different protein groups have different sizes; in some groups, proteins are more conserved, while proteins in other groups may be related to each other loosely according to some sequence similarity measures. In some groups, proteins may be longer than average, while shorter in others. Therefore, a similarity value for a protein pair from the same group s may indicate a strong relationship, while the same score s may indicate a weak relationship with other protein groups. Thus, we think the normalization of similarity scores would help obtain better AUCs. We tried the following two normalization techniques:

$$k^*(x,y) = \frac{k(x,y)}{\sqrt{k(x,x)}\sqrt{k(y,y)}} \tag{1}$$

and

$$\tilde{k}(x,y) = -\frac{\max(k(x,x),k(y,y)) - \min(k(x,x),k(y,y))}{k(x,y) - \min(k(x,x),k(y,y))}. \tag{2}$$

The first normalization technique is very commonly used in bioinformatics. The second normalization technique is used in NGD [3]. Both normalization techniques significantly improved the AUCs of the similarity methods and results shown in Table 1. For instance, the first normalization technique improved LAK from 0.57 to 0.76 in mean AUC, and it achieved similar performance to that obtained by NGD, while the second normalization technique greatly improved the performance of the LZW-Kernel, from 0.52 to 0.73 in mean AUC, and it even outperformed the SW method. In our opinion, normalization has a great impact on the AUC results.

The LZW-Kernel took 151 seconds to calculate all pairwise similarities in the Alfree benchmark dataset. This would have been the sixth fastest method in the Time table (at the bottom at http://150.254.122.234:8000/benchmark/#tooltips). Although these tests were run on different computers, we still think LZW-Kernel would remain among the fastest in a fair comparison.

In summary, the LZW-Kernel would be among the four best and six fastest methods among the methods in the Alfree benchmark dataset.

References

[1] Andrzej Zielezinski, Susana Vinga, Jonas Almeida, and Wojciech M Karlowski. Alignment-free sequence comparison: benefits, applications, and tools. *Genome biology*, 18(1):186, 2017.

[2] Naomi K Fox, Steven E Brenner, and John-Marc Chandonia. Scope: Structural classification of proteins — extended, integrating scop and astral data and classification of new structures. *Nucleic acids research*, 42(D1):D304–D309, 2013.

[3] Lee Jun Choi et al. Adapting normalized google similarity in protein sequence comparison. In *Information Technology, 2008. ITSim 2008. International Symposium on*, volume 1, pages 1–5. IEEE, 2008.

Table 1: Results on Alfree.

Method	Class	Fold	Superfamily	Family	Mean AUC	Comments
NGD	0.63	0.78	0.80	0.84	**0.76**	From Alfree[1]
SW	0.62	0.67	0.78	0.81	0.720	From Alfree[1]
LZW-NCD[2]	0.6282	0.7030	0.7522	0.7753	0.7147	
MMK	0.5929	0.5573	0.6278	0.6325	**0.6026**	
LAK	0.6057	0.5263	0.5957	0.5740	0.5753	no normalization
LZW-Kernel[3]	0.5971	0.5251	0.5972	0.5846	0.5760	
MMK	0.5921	0.5990	0.6712	0.6874	0.6374	
LAK	0.6242	0.7668	0.8013	0.8383	**0.7576**	normalized by
LZW-Kernel	0.5949	0.6170	0.6771	0.6900	0.6447	k^*
SW	0.4847	0.6841	0.7137	0.7940	0.6691	
MMK	0.6012	0.7441	0.7538	0.7935	0.7232	
LAK	0.4457	0.4502	0.4536	0.4827	0.4580	normalized by
LZW-Kernel	0.6043	0.7638	0.7702	0.8155	**0.7385**	$\tilde{k}$
SW	0.6051	0.7601	0.7695	0.8117	0.7366	

Performance is measured in AUC. [1]Data taken from the website of Alfree.
[2]Defined by Eq. (1) in the main article.
[3]Defined as $\tilde{K}_{LZW}(x,y) = \sum_{x_d \in D(x), y_d \in D(y)} k_c(x_d, y_d)$