Briefings in Bioinformatics, 23(1), 2022, 1–10

https://doi.org/10.1093/bib/bbab434
Problem Solving Protocol

MathFeature: feature extraction package for DNA, RNA and protein sequences based on mathematical descriptors

Robson P. Bonidia, Douglas S. Domingues, Danilo S. Sanches and André C.P.L.F. de Carvalho

Corresponding author: Robson P. Bonidia, Institute of Mathematics and Computer Sciences, University of São Paulo, São Carlos 13566-590, Brazil. E-mail: rpbonidia@gmail.com

Abstract

One of the main challenges in applying machine learning algorithms to biological sequence data is how to numerically represent a sequence in a numeric input vector. Feature extraction techniques capable of extracting numerical information from biological sequences have been reported in the literature. However, many of these techniques are not available in existing packages, such as mathematical descriptors. This paper presents a new package, MathFeature, which implements mathematical descriptors able to extract relevant numerical information from biological sequences, i.e. DNA, RNA and proteins (prediction of structural features along the primary sequence of amino acids). MathFeature makes available 20 numerical feature extraction descriptors based on approaches found in the literature, e.g. multiple numeric mappings, genomic signal processing, chaos game theory, entropy and complex networks. MathFeature also allows the extraction of alternative features, complementing the existing packages. To ensure that our descriptors are robust and to assess their relevance, experimental results are presented in nine case studies. According to these results, the features extracted by MathFeature showed high performance (0.6350–0.9897, accuracy), both applying only mathematical descriptors, but also hybridization with well-known descriptors in the literature. Finally, through MathFeature, we overcame several studies in eight benchmark datasets, exemplifying the robustness and viability of the proposed package. MathFeature has advanced in the area by bringing descriptors not available in other packages, as well as allowing non-experts to use feature extraction techniques.

Key words: package; feature extraction; mathematical descriptors; biological sequences; python; GUI-based platform

Robson P. Bonidia received the M.Sc. degree in bioinformatics from the Federal University of Technology - Paraná (UTFPR), Brazil. He is currently pursuing the Ph.D. degree in computer science and computational mathematics with the University of São Paulo-USP. His main research topics are in computational biology and pattern recognition, feature extraction and selection, metaheuristics, and sports data mining.

Douglas S. Domingues graduated in Biology in the São Paulo State University at Botucatu, Brazil, in 2003. He received the PhD degree in Biotechnology from the University of São Paulo, Brazil, in 2009. He is currently a research professor of Plant Gene Expression in the Department of Biodiversity, São Paulo State University at Rio Claro, Brazil, in charge of the Genomics and Transcriptomics in Plants Group. He is the Head of the PhD in Plant Biology in São Paulo State University at Rio Claro, Brazil. In his research, he uses genomics and transcriptomics approaches in non-model plants to understand gene function, the evolution of gene families and genome components, as well as molecular responses to environmental constraints.

Danilo S. Sanches received the Ph.D. degree in electrical engineering from the University of Sao Paulo, in 2013. He is currently an Associate Professor with the Computer Science Department, Federal University of Technology - Paraná (UTFPR), Brazil. His research includes data mining, machine learning, evolutionary algorithms, bioinformatics, and pattern recognition approaches.

André. C. P. L. F. de Carvalho is a full professor at the Department of Computer Science, University of São Paulo. He is the Vice Dean of the Mathematics and Computer Science Institute of University of São Paulo, ICMC-USP, Vice Director of the Center for Mathematical Sciences Applied to Industry, USP and Vice President of the Brazilian Computer Society, SBC. His research interests are in machine learning, data mining and data science.

Submitted: 29 June 2021; **Received (in revised form):** 18 September 2021

Background

Machine learning (ML) algorithms have been successfully applied to genomics, transcriptomics and proteomics problems [1, 2]. Nevertheless, their predictive performance depends on the representation of the sequences by relevant features, able to extract important aspects present in the original sequences. In [3, 4], the authors address the relevance of using an appropriate mathematical expression to extract features from biological data, which has been adopted by several studies [5–7], e.g. non-classical secreted proteins [8], phage virion proteins (PVP)[9], SARS-CoV-2 [10, 11], sigma70 promoters [12] and long non-coding RNAs [13, 14].

As a result, many techniques have been proposed and experimentally investigated [15, 16], and several of them were made available in public software packages, such as PROFEAT [17], PseAAC [18], propy [19], PseKNC-general [16], SPiCE [20], protr/ProtrWeb [21], ProFET [22], Pse-in-One [4], repDNA [23], Rcpi [24], repRNA [25], BioSeq-analysis [26], iFeature [27], PyBioMed [28], Seq2Feature [29], PyFeat [30], iLearn [7], periodicDNA[31] and iLearnPlus [32].

These software packages have been used to extract features from sequences. However, there are some aspects present in the sequences that the feature extraction techniques included in these tools cannot extract. These features, which were shown to be relevant in previous studies [33–36], describe mathematical aspects observed in biological sequences and will be named here mathematical descriptors [37]. These descriptors are based on several techniques, such as multiple numerical mappings, Fourier transform (FT), chaos game theory, entropy and complex networks (CN). To allow the extraction of these descriptors as features for the study of biological sequences, and also including conventional descriptors available in other packages, we created a novel open-source Python package, named MathFeature.

This package provides, in a single environment, many of the mathematical descriptors previously proposed for feature extraction from biological sequences [33–36]. MathFeature contains 37 descriptors, in which, 20 of them are mathematically organized into five groups (numerical mapping, chaos game, FT, entropy and graphs). Additionally, MathFeature extends our preliminary investigation [36], where we investigated nine sets of mathematical features. MathFeature also includes descriptors for Protein sequences, i.e. prediction of structural features along the primary sequence of amino acids. To the best of our knowledge, MathFeature is the first package to provide such a large and comprehensive set of feature extraction techniques based on mathematical descriptors for DNA, RNA and Proteins.

Related works

Fundamentally, we consider feature engineering a key step to ML application success [38–40], mainly in biological sequence preprocessing [3, 41, 42]. In terms of terminology, according to [38], feature is synonymous of an input variable or attribute. Nevertheless, studies also use the 'feature descriptor' terminology (the majority in our review—15 studies), which is the reason why we adopted this term, where a feature descriptor refers to the feature extraction method/technique that can present several measures/values.

In this section, we described 17 studies (cited in Background Section) related to feature extraction packages (tools, web servers, toolkits, etc), providing several feature descriptors for biological sequence analyses. We organized the selected studies into application categories (that is, DNA, RNA, or protein—Supplementary File S1). Furthermore, we also plotted a Venn Diagram (see Supplementary File S2), including all studies by application. In general, most studies are focused on the representation of proteins (eight studies), while DNA and RNA studies had one application each. Moreover, considering the intersection of applications, we found four studies of applications combining DNA, RNA and protein, whereas DNA+protein with two studies and DNA+RNA with one study, respectively.

In our literature review, we found 173 feature descriptors. It is not feasible to individually analyze and describe each descriptor. For this reason, based on our review, we divided these descriptors into 15 large groups, as shown in Table 1. The group column classifies the feature descriptors based on the reviewed studies, and the study column includes packages that have at least one descriptor from the related group.

Considering the groups introduced in Table 1, we realized that most descriptors are based on AAC, PseAAC, CTD and SO for proteins, while NAC and PseNAC descriptors for DNA/RNA, and autocorrelation for DNA, RNA and protein. Nevertheless, MathFeature overcomes other packages in different types of

Table 1. Descriptor groups in reviewed studies

Group	Initials	Application group	Study
Amino acid composition	AAC	Protein	[7] [4] [17] [19] [20] [21] [22] [24] [26] [27] [28] [29] [30]
Pseudo-amino acid composition	PseAAC	Protein	[7] [4] [18] [19] [20] [21] [24] [26] [27] [28]
Composition, transition, distribution	CTD	Protein	[7] [17] [19] [20] [21] [22] [24] [27] [28]
Sequence-order	SO	Protein	[7] [17] [19] [20] [21] [24] [27] [28]
Conjoint triad	CT	Protein	[7] [21] [24] [27] [28]
Proteochemometric descriptors	PCM	Protein	[7] [21] [24] [27]
Profile-based features	PF	Protein	[7] [20] [21] [24] [26] [27]
Nucleic acid composition	NAC	DNA, RNA	[7] [4] [16] [23] [25] [26] [28] [30]
Pseudo nucleic acid composition	PseNAC	DNA, RNA	[7] [4] [16] [23] [25] [26] [28]
Structure composition	SC	DNA, RNA, Protein	[7] [25] [26] [27]
Sequence similarity	SS	DNA, RNA, Protein	[24]
Autocorrelation	–	DNA, RNA, Protein	[7] [17] [19] [16] [20] [21] [4] [23] [24] [26] [27] [28]
Numerical mapping	–	DNA, RNA, Protein	[7] [27]
K-nearest neighbor	KNN	DNA, RNA, Protein	[7] [27]
Physicochemical property	PP	DNA, RNA, Protein	[7] [22] [27] [29]

Table 2. Descriptors calculated by MathFeature compared to the available feature extraction packages. This table shows the number of MathFeature descriptors that existing packages have implemented

Package	Mathematical descriptors	Conventional descriptors	Number of descriptors calculated
MathFeature	20	17	37
PROFEAT	0	2	2
PseAAC	0	2	2
propy	0	5	5
PseKNC-general	0	5	5
SPiCE	0	4	4
ProtrWeb	0	5	5
ProFET	2	3	5
Pse-in-One	0	5	5
repDNA	0	5	5
Rcpi	0	3	3
repRNA	0	5	5
BioSeq-analysis	0	9	9
iFeature	1	4	5
PyBioMed	0	7	7
Seq2Feature	0	0	0
PyFeat	1	8	9
iLearn	2	13	15

mathematical descriptors (e.g. chaos game, FT, entropy and graphs), except two descriptors in numerical mapping, available in only two packages [7, 27]. In addition, to better illustrate the advantages of MathFeature compared with other studies, we included Table 2, which shows the number of MathFeature descriptors that can also be found in other tools. In that case, it can be noticed that only iLearn has 15 descriptors from a total of 37 descriptors available in MathFeature. Moreover, we found only a few sets (2 up to 9) of similar descriptors from other packages compared to our study. Based on this analysis, we realized the novelty of MathFeature for providing different descriptors in biological sequences, which we believe to be an important contribution. Also, most studies (13, 76.47%) were dedicated to evaluating only one type of sequence, while 4 (23.53%) studies cover multiple types of sequences, including MathFeature. Finally, our package is also competitive in terms of the number of descriptors (total of 37).

Package description

MathFeature is a user friendly package that covers 20 mathematical descriptors, as illustrated by Figure 1. We also elaborate the MathFeature execution workflow, which can be divided into four simple steps, as shown in Figure 2. In Table 3, we organized the 20 descriptors into 5 groups (numerical mapping (7), chaos game (2), FT (7), entropy (2) and graphs (2)), according to their structure. MathFeature can be run on console, but we also provide a graphical user interface (GUI)-based platform (see Supplementary File: S3). We briefly describe each of the 5 groups representing the 20 descriptors:

- **Numerical mapping:** Several sequence analysis studies require converting a biological sequence into a numerical sequence. Previous studies [43–45] have proposed descriptors for such, which are able to represent important aspects of these sequences. This group contains 7 descriptors for numerical mapping: Voss [46] (known as binary mapping), Integer [45], real [47], Z-curve [43], electron-ion interaction potential (EIIP) [48, 49], complex Numbers [44, 50] and atomic number [35, 51].
- **FT:** This group consists of feature extraction methods, which generate sequence features based on genomic signal processing (GSP), using FT, a widely applied approach in several biological sequence analysis problems [34–36, 52]. To implement GSP techniques, we used all numerical mappings. A mathematical exploration can be seen in [36].
- **Chaos game representation (CGR):** This approach is also a mapping for a sequence, but scale-independent and iterative for geometric representation of DNA sequences [53]. Based on available CGR representations, the MathFeature package considers classical CGR [34, 53], frequency CGR [54] and CGR signal with FT [34].
- **Entropy:** Different studies have applied concepts from information theory for sequence feature extraction, mainly Shannon's entropy (SE) [33, 55]. According to [56], Tsallis entropy (TE) [57] has been successfully explored in several studies. Moreover, Tsallis entropy attempted to generalize the Boltzmann/Gibbs's traditional entropy. This group includes these two descriptors [36].
- **Graphs:** This group has descriptors based on graph theory (CN), which has been successfully used to represent biological sequence for classification tasks [58, 59]. The descriptors implemented in this group include techniques proposed in [60] and explored in [36].

MathFeature also provides well-known descriptors from other studies with biological sequences (called conventional descriptors here, see Table 4, due to the large number of implementations in the revised packages, see Table 1) such as NAC, dinucleotide composition (DNC), trinucleotide composition (TNC), pseudo K-tuple nucleotide composition (PseKNC) [16], accumulated nucleotide frequency (ANF—DNA, RNA and protein) [61], basic k-mer (DNA, RNA and protein) [62], AAC, dipeptide composition (DPC), tripeptide composition (TPC) and Xmer k-Spaced Ymer composition frequency (kGap - DNA, RNA and protein) [30]. In addition, we also implemented two widely known descriptors in coding sequence studies, e.g. open reading frame (ORF) or coding features [36] and Fickett score [63]. Finally, we summarized the set of features generated by each descriptor investigated in this study (mathematical and conventional), as described in Table 5. MathFeature is freely available at https://github.com/Bonidia/MathFeature, and its documentation is provided at https://bonidia.github.io/MathFeature/.

Results

The main aim of this paper is to make publicly available a large set of feature extraction techniques for biological sequences, including mathematical descriptors not found in similar packages. These descriptors have been successfully applied to extract relevant features from biological sequences, as can be seen in [36], [34], [52], [33] and [60]. For this reason, to assess the relevance of MathFeature descriptors, we provide case studies, which are detailed and presented in the experimental scenario section.

Experimental scenario

We ran experiments for nine case studies with distinct scenarios for the classification of DNA, RNA and protein sequences, as shown in Table 6. These case studies compare the use of several descriptors in distinct problem domains. Furthermore, we did not include any feature selection or hyperparameter

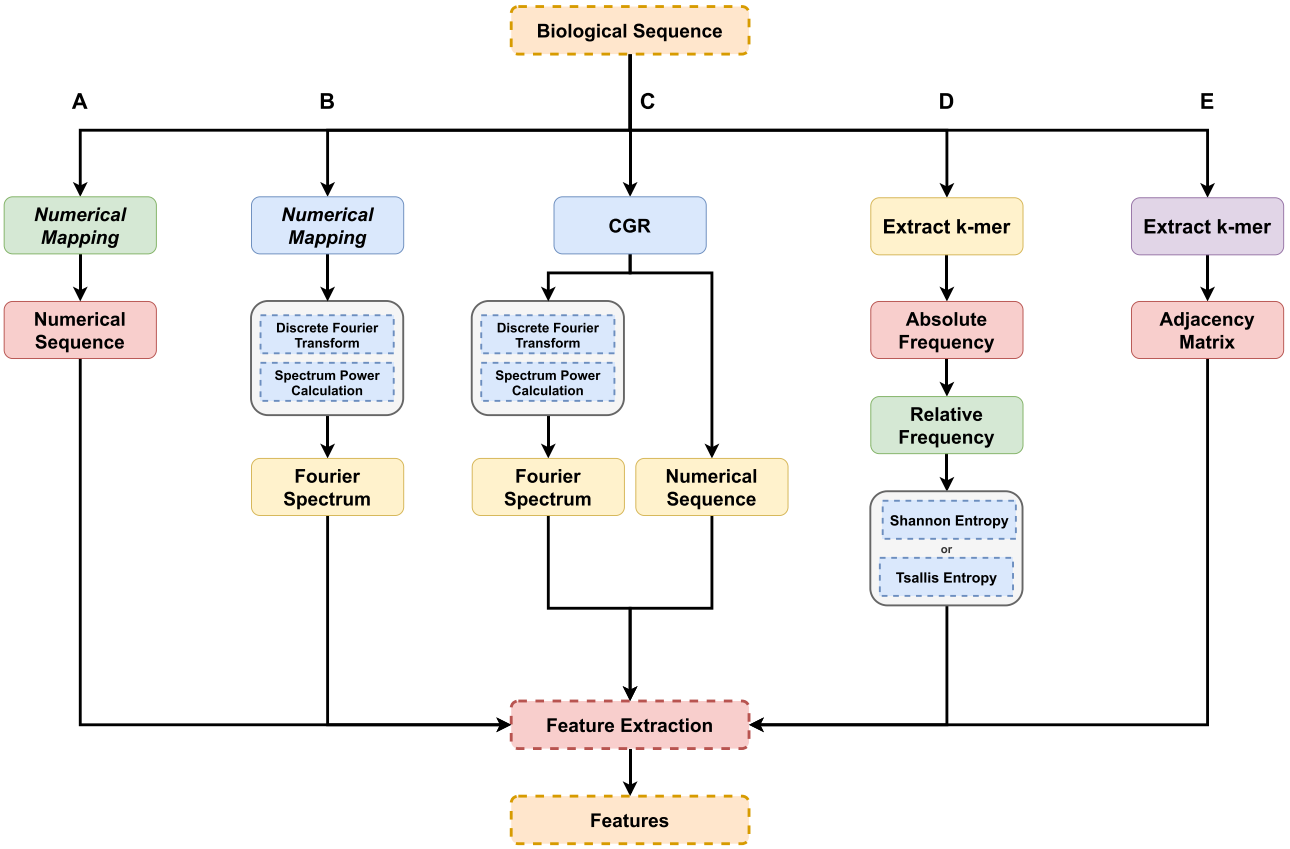

Figure 1. Pipeline of descriptors calculated by MathFeature. **A:** Numerical mapping; **B:** FT; **C:** Chaos game representation; **D:** entropy; **E:** complex networks.

Table 3. Mathematical descriptors calculated by MathFeature for DNA, RNA and Protein sequences

Descriptor groups	Descriptor	Dimension	Biological Sequence
	Binary	$L \cdot 4$	DNA/RNA
	Z-curve	$L \cdot 3$	DNA/RNA
	Real	L	DNA/RNA
Numerical mapping	Integer	L	DNA/RNA/Protein
	EIIP	L	DNA/RNA/Protein
	Complex Number	L	DNA/RNA
	Atomic Number	L	DNA/RNA
	Binary + Fourier	19	DNA/RNA
FT	Z-curve + Fourier	19	DNA/RNA
	Real + Fourier	19	DNA/RNA
	Integer + Fourier	19	DNA/RNA/Protein
	EIIP + Fourier	19	DNA/RNA/Protein
	Complex Number + Fourier	19	DNA/RNA
	Atomic Number + Fourier	19	DNA/RNA
	CGR	$L \cdot 2$	DNA/RNA
Chaos game	Chaos Game Signal (with Fourier)	19	DNA/RNA
entropy	Shannon	k	DNA/RNA/Protein
	Tsallis	k	DNA/RNA/Protein
Graphs	CN (with threshold)	$12 \cdot t$	DNA/RNA/Protein
	CN (without threshold)	$26 \cdot k$	DNA/RNA/Protein

L = length of the longest sequence, k = frequencies of k-mer, t = threshold - number of subgraphs.

optimization technique. Hence, for a fair comparison, we selected descriptors using stratified random sampling (choosing descriptors in each group defined in the article, e.g. numerical mapping, FT, chaos game, entropy, graphs and conventional) in all case studies to avoid any biased choices according to the problem domain. In addition, to compare our results with state-of-the-art studies, we used different ML algorithms, performance measures and dataset partitions to adapt our

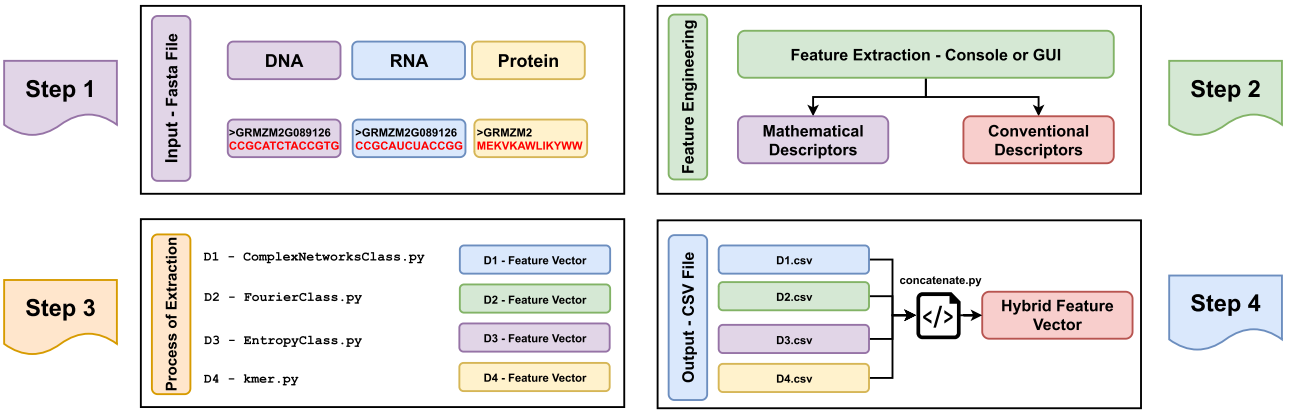

Figure 2. MathFeature execution workflow. **Step 1:** Select input sequence (DNA/RNA/Protein - MathFeature only accepts fasta format); **Step 2:** Choose the descriptor (mathematical or conventional); **Step 3:** It is necessary to run each descriptor separately; **Step 4:** The generated vectors can be used separately or they can be hybridized in a single vector.

Table 4. Conventional descriptors calculated by MathFeature for DNA, RNA and Protein sequences

Descriptor groups	Descriptor	Dimension	Biological sequence
Other descriptors	Basic k-mer	4^k or 20^k	DNA/RNA/Protein
	Customized k-mer	4^k or 20^k	DNA/RNA/Protein
	NAC	4	DNA/RNA
	DNC	16	DNA/RNA
	TNC	64	DNA/RNA
	ORF Features or Coding Features	10	DNA/RNA
	Fickett score	2	DNA/RNA
	PseKNC	–	DNA/RNA
	ANF	L	DNA/RNA/Protein
	kGap	$4^X \cdot 4^Y$ or $20^X \cdot 20^Y$	DNA/RNA/Protein
	AAC	20	Protein
	DPC	400	Protein
	TPC	8000	Protein

L = length of the longest sequence, k = frequencies of k-mer

pipeline to the benchmark dataset. Finally, we also selected hybridized features using stratified random sampling, to assess how these feature sets can improve the ML model prediction.

Case study I-non-classical secreted proteins

Here, we induced a classifier for the non-classical secreted proteins using benchmark datasets provided by [8] (training: 141 positive and 446 negative samples; test: 34 positive and 34 negative objects). We extracted features using integer mapping, FT + integer mapping and AAC. Afterward, we applied the CatBoost algorithm to the new datasets and assessed the predictive performance using Accuracy (ACC), F1-score and Matthews Correlation Coefficient (MCC). Our performance (ACC: 0.8382, F1-score: 0.8070 and MCC: 0.7149) was superior to state-of-the-art tools, such as SecretomeP [70] (ACC: 0.5880, F1-score: 0.4620 and MCC: 0.2000) and PeNGaRoo [8] (ACC: 0.7790, F1-score: 0.7890 and MCC: 0.5610).

Case study II-PVP

This study, considering the prediction of PVP, is reported in [9]. For the experiments carried out, we used benchmark data provided by [64], with 500 sequences for training (250 PVP and 250 non-PVP) and 126 for tests (63 PVP and 63 non-PVP). To numerically represent the sequences, we built a hybrid feature set with SE (k = 12), CN (k = 1, t = 2) and AAC. To generate our predictive model, a classifier was induced using an ensemble method (bagging) of Support Vector Machines (SVMs), assessing its predictive performance with the F1-score, ACC, area under the curve (AUC) and MCC. Experimental results showed high performance for F1-score: 0.7934, ACC: 0.8016, AUC: 0.8661 and MCC: 0.6051. The results using the hybrid set of features were superior to the performance obtained using conventional features extracted from the same dataset [64]. Using the hybrid feature set also improved the predictive performance, when compared with the feature set used by PVPred [71] (ACC: 0.7300, AUC: 0.8570 and MCC: 0.5050), PVP-SVM [9] (ACC: 0.7460, AUC: 0.8440 and MCC: 0.5050) and PVPred-SCM [72] (ACC: 0.7140, AUC: – and MCC: 0.4320) and slightly worse than Meta-iPVP [64] (ACC: 0.8170, AUC: 0.8700 and MCC: 0.6420).

Case study III-SARS-CoV-2 sequences

For this case study, we conducted experiments using a dataset to differentiate SARS-CoV-2 from other viruses (e.g. HIV, Influenza, hepatitis, Ebolavirus, SARS). We downloaded all available virus sequences (29 135) from the NCBI Viral Genome database [65] (complete genomic sequences (DNA), e.g. Nucleotide Completeness = 'complete' AND host = 'homo sapiens'). In a preprocessing phase, we removed sequences smaller than 2000bp and larger than 50 000 bp [73] to eliminate any bias in the sequence size,

Table 5. Features generated by each mathematical and conventional descriptor calculated by MathFeature

Descriptors	Features
Binary, Z-curve, Real, Integer, EIIP, complex number, atomic number, CGR, ANF	Convert a biological sequence into a numerical sequence, e.g. Integer representation: GAGAGTGACCA == 3, 2, 3, 2, 3, 0, 3, 2, 1, 1, 2.
Binary + Fourier, Z-curve + Fourier, real + Fourier, integer + Fourier, EIIP + Fourier, complex number + Fourier, atomic number + Fourier, Chaos Game Signal (with Fourier)	Peak to average power ratio (2 features), average power spectrum, median, maximum, minimum, sample SD, population SD, percentile (15/25/50/75), range, variance, interquartile range, semi-interquartile range, coefficient of variation (cv), skewness and kurtosis.
Shannon, Tsallis	For each k-mer (e.g. 1-mer, 2-mers,..., k-mers), we generated an entropic measure.
CN (with threshold)	Betweenness, assortativity, average degree, average path length, minimum degree, maximum degree, number of edges, degree SD, frequency of motifs (size 3 and 4), clustering coefficient (local and global).
CN (without threshold)	Betweenness, assortativity, average degree, average path length, minimum degree, maximum degree, number of edges, degree SD, frequency of motifs (size 3 and 4), clustering coefficient (local and global), Kleinberg's authority centrality scores, closeness centralities, Burt's constraint scores, multiplicities, density, diameter, eccentricity, edge betweenness, Kleinberg's hub score, maximum degree of a vertex set, neighborhood size, radius, strength (weighted degree), number of vertices.
k-mer, Customized k-mer, NAC, DNC, TNC, AAC, DPC, TPC, kGap	Generation of nucleic acid or amino acid statistical information, e.g. NAC for DNA: relative frequency of A, C, T, G.
ORF features or coding features	Maximum ORF length, minimum ORF length, std ORF length, average ORF length, cv ORF length, maximum GC content - ORF, minimum GC content - ORF, std GC content - ORF, average GC content - ORF, cv GC content - ORF.
Fickett score	Fickett:orf, Fickett:full:sequence
PseKNC	Modes of PseKNC with physicochemical properties

Table 6. Experimental scenario in nine case studies

Problem	Reference	Case study	Application	Number of sequences	Classifier
Non-classical secreted proteins	[8]	I	Protein	655	CatBoost
PVP	[64]	II	Protein	626	Support Vector Machines
SARS-CoV-2 sequences	[65]	III	DNA	24 815	Random Forest
Sigma70 promoters	[12]	IV	DNA	2141	Support Vector Machines
Anticancer Peptides	[66]	V	Protein	344	Random Forest
Protein lysine crotonylation	[67]	VI	Protein	40 587	Random Forest
Long non-coding RNAs	[13]	VII	RNA	21 000 and 12 000	CatBoost
Long non-coding RNAs	[68]	VIII	RNA	36 000	Deep Learning
Sigma70 promoters	[69]	IX	DNA	2141	Random Forest

since SARS-CoV-2 has an average length of 29 838 bp, resulting in a dataset with 22 442 and 2373 sequences from other viruses and SARS-CoV-2, respectively. In this experiment, we extracted the TE-based features ($k = 12$ and $q = 6$). We applied the Random Forest (RF) algorithm to the dataset represented by TE-based features, using 10-fold cross-validation (mean). It is important to note that we continued with an unbalanced dataset, keeping performance metrics (e.g. F1-score, balanced accuracy (BACC), and also including Cohen's kappa coefficient). In the experimental results, the predictive performance of the RF model to discriminate SARS-CoV-2 from several other viruses with F1-score, BACC and kappa of 0.9873, 0.9919, 0.9860, respectively. Moreover, we tested other conventional descriptors (e.g. k-mer, PseKNC, ORF features, Fickett score and TNC). These descriptors performed between (0.9800-0.9900, balanced accuracy), and hence, we carried out the classification task between SARS-CoV-2 and other viruses, which are linearly separable even using different feature vectors. In addition, these results are supported by [10, 11].

Case study IV-Sigma70 promoters

In this case study, we trained a SVM classifier to induce a sigma70 promoter predictor based on the benchmark dataset from [12]. This dataset contains 741 positive samples (promoter) and 1400 negative samples (non-promoter). For the feature extraction, we used the CGR descriptor. The experiments were assessed partitioning the dataset with 5-fold cross-validation (same as in [12]), when the following mean performance values were obtained: 0.8594, 0.8346, 0.7872 and 0.6852 for ACC, BACC, F1-score and MCC, respectively. In [12], the authors report the performance of their tool, iPro70-PseZNC, also using SVM, for 2 of these metrics, ACC: 0.8450 and MCC: 0.6630. Thus, by using the mathematical descriptors, the results improved by 0.0144 (1.44%), for ACC and 0.0222 (2.22%), for MCC.

Case study V-anticancer peptides

In this case study, our aim is to identify anticancer peptides based on [66]. For such, we extracted features CN ($k = 2$, $t = 1$)

and AAC from the benchmark dataset provided by the authors (206 non-anticancer peptides and 138 anticancer peptides). The RF algorithm was applied to the transformed dataset using 10-fold cross-validation. The mean predictive performance of the trained model was assessed using ACC, F1-score and MCC. The performance of this model was superior to the performance reported in [66], (ACC: 0.9300, F1-score: 0.9061 and MCC: 0.8563 against ACC: 0.9273, F1-score: 0.9270 and MCC: 0.8490).

Case study VI-protein lysine crotonylation

Based on [67], we induced and assessed the RF algorithm to identify protein lysine crotonylation sites. The benchmark data provided by the author contains 32 418 sequences for training (2742 positive and 29 676 negative peptides - papaya) and 8169 sequences for tests (711 positive and 7458 negative peptides - papaya). For feature extraction, we applied numerical mapping with EIIP. We assessed the predictive performance with BACC and MCC, which were 0.6450 and 0.1652, respectively. These results were better than those obtained with the some feature extraction techniques used in [67], e.g. RF_{AAC} (MCC: 0.1030) and RF_{CKSAAP} (MCC: 0.1110).

Case study VII-long non-coding RNAs

In this case study, we trained the CatBoost algorithm to classify long non-coding RNAs (lncRNAs) sequences from protein-coding genes (mRNAs), using two datasets made available by [13]: Human (training set: 16 000 sequences and test set: 5000 sequences) and Wheat (training set: 8000 sequences and test set: 4000 sequences). From these datasets, we extracted the FT + real mapping, TNC and coding descriptors. Essentially, we followed the same pipeline of previous case studies. Once again, the predictive model induced using our descriptors showed a high predictive performance in the datasets, e.g. Human (ACC: 0.9652, F1-score: 0.9646, MCC: 0.9309) and Wheat (ACC: 0.8870, F1-score: 0.8907, MCC: 0.7757). Our results were better than several tools shown in [13], e.g. CPC [74] (Human - ACC: 0.8304; Wheat - ACC: 0.9595), CNCI [75] (Human - ACC: 0.9450; Wheat - ACC: 0.6158), CPAT [63] (Human–ACC: 0.9642; Wheat–ACC: 0.8743), PLEK [76] (Human–ACC: 0.9274; Wheat–ACC: 0.8773), and CPC2 [77] (Human–ACC: 0.9614; Wheat–ACC: 0.7870).

Case study VIII-using MathFeature with deep learning

According to [78], deep learning (DL) is a field of ML responsible for several advances, due to its high predictive performance in big data [79]. Therefore, we assess our descriptors with a DL architecture, using the same case study problem VII [lncRNAs versus mRNAs - feature vector (FT + real mapping and coding descriptors)], but with a benchmark dataset from [68] (*Zea mays* dataset (36 000 sequences: 18 000 lncRNA and 18 000 mRNA), whose article is dedicated to a DL approach. Our classifier was generated using Keras [80] (default parameters). Furthermore, we compared our model with three DL tools used in [68] (PlncRNA-HDeep [68], lncRNAnet [81] and LncADeep [82]), using the same pipeline (hold-out (80% of samples for training and 20% for testing), ACC, Recall and F1-score). Our model showed a high predictive performance in the dataset, e.g. ACC: 0.9605, Recall: 0.9917 and F1-score: 0.9616, overcoming lncRNAnet (ACC: 0.7290, Recall: 0.7200, F1-score: 0.7260), LncADeep (ACC: 0.8000, Recall: 0.6660, F1-score: 0.7690) and PlncRNA-HDeep (Recall: 0.9790), but with a small decimal loss in relation (ACC: 0.0045 and F1-score: 0.0034) to PlncRNA-HDeep (ACC: 0.9650 and F1-score: 0.9650). Therefore, based on our results, MathFeature can also generate robust and efficient feature vectors for DL approaches.

Case study IX-MathFeature versus other packages

So far, we have evaluated MathFeature with eight experiments in well-established problems. Nevertheless, in this last case study, we also compared MathFeature with five packages, e.g. BioSeq-Analysis [26], Seq2Feature [29], PyFeat [30], iLearn [7] and SubFeat [69]. The experiments were carried out using the dataset provided by [69], which was the same dataset used in case study IV (Sigma70 Promoters). For this study, we considered 741 positive samples (promoter) and 1400 negative samples (non-promoter) and three metrics (ACC, AUC, MCC), evaluating the RF classifier using 10-fold cross-validation (as our reference). We kept our CGR descriptor. MathFeature (ACC: 0.8576, AUC: 0.9252 and MCC: 0.6797) outperformed all packages, BioSeq-Analysis (ACC: 0.7637, AUC: 0.8297 and MCC: 0.4726), Seq2Feature (ACC: 0.7197, AUC: 0.7637 and MCC: 0.3723), PyFeat (ACC: 0.7842, AUC: 0.8589 and MCC: 0.5064), iLearn (ACC: 0.7597, AUC: 0.8173 and MCC: 0.5275) and SubFeat (ACC: 0.8098, AUC: 0.9232 and MCC: 0.5664). Moreover, based on the results obtained comparing MathFeature and Seq2Feature, we generated a hybrid vector with features from both packages (MathFeature: CGR and Seq2Feature: Nucleotide content, random choice), which provided the best result (ACC: 0.8627, AUC: 0.9332 and MCC: 0.6927). Therefore, we achieved a high predictive performance, applying only MathFeature or a hybrid combination of packages.

Discussion

We assessed the MathFeature package in nine case studies grouped by protein and DNA/RNA sequences. We considered four protein problems and three DNA/RNA problems in the experiments. The classification problems in each case were chosen based on recent articles with distinct domains. For example, for protein molecules, we used the following datasets: (i) non-classical secreted proteins, that according to [8], are important for understanding pathogenesis mechanisms of Gram-positive bacteria; (ii) The PVP identification, e.g. to develop new antibacterial drugs [9]; (iii) anticancer peptides that present a new direction in the treatment of cancer [66, 83] and (4) protein lysine crotonylation, a type of post-translational modification [67, 84]. In these studies, we noticed that the hybrid combination of mathematical and conventional descriptors (available at MathFeature) improves the performance of the models, mainly applying CN, FT, numerical mapping (e.g. EIIP and integer) and AAC, varying the ACC/BACC of 0.6450–0.9300 in all problems. For DNA/RNA molecules, the problems used are (i) SARS-CoV-2, hot topic in bioinformatics [10, 11]; (ii) detection of sigma70 promoters to study the dynamics of gene expression [12, 85]; and (iii) lncRNA sequences, that can play essential roles in biological processes, e.g. transcriptional regulation [68, 86]. For these problems, we obtained highly robust results (varying the ACC/BACC of 0.8594-0.9900), both applying only mathematical descriptors or a hybrid combination, highlighting TE-based features, CGR, FT, TNC and coding descriptors. Finally, our findings report the relevance of MathFeature descriptors in several applications, e.g. humans, plants and bacteria data.

Conclusion

In this study, we described a new package, called MathFeature, comprising an extensive and comprehensive set of 37 feature descriptors for biological sequences. From these 37 descriptors, 20 are based on mathematical approaches and are not available in other feature extraction packages. Seventeen other descriptors, called conventional descriptors, were selected from those often used in the literature. The main motivation for this new package was that, despite the relevance of the features extracted by mathematical descriptors, they are not available in current packages. Thus, MathFeature extends the existing packages, including mathematical techniques. To experimentally assess the descriptors implemented in this package, we conducted nine case studies, using several biological scenarios, e.g. DNA, RNA and Proteins (primary sequence of amino acids), applied in different problem domains. Furthermore, we avoided including any type of bias from selected features, and hence, the quality assessment of each feature can be made by the community with regards to the specific problem of interest. In the experiments, we obtained high predictive performance, both applying only mathematical descriptors (e.g. case studies II, III, VI) and applying a hybrid combination of them with well-known conventional descriptors found in the literature (e.g. AAC, TNC, Coding). Finally, through MathFeature, we outperformed several studies in benchmark datasets, indicating that all descriptors within MathFeature can improve the performance of predictive models induced by ML algorithms. Regarding the limitations, we observed that some of these descriptors (e.g. Fourier, Shannon and Tsallis) have a low performance for short sequences. However, when mathematical descriptors are combined with conventional ones, in hybrid sets, there is a clear improvement in the predictive performance. Finally, as future work, we intend to investigate descriptors for short sequences, especially in prokaryotic organisms, and also include more protein descriptors.

Key Points

- A novel open-source Python package, called MathFeature.
- MathFeature provides 37 descriptors, 20 of them are mathematical, organized into five categories.
- MathFeature can be run on the console, but also provide a GUI-based platform.
- MathFeature is an extensive and comprehensive set of feature extraction techniques based on mathematical descriptors for encoding DNA, RNA and Proteins (primary sequence of amino acids) sequences.
- MathFeature is the first package to provide a large set of features based on mathematical descriptors and also well-known descriptors from other studies with biological sequences.

Supplementary data

Supplementary data are available online at *Briefings in Bioinformatics*.

Acknowledgments

The authors would like to thank USP, CAPES, CNPq and FAPESP (2013/07375-0) for the financial support for this research.

Availability of data and materials

The datasets, experiments and descriptors are available in the Github repository: https://github.com/Bonidia/MathFeature.

Financial support

This project was supported by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) - Finance Code 001 and PROEX-11919694/D, USP, CNPq and FAPESP (2013/07375-0).

Availability and implementation

MathFeature is freely available at https://github.com/Bonidia/MathFeature **Documentation:** https://bonidia.github.io/MathFeature/

References

1. daSilva Diniz WJ, Canduri F. Bioinformatics: an overview and its applications. *Genet Mol Res* 2017; **16**(1).
2. deSouza KP, Setubal JC, deLeon ACP, *et al*. Machine learning meets genome assembly. *Brief Bioinform* 2018; **20**(6): 2116–29.
3. Chou K-C. Some remarks on protein attribute prediction and pseudo amino acid composition. *J Theor Biol* 2011; **273**(1): 236–47.
4. Liu B, Liu F, Wang X, *et al*. Pse-in-One: a web server for generating various modes of pseudo components of DNA, RNA, and protein sequences. *Nucleic Acids Res* 2015; **43**(W1): W65–71.
5. Bonidia RP, Sampaio LDH, Lopes FM, *et al*. Feature extraction of long non-coding rnas: A fourier and numerical mapping approach. In: Nyström I, Heredia YH, Núñez VM (eds). *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*. Springer International Publishing, 2019, 469–79 Cham.
6. Liu B, Gao X, Zhang H. BioSeq-Analysis2.0: an updated platform for analyzing DNA, RNA and protein sequences at sequence level and residue level based on machine learning approaches. *Nucleic Acids Res* 2019; **47**(20): e127–7.
7. Chen Z, Zhao P, Li F, *et al*. iLearn: an integrated platform and meta-learner for feature engineering, machine-learning analysis and modeling of DNA, RNA and protein sequence data. *Brief Bioinform* 2019; **21**(3): 1047–57.
8. Zhang Y, Yu S, Xie R, *et al*. Pengaroo, a combined gradient boosting and ensemble learning framework for predicting non-classical secreted proteins. *Bioinformatics* 2020; **36**(3): 704–12.
9. Manavalan B, Shin TH, Lee G. Pvp-svm: sequence-based prediction of phage virion proteins using a support vector machine. *Front Microbiol* 2018; **9**:476.
10. Naeem SM, Mabrouk MS, Marzouk SY, *et al*. A diagnostic genomic signal processing (GSP)-based system for automatic feature analysis and detection of COVID-19. *Brief Bioinform* 2020; **22**(2): 1197–205.
11. Arslan H. Machine learning methods for covid-19 prediction using human genomic data. *Proceedings* 2021; **74**(1).
12. Lin H, Liang Z-Y, Tang H, *et al*. Identifying sigma70 promoters with novel pseudo nucleotide composition. *IEEE/ACM Trans Comput Biol Bioinform* 2017; **16**(4): 1316–21.
13. Han S, Liang Y, Ma Q, *et al*. Lncfinder: an integrated platform for long non-coding rna identification utilizing sequence

intrinsic composition, structural information and physicochemical property. *Brief Bioinform* 2018.
14. Bonidia RP, Machida JS, Negri TC, *et al*. A novel decomposing model with evolutionary algorithms for feature selection in long non-coding rnas. *IEEE Access* 2020; **8**:181683–97.
15. Chen W, Lei T-Y, Jin D-C, *et al*. Pseknc: A flexible web server for generating pseudo k-tuple nucleotide composition. *Anal Biochem* 2014; **456**:53–60.
16. Chen W, Zhang X, Brooker J, *et al*. PseKNC-General: a cross-platform package for generating various modes of pseudo nucleotide compositions. *Bioinformatics* 2014; **31**(1): 119–20.
17. Li ZR, Lin HH, Han LY, *et al*. PROFEAT: a web server for computing structural and physicochemical features of proteins and peptides from amino acid sequence. *Nucleic Acids Res* 2006; **34**:W32–7.
18. Shen H-B, Chou K-C. Pseaac: A flexible web server for generating various kinds of protein pseudo amino acid composition. *Anal Biochem* 2008; **373**(2): 386–8.
19. Cao D-S, Xu Q-S, Liang Y-Z. propy: a tool to generate various modes of Chou's PseAAC. *Bioinformatics* 2013; **29**(7): 960–2.
20. van denBerg BA, Reinders MJT, Roubos JA, *et al*. Spice: a web-based tool for sequence-based protein classification and exploration. *BMC bioinformatics* 2014; **15**(1): 93.
21. Xiao N, Cao D-S, Zhu M-F, *et al*. protr/ProtrWeb: R package and web server for generating various numerical representation schemes of protein sequences. *Bioinformatics* 2015; **31**(11): 1857–9.
22. Ofer D, Linial M. ProFET: Feature engineering captures high-level protein functions. *Bioinformatics* 2015; **31**(21): 3429–36.
23. Liu B, Liu F, Fang L, *et al*. repDNA: a Python package to generate various modes of feature vectors for DNA sequences by incorporating user-defined physicochemical properties and sequence-order effects. *Bioinformatics* 2014; **31**(8): 1307–9.
24. Chiu T-P, Comoglio F, Zhou T, *et al*. DNAshapeR: an R/Bioconductor package for DNA shape prediction and feature encoding. *Bioinformatics* 2015; **32**(8): 1211–3.
25. Liu B, Liu F, Fang L, *et al*. reprna: a web server for generating various feature vectors of rna sequences. *Mol Genet Genomics* 2016; **291**(1): 473–81.
26. Liu B. Bioseq-analysis: a platform for dna, rna and protein sequence analysis based on machine learning approaches. *Brief Bioinform* 2017; **20**(4): 1280–94.
27. Chen Z, Zhao P, Li F, *et al*. iFeature: a Python package and web server for features extraction and selection from protein and peptide sequences. *Bioinformatics* 2018; **34**(14): 2499–502.
28. Dong J, Yao Z-J, Zhang L, *et al*. Pybiomed: a python library for various molecular representations of chemicals, proteins and dnas and their interactions. *J Chem* 2018; **10**(1).
29. Nikam R, Gromiha MM. Seq2Feature: a comprehensive web-based feature extraction tool. *Bioinformatics* 2019; **35**(22): 4797–9.
30. Muhammod R, Ahmed S, Farid DM, *et al*. PyFeat: a Python-based effective feature generation tool for DNA, RNA and protein sequences. *Bioinformatics* 2019; **35**(19): 3831–3.
31. Serizay J, Ahringer J. periodicdna: an r/bioconductor package to investigate k-mer periodicity in dna. *F1000Research* 2021.
32. Chen Z, Zhao P, Li C, *et al*. iLearnPlus: a comprehensive and automated machine-learning platform for nucleic acid and protein sequence analysis, prediction and visualization. *Nucleic Acids Res* 2021;gkab122.
33. Machado JAT, Costa AC, Quelhas MD. Shannon, rényie and tsallis entropy analysis of dna using phase plane. *Nonlinear Analysis: Real World Applications* 2011; **12**(6): 3135–44.
34. Hoang T, Yin C, Yau SS-T. Numerical encoding of dna sequences by chaos game representation with application in similarity comparison. *Genomics* 2016; **108**(3–4): 134–42.
35. Mendizabal-Ruiz G, Román-Godínez I, Torres-Ramos S, *et al*. On dna numerical representations for genomic similarity computation. *PloS one* 2017; **12**(3):e0173288.
36. Bonidia RP, Sampaio LDH, Domingues DS, *et al*. Feature extraction approaches for biological sequences: a comparative study of mathematical features. *Brief Bioinform* 2021;bbab011.
37. Nguyen DD, Cang Z, Wei G-W. A review of mathematical representations of biomolecular data. *Phys Chem Chem Phys* 2020; **22**(8): 4343–67.
38. Guyon I, Gunn S, Nikravesh M, *et al*. *Feature extraction: foundations and applications*, Vol. **207**. Springer, 2008.
39. Vishnoi S, Garg P, Arora P. Physicochemical n-grams tool: A tool for protein physicochemical descriptor generation via chou's 5-step rule. *Chem Biol Drug Des* 2020; **95**(1): 79–86.
40. Ghannam RB, Techtmann SM. Machine learning applications in microbial ecology, human microbiome studies, and environmental monitoring. *Comput Struct Biotechnol J* 2021.
41. Saidi R, Aridhi S, Nguifo EM, *et al*. Feature extraction in protein sequences classification: a new stability measure. In: *Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Biomedicine*. ACM, 2012, 683–9.
42. Zhang Z-Y, Yang Y-H, Ding H, *et al*. Design powerful predictor for mrna subcellular location prediction in homo sapiens. *Brief Bioinform* 2021; **22**(1): 526–35.
43. Zhang R, Zhang C-T. Z curves, an intutive tool for visualizing and analyzing the dna sequences. *Journal of Biomolecular Structure and Dynamics* 1994; **11**(4): 767–82.
44. Anastassiou D. Genomic signal processing. *IEEE Signal Processing Magazine* 2001; **18**(4): 8–20.
45. Cristea PD. Conversion of nucleotides sequences into genomic signals. *J Cell Mol Med* 2002; **6**(2): 279–303.
46. Richard F Voss. Evolution of long-range fractal correlations and 1/f noise in dna base sequences. *Phys Rev Lett*, **68**(25): 3805, 1992.
47. Chakravarthy N, Spanias A, Iasemidis LD, *et al*. Autoregressive modeling and feature analysis of dna sequences. *EURASIP Journal on Applied Signal Processing* 2004; **13–28**:2004.
48. Nair AS, Sreenadhan SP. A coding measure scheme employing electron-ion interaction pseudopotential (eiip). *Bioinformation* 2006; **1**(6): 197.
49. Bloch KM, Arce GR. Analyzing protein sequences using signal analysis techniques. In: *Computational and Statistical Approaches to Genomics*. Springer, 2006, 137–61.
50. Yu N, Li Z, Yu Z. Survey on encoding schemes for genomic data representation and feature learning–from signal processing to machine learning. *Big Data Mining and Analytics* 2018; **1**(3): 191–210.
51. Holden T, Subramaniam R, Sullivan R, *et al*. Atcg nucleotide fluctuation of deinococcus radiodurans radiation genes. In: *Instruments, Methods, and Missions for Astrobiology X*, Vol. **6694**. International Society for Optics and Photonics, 2007, 669417.
52. Yin C, Chen Y, Yau SS-T. A measure of dna sequence similarity by fourier transform with applications on hierarchical clustering. *J Theor Biol* 2014; **359**:18–28.
53. Joel H. Jeffrey, Chaos game representation of gene structure. *Nucleic Acids Res* 1990; **18**(8): 2163–70.

54. Almeida JS, Carrico JA, Maretzek A, *et al*. Analysis of genomic sequences by chaos game representation. *Bioinformatics* 2001; **17**(5): 429–37.
55. Akhter S, Bailey BA, Salamon P, *et al*. Applying shannon's information theory to bacterial and phage genomes and metagenomes. *Sci Rep* 2013; **3**:1033.
56. Yamano T. Information theory based on nonadditive information content. *Physical Review E* 2001; **63**(4): 046105.
57. Tsallis C, Mendes RS, Plastino AR. The role of constraints within generalized nonextensive statistics. *Physica A: Statistical Mechanics and its Applications* 1998; **261**(3–4): 534–54.
58. Pavlopoulos GA, Secrier M, Moschopoulos CN, *et al*. Using graph theory to analyze biological networks. *BioData Min* 2011; **4**(1).
59. Aittokallio T, Schwikowski B. Graph-based methods for analysing networks in cell biology. *Brief Bioinformatics* 2006; **7**(3): 243–55.
60. Ito EA, Katahira I, daRocha Vicente FF, *et al*. Basinet–biological sequences network: a case study on coding and non-coding rnas identification. *Nucleic Acids Res* 2018.
61. Narayan P, Ludwiczak RL, Goodwin EC, *et al*. Context effects on n 6-adenosine methylation sites in prolactin mrna. *Nucleic Acids Res* 1994; **22**(3): 419–26.
62. Mapleson D, Accinelli GG, Kettleborough G, *et al*. KAT: a K-mer analysis toolkit to quality control NGS datasets and genome assemblies. *Bioinformatics* 2016; **33**(4): 574–6.
63. Wang L, Park HJ, Dasari S, *et al*. Cpat: Coding-potential assessment tool using an alignment-free logistic regression model. *Nucleic Acids Res* 2013; **41**(6): e74–4.
64. Charoenkwan P, Nantasenamat C, Hasan MM, *et al*. Meta-ipvp: a sequence-based meta-predictor for improving the prediction of phage virion proteins using effective feature representation. *J Comput Aided Mol Des* 2020; **34**(10): 1105–16.
65. Hatcher EL, Zhdanov SA, Bao Y, *et al*. Virus Variation Resource – improved response to emergent viral outbreaks. *Nucleic Acids Res* 2016; **45**(D1): D482–90.
66. Li Q, Zhou W, Wang D, *et al*. Prediction of anticancer peptides using a low-dimensional feature model. *Front Bioeng Biotechnol* 2020; **8**:892.
67. Zhao Y, He N, Chen Z, *et al*. Identification of protein lysine crotonylation sites by a deep learning framework with convolutional neural networks. *IEEE Access* 2020; **8**:14244–52.
68. Meng J, Kang Q, Zheng C, *et al*. Plncrna-hdeep: plant long noncoding rna prediction using hybrid deep learning based on two encoding styles. *BMC bioinformatics* 2021; **22**(3): 1–16.
69. Haque HMF, Rafsanjani M, Arifin F, *et al*. Subfeat: Feature subspacing ensemble classifier for function prediction of dna, rna and protein sequences. *Comput Biol Chem* 2021; **92**:107489.
70. Bendtsen JD, Kiemer L, Fausbøll A, *et al*. Non-classical protein secretion in bacteria. *BMC Microbiol* 2005; **5**(1): 1–13.
71. Ding H, Feng P-M, Chen W, *et al*. Identification of bacteriophage virion proteins by the anova feature selection and analysis. *Mol Biosyst* 2014; **10**(8): 2229–35.
72. Charoenkwan P, Kanthawong S, Schaduangrat N, *et al*. Pvpred-scm: improved prediction and analysis of phage virion proteins using a scoring card method. *Cell* 2020; **9**(2): 353.
73. Randhawa GS, Soltysiak MPM, Roz, *et al*. Machine learning using intrinsic genomic signatures for rapid classification of novel pathogens: Covid-19 case study. *Plos one* 2020; **15**(4):e0232391.
74. Kong L, Zhang Y, Ye Z-Q, *et al*. Cpc: assess the protein-coding potential of transcripts using sequence features and support vector machine. *Nucleic Acids Res* 2007; **35**(suppl_2): W345–9.
75. Liang S, Luo H, Dechao B, *et al*. Utilizing sequence intrinsic composition to classify protein-coding and long non-coding transcripts. *Nucleic Acids Res* 2013; **41**(17): e166–6.
76. Li A, Zhang J, Zhou Z. Plek: a tool for predicting long non-coding rnas and messenger rnas based on an improved k-mer scheme. *BMC bioinformatics* 2014; **15**(1): 311.
77. Kang Y-J, Yang D-C, Kong L, *et al*. Cpc2: a fast and accurate coding potential calculator based on sequence intrinsic features. *Nucleic Acids Res* 2017; **45**(W1): W12–6.
78. Min S, Lee B, Yoon S. Deep learning in bioinformatics. *Brief Bioinform* 2017; **18**(5): 851–69.
79. Tang B, Pan Z, Yin K, *et al*. Recent advances of deep learning in bioinformatics and computational biology. *Front Genet* 2019; **10**:214.
80. F Chollet. Keras: https://keras. io, 2015.
81. Baek J, Lee B, Kwon S, *et al*. lncrnanet: Long non-coding rna identification using deep learning. *Bioinformatics* 2018; **1**:9.
82. Cheng Y, Yang L, Zhou M, *et al*. Lncadeep: An ab initio lncrna identification and functional annotation tool based on deep learning. *Bioinformatics* 2018.
83. Chen W, Ding H, Feng P, *et al*. iacp: a sequence-based tool for identifying anticancer peptides. *Oncotarget* 2016; **7**(13): 16895.
84. Wang R, Wang Z, Wang H, *et al*. Characterization and identification of lysine crotonylation sites based on machine learning method on both plant and mammalian. *Sci Rep* 2020; **10**(1): 1–12.
85. Cassiano MHA, Silva-Rocha R. Benchmarking bacterial promoter prediction tools: Potentialities and limitations. *Msystems* 2020; **5**(4): e00439–20.
86. Pisignano G, Ladomery M. Post-transcriptional regulation through long non-coding rnas (lncrnas). *Non-Coding RNA* 2021; **7**(2).