PeptiDesCalculator: A Computational Platform for Peptide Descriptor Calculation, Bioactivity Prediction, and Multi-Endpoint Modeling

Dr. Michael Johnson[1]*, Dr. Emily Carter[1], Dr. David Williams[2], Dr. Sarah Thompson[2], Dr. James Miller[3], Dr. Laura Bennett[3]
[1]Harvard Medical School, Boston, USA
[2]Johns Hopkins University, Baltimore, USA
[3]Stanford University School of Medicine, Stanford, USA

ABSTRACT

We present a novel Java-based program denominated PeptiDesCalculator for computing peptide descriptors. These descriptors include: redefinitions of known protein parameters to suite the peptide domain, generalization schemes for the global descriptions of peptide characteristics, as well as empirical descriptors based on experimental evidence on peptide stability and interaction propensity. The PeptiDesCalculator software provides a user-friendly Graphical User Interface (GUI) and is parallelized to maximize the use of computational resources available in current work stations. The PeptiDesCalculator indices are employed in modeling 8 peptide bioactivity endpoints demonstrating satisfactory behavior. Moreover, we compare the performance of a support vector machine (SVM) classifier built using 15 PeptiDesCalculator indices with that of a recently reported deep neural network (DNN) antimicrobial activity classifier, demonstrating comparable test set performance notwithstanding the remarkably lower degree of freedom for the former. This software will facilitate the development of *in silico* models for the prediction of peptide properties.

Key words: Peptide, PeptiDesCalculator, Antimicrobial, Machine Learning

1. INTRODUCTION

Over the past two decades, peptide drug discovery (PDD) has experienced renewed interest and momentum, thanks to the greater appreciation of the possible utility of peptides in addressing unmet clinical conditions and/or as better alternatives to small molecule therapeutics. Concurrently, the remarkable advancement of recombinant biologics in the recent years has rendered the high-throughput synthesis of macromolecules into a routine and cost-effective process, further contributing to the renaissance of PDD [1].

Peptides, defined as macromolecules composed of 2-50 amino acids, will probably attract increasing interest in the coming decades. Their advantages include: high specificity and activity, easy degradation, do not yield toxic metabolites, and may be reutilized by the organism instead of being converted into waste products [1,2]. This implies that they generally possess reduced toxicity and few secondary effects. Indeed, the number of commercially available therapeutic peptides has in the last decades progressively increased (about 68 currently approved in the EU) [3], covering multiple clinical applications such as antineoplastics, antivirals, antifungals, antibiotics, modulators of the immune, cardiovascular and nervous systems, in addition to their utility in diagnosis.

Notwithstanding the benefits of peptide-based therapy, the translation of promissory peptides into clinical therapeutics continues to be a challenge due to their inherent bio- and physicochemical properties, *i.e.* are water-soluble and hence generally exhibit limited capacity to diffuse across biomembranes such as the gastrointestinal epithelium, are biologically unstable as they are rapidly metabolized by human proteolytic enzymes and thus yielding short plasma half-lives. Consequently, peptides are generally administered through injections, often several times a day, in detriment of patients' compliance and convenience [3].

The ultimate and long sought after goal is to achieve orally administrable therapeutic peptides. Nonetheless, this will require PDD paradigms that integrate comprehensive analyses of bioactivity, pharmacodynamic, pharmacokinetic and toxicological profiles of peptides in different phases of the PDD. Such workflows will allow for the design of peptides not only with favorable therapeutic efficacy, but also ensure their adequate bioavailability and administration.

In the path towards this goal, computational tools customized for predictive peptide modeling will be crucial, particularly in the context of the analysis of the existing experimental evidence to offer inferences on possible peptide bioactivity profiles. The utility of *in silico* tools in accelerating and optimizing drug discovery has long been recognized [4]. Moreover, the recent advances in machine learning algorithms and computing technology offer an opportunity to incorporate the state-of the art computational techniques in PDD workflows.

As maybe anticipated, successful *in silico* predictive modeling requires adequate characterization of compositional, chemical and physicochemical attributes of peptide molecules. However, from our extensive review of the literature we noted that while there is software for calculating descriptors for small molecules and proteins, there is no equivalent software particularly customized for peptide descriptor calculation, as macromolecules at the interface of small organic molecules and proteins. Usually, research groups build in-house scripts to compute descriptors from peptide sequences and amino acid properties or utilize the “Peptides package” of the R programming language which provides 10 structural characteristics for antimicrobial peptides [5-9].

Recently, there have been attempts to employ small molecule descriptor programs (e.g. Dragon, PaDEL, CoMFA) to build peptide bioactivity models but these have been limited to short lengths peptides *i.e.* less than 10 amino acids and mainly di-, tri- and tetrapeptides probably due

to the prohibitive computational cost of applying small molecule software [10,11]. Considering that in the last decade average length of peptides entering clinical development is of 20 amino acids [3], it is clear that the chemical space covered by these models is narrow. Additionally, in a recent study an effort to consider diverse lengths yielded rather modest correlations, i.e. $R^2 < 0.56$ [12], below the recommended limit of acceptability [13]. There is clearly a need for a user-friendly descriptor computing software customized for peptides.

On the other hand, while it is plausible that protein descriptors may be adopted as alternatives, these seem not to have gained traction in modeling of peptide bioactivity endpoints, probably because some protein descriptors may be redundant (e.g. popular sequence autocorrelation indices, defined to consider up to 30 lag values, would be redundant for short length peptides). Moreover, important protein descriptors such as the solvent accessible surface area, would not make much sense for short lengths peptide sequences.

We present herein, a user-friendly and cross-platform java-based software denominated **PeptiDesCalculator** for computing descriptors for peptide molecules. The following contributions may be highlighted: 1) we have collected and reimplemented existent sequence based protein descriptors, normalized and/or truncated to suite the peptide domain, 2) applied aggregation operators that generalize the traditional approach of the summation of the amino acid contributions to obtain global peptide descriptions [14-19] 3) selected the most orthogonal physicochemical, biochemical and topological amino acid indices from the amino acid index database and the literature [20,21], using the cluster analysis method, and incorporated in the aforementioned descriptor and generalization schemes, 4) provided a Graphical User Interface (GUI) to allow for the quick and straightforward descriptor computation by both experts and non-experts 4) parallelized the peptide descriptor computation to maximize the computation power available in state-of-the art

work stations. 5) a standalone version is provided instead of exclusive reliance on web platforms which present several limitations such as long queuing times, overwhelmed computational resources (users may not use private computing resources), or web disruptions, among others.

2. MATERIALS AND METHODS

2.1 Molecular Descriptors for Peptides

The following descriptors have been implemented in the PeptiDesCalculator software:

1) **Compositional descriptors**, which include the amino acid, dipeptide and tripeptide sequence composition.
2) **Composition Transition and Distribution** descriptors as proposed by Dubchak *et al.*[22] These descriptors characterize the global composition of given amino acid properties, the frequency with which these properties vary along the peptide sequences, and the corresponding property distribution patterns[22]. Taking hydrophobicity as an example, the amino acids may be classified as hydrophobic, neutral, and polar, respectively. For a given peptide sequence, the composition descriptors are defined as percentages for each class of amino acids. On the other hand, the transition descriptors are defined as percentages of the frequency with which an amino acid in one class is followed by another from a different class *i.e.* hydrophobic followed by neutral (or neutral followed by hydrophobic), polar followed by hydrophobic (or hydrophobic followed by polar) and neutral followed by polar (or polar followed by neutral). Finally, the distribution descriptors are percentages of sequence lengths within which the first amino acid, 25%, 50%, 75% and 100% of the amino acids with a given property are included.

3) **Conjoint Triad** descriptors as proposed by Shen *et al*.[23] These descriptors are defined following 3 main steps. Firstly, the 20 standard amino acids are clustered into 7 classes based on the dipoles and volumes of their side chains (Table 1).

Table 1. Classification of amino acids based on the sidechain dipoles and volumes.

Cluster No.	Dipole Scale(Debye)‡	Volume Scale(Å^3)†	Class
1	-	-	Ala, Gly, Val
2	-	+	Ile, Leu, Phe, Pro
3	+	+	Tyr, Met, Thr, Ser
4	++	+	His, Asn, Gln, Tpr
5	+++	+	Arg, Lys
6	+'+'+'	+	Asp, Glu
7	+	+	Cys§

‡ Scale: (-) dipole < 1.0 ; (+) 1.0 < dipole < 2.0; (++) 2.0 < dipole < 3.0; (+++) dipole > 3.0; (+'+'+') dipole > 3.0 with opposite orientation. †Scale: (-) volume < 50; (+) volume > 50. § Cys(Cysteine) not included in cluster 3 due to its capacity to form disulfide bonds.

Next, the frequency of amino acid triads (*i.e.* units of 3 contiguous amino acids) is determined, with a particularity that units with amino acids belonging to the same classes (Table 1) are considered as equivalent since they are deemed to play a similar role. Bearing in mind that the amino acids are stratified into 7 clusters, the total number of triads is 343 (*i.e.* 7 x 7 x 7). For a given peptide sequence, the frequency of each triad is determined yielding a vector **F** (f_i) where f_i is the frequency of triad t_i.

Finally, the conjoint triad descriptor is a vector **D**(t_i), defined as:

$$\mathbf{D}(t_i) = (f_i - min\ \{f_1, f_2, f_3, ..., f_{343}\}) / max\ \{f_1, f_2, f_3, ..., f_{343}\} \quad \textbf{(1)}$$

where *min* and *max* refer to the minimum and maximum frequencies in the vector **D**(t_i).

1) **Generalized Peptide Indices**: these are descriptor families to which the aggregation operators as alternatives to the classical linear combination of amino acid contributions are applied. These generalizable indices may be stratified into 3 classes:

a) ***Global Peptide Indices***: derived from topological, physicochemical, chemical and biological properties of amino acids comprising peptide sequences. For a given peptide sequence, a vector $\mathbf{V}_p$ is generated based on the selected property values of constituent amino acids,

$\mathbf{V}_p = [p_1, p_2, p_3 \ldots p_N]$,

where N refers to the number of amino acids in a sequence. The amino acid properties considered in the present study were compiled from the AA index database and the literature[20,21]. A total of 520 comprising of physicochemical, biochemical and topological amino acid properties were retrieved. Given this high number of properties and their possible correlation, dimensionality reduction was deemed necessary. To this end, *k*-means cluster analysis (*k*-CA) was employed. The *k*-CA algorithm aims to stratify a set of objects (features or instances) into *k* clusters such that similar objects, as determined by a given similarity score, are assigned to the same clusters. From an optimization perspective, the *k*-CA may be understood as a min-max problem, where the intra-cluster variance is sought to be minimized while the inter-cluster variance is maximized. The partitioning of objects into *k* clusters allows for the selection of representative members from each cluster, and thus serving as a dimensionality reduction tool. For the *k*-CA performed herein, the squared Euclidean distance was employed as the similarity measure and the number of clusters (*k*) was set at 12. Subsequently, 176 representative amino acid properties were selected for computing the global peptide descriptors. For a given peptide sequence, $\mathbf{V}_p$ is derived for each property and subsequently the aggregation operators in subsection **Generalization**

Scheme of the Linear Combination of Parts are applied yielding the corresponding global peptide descriptors.

b) ***Sequence Order Coupling derived Descriptors***: include the quasi-sequence order (QSO), pseudo-amino acid composition (PseAAC) and amphiphilic PseAAC indices, as proposed by Chou[24,25]. The quasi-sequence order (20 + *lag*) dimensional vector $\mathbf{V}_{QSO}$ is comprised of a union of $\mathbf{QSO}_a$ and $\mathbf{QSO}_{a+l}$ vectors derived as follows:

$$\mathbf{QSO}_{\boldsymbol{a}} = \frac{f_a}{\sum_{a=1}^{20} f_a + w\sum_{l=1}^{4} \nabla_l}, \quad l = 1,2,3,4\,; \quad 1 \le a \le 20 \qquad (\mathbf{2})$$

$$\mathbf{QSO}_{\boldsymbol{a+l}} = \frac{w\nabla_{a-20}}{\sum_{a=1}^{20} f_a + w\sum_{l=1}^{4} \nabla_l}, \quad l = 1,2,3,4\,; \;\; 20+1 \le a \le 20+l \qquad (\mathbf{3})$$

where f_a is the frequency of amino acid a, w is an empirical weighting factor set to 0.75, $\nabla_l = \sum_{i=1}^{N-l}\left(d_{i,i+l}\right)^2$, also known as the sequence order coupling number, $d_{i,\,i+l}$ is the physicochemical distance between the amino acids at positions i and $i+l$, as defined by Schneider and Wrede[26]. The physicochemical distance metric is defined the Euclidean distance between vectors comprising of 4 physicochemical properties for amino acids, *i.e.* hydrophobicity, hydrophilicity, polarity and side-chain volume.

The pseudo-amino acid composition vector $\mathbf{V}_{PseAAC}$ is comprised of the $\mathbf{PseAAC}_a$ and $\mathbf{PseAAC}_{a+l}$, and are defined as follows:

$$\mathbf{PseAAC}_{\boldsymbol{a}} = \frac{f_a}{\sum_{a=1}^{20} f_a + w\sum_{l=1}^{4} \nabla_l}, \quad l = 1,2,3,4\,; \qquad 1 \le a \le 20 \qquad (\mathbf{4})$$

$$\mathbf{PseAAC}_{\boldsymbol{a+l}} = \frac{w\nabla_{a-20}}{\sum_{a=1}^{20} f_a + w\sum_{l=1}^{4} \nabla_l}, \quad l = 1,2,3,4\,;\; 20+1 \le a \le 20+l \qquad (\mathbf{5})$$

where f_a is the frequency of amino acid a, $\nabla_l = \frac{1}{N-l}\sum_{l=1}^{N-l}\Theta(A_i, A_{i+l})$, denominated as the sequence order correlation factor, $\Theta(A_i, A_{i+l})$ is the correlation amino acid properties and w is an empirical weighting factor set to 2.5. The correlation factor describes the similarity between amino acids based on the average squared Euclidean distance between normalized hydrophobicity, hydrophilicity and side-chain mass values, as expressed by equation 6 [25]:

$$\Theta(A_i, A_j) = \frac{1}{3}\left\{\left[H_{pho}(A_i) - H_{pho}(A_j)\right]^2 + \left[H_{phi}(A_i) - H_{phi}(A_j)\right]^2 + \left[M(A_i) - M(A_j)\right]^2\right\} \quad \textbf{(6)}$$

where $H_{\text{pho}}(A_i)$, $H_{\text{phi}}(A_i)$, $M(A_i)$ are normalized hydrophobicity, hydrophilicity and side-chain mass of the amino acid A_i, obtained as follows:

$$H_{pho}(A_i) = \frac{H^0_{pho}(i) - \sum_{i=1}^{20}\frac{H^0_{pho}(i)}{20}}{\sqrt{\frac{\sum_{i=1}^{20}\left[H^0_{pho}(i) - \sum_{i=1}^{20}\frac{H^0_{pho}(i)}{20}\right]^2}{20}}}$$

$$H_{phi}(A_i) = \frac{H^0_{phi}(i) - \sum_{i=1}^{20}\frac{H^0_{phi}(i)}{20}}{\sqrt{\frac{\sum_{i=1}^{20}\left[H^0_{phi}(i) - \sum_{i=1}^{20}\frac{H^0_{phi}(i)}{20}\right]^2}{20}}}$$

$$M(A_i) = \frac{M^o(i) - \sum_{i=1}^{20}\frac{M^o(i)}{20}}{\sqrt{\frac{\sum_{i=1}^{20}\left[M^o(i) - \sum_{i=1}^{20}\frac{M^o(i)}{20}\right]^2}{20}}}$$

where $H^0_{pho}(i)$, $H^0_{pho}(i)$ and $M^o(i)$ are the original hydrophobicity, hydrophilicity and side-chain mass of the i^{th} amino acid, respectively.

For the amphiphilic pseudo amino acid composition $\mathbf{V}_{\text{APseAAC}}$ is comprised of the $\mathbf{APseAAC}_{a}$ and $\mathbf{APseAAC}_{a+l}$, and are defined as follows:

$$\mathbf{APseAAC}_{\boldsymbol{a}} = \frac{f_a}{\sum_{a=1}^{20} f_a + w\sum_{l=1}^{4} \nabla_l}, \quad l = 1,2,3,4\,;\ 1 \le a \le 20 \qquad (\mathbf{7})$$

$$\mathbf{APseAAC}_{\boldsymbol{a+l}} = \frac{w\nabla_a}{\sum_{a=1}^{20} f_a + w\sum_{l=1}^{2l} \nabla_l}, \quad l = 1,2,3,4\,;\ 20+1 \le a \le 20+2l \qquad (\mathbf{8})$$

where f_a is the frequency of amino acid a, $\nabla_l = \frac{1}{N-l}\sum_{l=1}^{N-l} H_{i,i+l}$ (H is correlation function for hydrophobicity $H_{i,i+l}^{1}$ and hydrophilicity $H_{i,i+l}^{2}$, respectively) and w is 2.5. To compute the generalized peptide indices, aggregation operators are applied to $\mathbf{V}_{\text{QSO}}$, $\mathbf{V}_{\text{PseAAC}}$ and $\mathbf{V}_{\text{APseAAC}}$ yielding various peptide descriptors (see subsection **Generalization Scheme of the Linear Combination of Parts**).

c) ***Autocorrelation Descriptors*** : comprised of the Geary, Moran and normalized Moreau-Broto Autocorrelation descriptors, and are expressed by the equations 9, 10 and 11 respectively [27-29].

$$\text{AC}_l = \frac{\sum_{i=1}^{N-l} P_i P_{i+l}}{(N-l)} \qquad l = 1,2,3,4 \qquad (\mathbf{9})$$

$$\text{MA}_l = \frac{\frac{1}{N-l}\sum_{i=1}^{N-l}(P_i - \bar{P})(P_{i+l} - \bar{P})}{\frac{1}{N}\sum_{i=1}^{N}(P_i - \bar{P})^2} \qquad l = 1,2,3,4 \qquad (\mathbf{10})$$

$$\text{GA}_l = \frac{\frac{1}{2(N-l)}\sum_{i=1}^{N-l}(P_i - P_{i+l})^2}{\frac{1}{N-1}\sum_{i=1}^{N}(P_i - \bar{P})^2} \qquad l = 1,2,3,4 \qquad (\mathbf{11})$$

where l is the autocorrelation lag, P_i and P_{i+l} are properties of amino acid at position i and $i+l$, and $\bar{P}$ the average value of property P, $\bar{P} = \sum_{i=1}^{N} P_i/N$. As is evident in equation 9, 10 and 11 the total AC_l, MA_l and GA_l indices for each lag value involve the summation of the calculated autocorrelation values. This formalism paves way for other aggregation operators other than summation operator to be applied as explained below.

2.2 Generalization Scheme of the Linear Combination of Parts

Classically, sequence-based protein descriptors apply the summation of the amino acid/pair-wise contributions. This formalism in effect suggests that these contributions are necessarily additive. However, in biomolecules this assumption is often inaccurate and corrections are often applied to account for the non-additive nature of macromolecular properties (e.g. interaction potentials are often volume corrected when applied to protein systems [30]).

In the same spirit, we apply different aggregation operators (AOs) that generalize the linear combination of amino acid and/or pairwise contributions. These AOs have been applied in the definition of descriptors for small molecules and comparative studies demonstrated that these generally performed better in predicting physicochemical and biological properties of organic molecules, when compared to the summation operator [14-19,31-33]. These AOs are stratified in 3 groups: 1) Norms which are comprised of: Minkowski's norms N1, N2 and N3 (note that N1 is equivalent to the summation operator). 2) Means: Arithmetic Mean (M), Geometric Mean (G), Harmonic Mean (A), Quadratic Mean (P2) and Power Mean (P3). 3) Statistical invariants which include: Variance (V), Standard Deviation (SD), Variation Coefficient (VC), Skewness (S), Kurtosis (K), Percentile 25 (Q1), Percentile 50 (Q2), Percentile 75 (Q3), Inter-quartile Range (I50), X min (MN), X max (MX) and Range (R). The mathematical expressions for these aggregation operators are provided in Table 2.

Note that the Geary, Moran and normalized Moreau-Broto autocorrelation descriptors may in turn be employed as generalization schemes to other descriptors formalisms (e.g. sequence order coupling derived descriptors) which yield vectors of amino acid/pair-wise contributions.

Table 2. Norms, Means and Statistical AOs employed to Generalize the Summation Operator.

Group	Name	Identifier	Formula
Norms	Minkowsky norm (p = 1) *Manhattan norm*	N1	$N1 = \sum_{i=1}^{n} A_i$
	Minkowsky norm (p = 2) *Euclidean norm*	N2	$N2 = \sqrt{\sum_{i=1}^{n} A_i}^{2}$
	Minkowsky norm (p = 3)	N3	$N3 = \sqrt[3]{\sum_{i=1}^{n} A_i}^{3}$
Means	Geometric Mean	G	$G = \sqrt[n]{\prod_{i=1}^{n} A_i}$
	Arithmetic Mean (Power mean of degree β = 1)	M	$M_\beta = \left(\frac{A_1^\beta + A_2^\beta + \cdots + A_n^\beta}{n}\right)^{\frac{1}{\beta}}$
	Quadratic Mean (Power mean of degree β = 2)	P2	
	Power mean of degree β = 3	P3	
	Harmonic Mean (Power mean of degree β = -1)	H	
Statistical Operators	Variance	V	$V = \frac{\sum_{i=1}^{n}(A_i - M)}{n-1}$
	Skewness	S	$S = \frac{n * (X_3)}{(n-1)(n-2)(\sigma)^3}$ $X_3 = \sum_{i=1}^{n}(A_i - M)^3$ M, arithmetic mean σ, standard deviation
	Kurtosis	K	$K = \frac{n(n+1)X_4 - 3(X_2)(X_2)(n-1)}{(n-1)(n-2)(n-3)(\sigma)^4}$ $X_j = \sum_{i=1}^{n}(A_i - M)^j$ M, arithmetic mean SD, standard deviation
	Standard Deviation	SD	$SD = \sqrt{\frac{(\sum_{i=1}^{n} A_i - M)^2}{n-1}}$
	Variation Coefficient	VC	$VC = \sigma / M$
	Range	R	$R = A_{max} - A_{min}$

Percentile 25	Q1	$Q1 = \left\lceil \frac{N}{4} + \frac{1}{2} \right\rceil$ N, vector size
Percentile 50	Q2	$Q2 = \left\lceil \frac{N}{2} + \frac{1}{2} \right\rceil$ N, vector size
Percentile 75	Q3	$Q3 = \left\lceil \frac{3N}{4} + \frac{1}{2} \right\rceil$ N, vector size
Inter-quartile Range	I50	$I50 = Q3 - Q2$
Maximum value	MX	MX = A_i max
Minimum value	MN	MN = A_i min

2.3 Design and Implementation

The PeptiDesCalculator is standalone software developed in Java programming language (version 1.8) and can thus be run on any operating system that has the Java Virtual Machine (JVM) installed. The PeptiDesCalculator integrates both the front-end and back-end layers. The former contains the Graphic User Interface (GUI), which allows for the descriptor configuration, while the latter contains implementations for these molecular descriptors.

Front end: PeptiDesCalculator Graphic User Interface

The GUI was designed to allow for a simple and user-friendly configuration of the peptide molecular descriptors (MDs) computation. Figure 1 is a snapshot of the PeptiDesCalculator GUI. This contains 2 main tabs, the first (denominated Descriptors) for selecting the MDs to be computed and the second (Generalization Scheme) contains the different operators that generalize the classical approach of summing amino acid contributions to obtain global peptide parameters. Moreover, dialogs for selecting the input file(s) and defining the output file paths are provided. The PeptiDesCalculator supports the Tab Separated Value (.txt) and Protein Data Bank (.pdb) as input file formats. The computed descriptors are saved as Tab Separated Value (.txt) files. Considering that the software provides as many as 48485 MDs, Comma Separated Value

(.csv) file format was not considered as a possible output file format as it is not amenable to process such an enormous number of features.

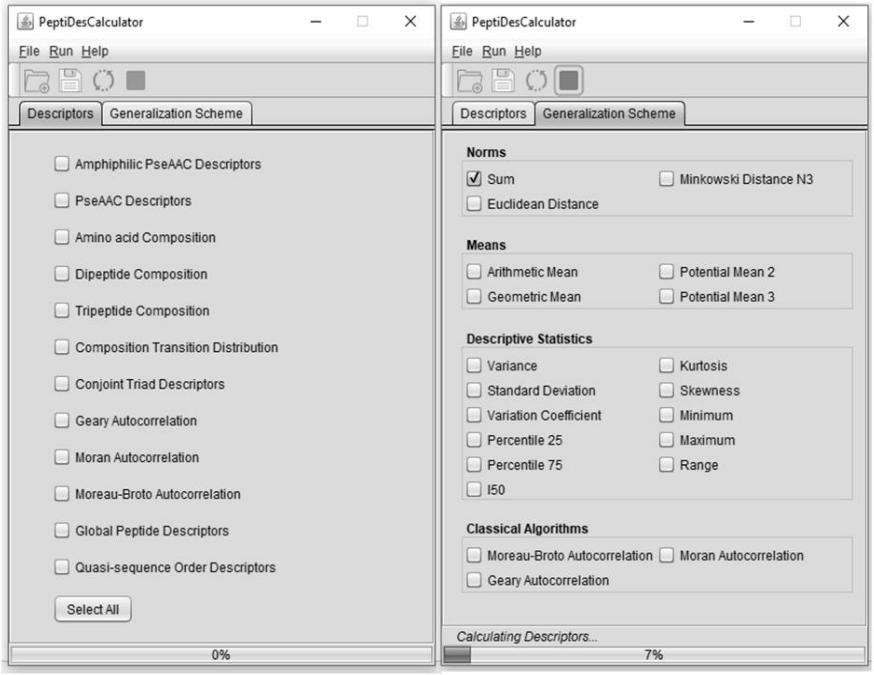

Figure 1. PeptiDesCalculator Graphic User Interface (GUI) showing 2 main tabs: Descriptors - for selecting the MDs to be computed and Generalization Scheme – for configuring the different operators.

Back end: Infrastructure for Peptide Descriptor Computation

The tasks (descriptor calculation) determined by the client through the GUI configuration are processed by the PeptideDescriptor library. This library is stratified in the peptides and AOs packages, where the latter contains the classes for the MD computation, while the former is comprised of classes dedicated to the different generalization schemes.

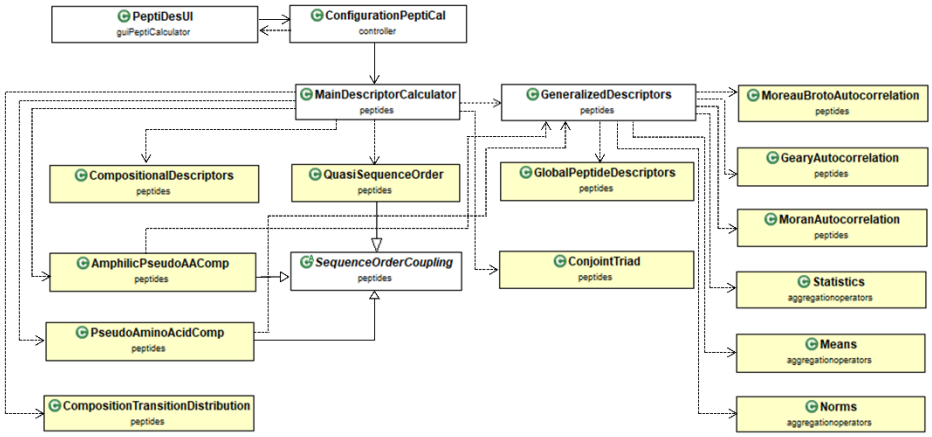

Figure 2. UML diagram for the key classes responsible for the MD calculation.

Figure 2 is the Unified Modeling Language (UML) diagram for the key classes responsible for the MD calculation. As may be observed, the *Quasi Sequence Order*, *AmiphilicPseudoAminoAcidComposition* and the *Pseudo Aminoacid Composition* classes support the *SequenceOrderCoupling* abstraction. Moreover, with the exception of the *Compositional* and *CompositionTransitionDistribution* classes, the rest of the peptide descriptor computation classes invoke the *GeneralizedDescriptors* framework, consistent with the notion of providing alternative descriptions to the linear combination of amino acid contributions. On the other hand, the controller package acts as an intermediate layer, handling the interaction between the backend and frontend layers.

Finally, considering that each descriptor calculation may be performed independently, the PeptiDesCalculator software was designed following a parallel processing framework. In this sense, the submitted tasks are assigned to distinct threads depending on the number of available cores and consequently enabling their parallel execution.

3. RESULTS AND DISCUSSION

3.1 Parallel Computing Efficiency

Contemporary computer workstations are typically multiprocessor systems, and thus often yield improved absolute performance (relative to uniprocessor systems), when computations are divided into sub-tasks and executed simultaneously on different processing units. In this subsection, we sought to examine the possible efficiency of the implemented parallel computing framework in accelerating the speed of the peptide MDs computation when performed on multiprocessor systems. For this study we employed the starPepDB dataset comprising of 48,335 peptides of diverse lengths and composition (3318 were skipped as they contained more than 50 amino acids, consistent with the standard peptides definition) [34]. The starPepDB dataset is freely available at http://mobiosd-hub.com/starpep. The peptide descriptor computations were performed on a Medion computer workstation with the following properties: AMD A10-8750 Radeon R7, 12 Compute Cores 4C + 8G 3.60 GHz, 8GB RAM. Note that for this experiment, only 4GB RAM were allocated to the JVM to execute PeptiDesCalculator software. For this study, two descriptor groups were formed with the first comprising of the pseudo-amino acid composition, Geary autocorrelation, global peptide and conjoint triad descriptors, and the second group contained the compositional and composition, transition and distribution descriptors. Table 3 illustrates the total and average computation time, as well as speedup and efficiency metrics for the two descriptor groups.

It is evident from Table 3 that the total processing time generally decreases with an increase in the number of processors, and it can therefore be inferred that the parallel computing architecture was adequately implemented.

Table 3. Speedup analysis of the PeptiDesCalculator indices for a dataset comprising of 48,335 peptides of diverse sequence lengths.

Number of Processors	Processing time(sec)	Speedup†	Efficiency‡	Processing time for one molecule (sec)	Processing time for one MD (sec)
PseAAC, Geary Autocorrelation, Global Peptide Descriptors, Conjoint Triad (15721 Descriptors)					
1	6785.72	1.00	1.00	0.14	0.43
2	5646.53	1.20	0.83	0.12	0.36
4	4321.52	1.57	0.64	0.09	0.27
Compositional Descriptors, Composition Transition Distribution (4577 Descriptors)					
1	1964.68	1.00	1.00	0.04	0.43
2	2178.44	0.90	1.11	0.05	0.48
4	923.56	2.13	0.47	0.02	0.20

†**Speedup**: ratio of the processing time for a baseline sequential workflow (*i.e.* using one processor) to the time taken with a parallelized framework to execute the same task on n processors ($n > 1$). ‡**Efficiency**: ratio of speedup to the number of processors.

3.2 Evaluation of Predictive Capacity of PeptiDesCalculator Indices.

In order to assess the utility of the PeptiDesCalculator indices in the modeling of peptide bioactivity profiles, we selected 8 endpoints *i.e.* Hepatitis C inhibitory activity (407 peptides), anti-breast cancer (240 peptides), anti-colon cancer (227 peptides), HIV inhibitory activity(532 peptides), anti-skin cancer (188 peptides), *C. albicans* (781 peptides), *P. aeruginosa* (890 peptides) and Listeria (141 peptides) inhibitory activities. For each endpoint, a binary classification model was developed. Bearing in mind that the reported metric for the different inhibitory profiles was the half-maximal inhibitory concentration (IC_{50}) [34], each bioactivity endpoint (response variable) was first transformed into a binary variable (*i.e.* active/inactive) to render it amenable for classification model building. For the antineoplastic activity, a threshold value of 20μM was considered (*i.e.* peptides with IC_{50} values $\leq$ 20μM were labelled as active, while those with IC_{50} values > 20μM were labelled as inactives). For the rest of the endpoints, a threshold value of 10μM was employed.

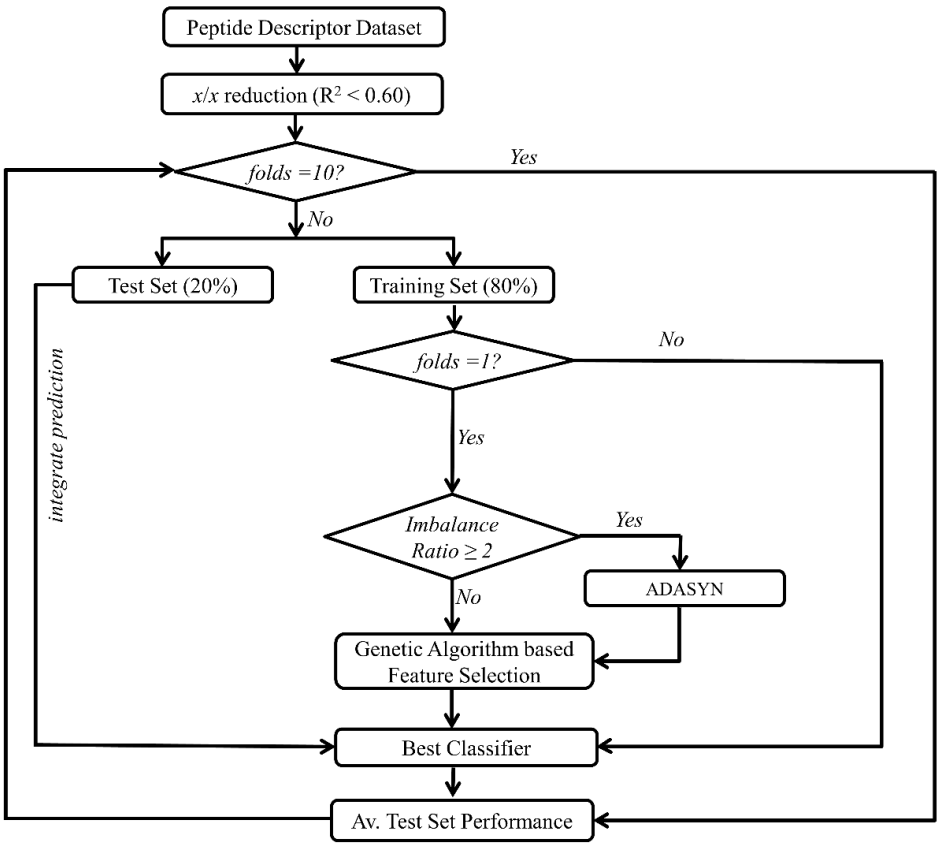

Figure 3. General workflow followed in the modeling of the 9 peptide bioactivity endpoints in the present report.

Figure 3 shows the workflow followed in modeling of these endpoints. Briefly, the PeptiDesCalculator indices were computed for each of the datasets and the pair-wise (*x*/*x*) coefficient of determination calculated to exclude the highly correlated indices (*i.e.* only indices with $R^2 < 0.60$ were retained). The resulting data matrices were randomly divided into training (80%) and test (20%) sets, with the former dedicated to the classification model building and the latter to the evaluation of the models' predictive capacity. For imbalanced datasets (*i.e.* containing disproportionately more peptides in one class relative to the other) the respective imbalance ratios, IR = [number of peptides in majority class/number of peptides in minor class] were determined and for datasets with IR ≥ 2, the Adaptive Synthetic (ADASYN) oversampling approach was applied to the minority class to yield balanced class distributions.

Next, a genetic algorithm (GA) was applied to the training data matrix in order to select the subsets of PeptiDesCalculator indices that yield the best classification models. The GAs are optimization methods designed to mimic the natural selection process in that the fittest individuals (solutions), as represented by their chromosomes, progressively evolve towards more optimum solutions. From a model building perspective, the chromosomes (solutions) are the classification models, and the genes the variables. For each generation, segments of the most optimum chromosomes are crossed over (*i.e.* reproduce) and some genes (variables) randomly mutated for others, thus yielding new chromosomes whose performance is in turn evaluated. For the model building procedure employed herein, following GA configuration setting was employed: population size = 100, crossover probability=0.5, mutation probability = 0.2 and number of generations =100.

The predictivity of the built classifiers was assessed over 10 fold external validation sets in terms of the classification metrics: accuracy (ACC), sensitivity (SE), specificity(SP), Precision (PR) and Mathew's correlation coefficient (MCC), respectively. Table 4 shows the average test set classification parameters obtained using a 10-fold external validation procedure for each of the modeled bioactivity endpoints (configuration parameters and matrices of final features contained in built classifiers provided as supplementary information, SI1-2). As may be observed, the built classifiers generally demonstrate robust predictive power as demonstrated by the respective Cooper statistics, *i.e.* ACC = 0.687 - 0.858, SE = 0.609 - 0.838, SP = 0.591 - 0.912, PR = 0.661 - 0.884 and MCC = 0.349 - 0.577. The limits of acceptability for test set classifier performance are: ACC, SE, SP and PR > 0.5 and MCC > 0 [35]. The best performance was obtained for anti-listeria activity Support Vector Machine (SVM) classifier (ACC = 0.819, SE = 0.838, SP = 0.753, PR = 0.884, MCC = 0.577), followed by Hepatitis C inhibitory activity

Random Forest (RF) classifier (ACC = 0.792, SE = 0.807, SP = 0.781, PR = 0.748, MCC = 0.587) and HIV inhibitory activity RF classifier (ACC = 0.788, SE = 0.767, SP = 0.810, PR = 0.805, MCC = 0.579). The least favorable performance is provided by the *C. albicans* inhibitory activity Gradient Boosting (GB) classifier (ACC = 0.687, SE = 0.752, SP = 0.591, PR = 0.713, MCC = 0.349). Even then the parameters for this classifier are well above the limit of random performance.

Table 4. Average 10-fold external validation performance of classification models for the 8 peptide bioactivity endpoints.[†]

Activity	Act./Inact.[‡]	Classifier[§]	ACC	SE	SP	PR	MCC
HIV	261/270	RF	0.788	0.767	0.810	0.805	0.579
Breast cancer	75/165	RF	0.787	0.648	0.859	0.700	0.517
Colon cancer	47/180	GB	0.781	0.609	0.849	0.571	0.441
Skin cancer	39/149	RF	0.858	0.658	0.912	0.661	0.569
C. albicans	120/661	GB	0.687	0.752	0.591	0.713	0.349
Hepatitis C	182/225	RF	0.792	0.807	0.781	0.748	0.587
Listeria	39/149	SVM	0.819	0.838	0.753	0.884	0.577
P. aeruginosa	505/385	RF	0.781	0.809	0.746	0.795	0.558

[†]Classifier configurations are provided as supporting information; [§]RF: Random Forest, GB: Gradient Boosting, SVM: Support Vector Machine; [‡]Act.: Active, Inact.: Inactive

3.3 Comparison with Other Approaches in the Literature.

Herein, we sought to evaluate the predictivity of the PeptiDesCalculator indices relative to the state-of-the art approaches employed in modeling the bioactivity of peptides. To this end, we retrieved from the literature a recently built dataset comprising of 1778 antimicrobial peptides (AMPs) and an equal number of decoys (non-AMPs) with a sequence length distribution similar to the former, yielding a total of 3,556 peptides [36]. Bearing in mind that PeptiDesCalculator software computes over 48000 indices, we sought to reduce the initial set of indices to be considered for the model building. In this sense, the PeptiDesCalculator indices computed for the AMPs dataset considered only the following representative operators from each group of the

AOs: N1 and N2 for the norms, GM, M, P2 and H for the means, and V, S, K, 5, I50 for the statistical invariants. Consistent with the standard definition of peptides (*i.e.* macromolecules composed of 2-50 amino acids), the PeptiDesCalculator software filtered out macromolecules with more than 50 amino acids; their identity is provided in the *inputErrors.log* file. Following the same procedure discussed in the previous section and illustrated in Figure 3, antimicrobial activity classification models were built using the retrieved dataset. To approximate the dataset size employed in the reference study, the test dataset size was set to 33% of the entire dataset. Table 5 compares test set performance of the Deep Neural Network (DNN), Collection of Anti-Microbial Peptides (CAMP) models based on RF, SVM, Artificial Neural Networks (ANN) and Discriminant Analysis (DA), as well as the gapped-kmer-SVM classifier (gkmSVM) classifier [37,38].

Table 5. Comparison of test set performance of PeptiDesCalculator based models and the state-of the art approaches reported in the literature.

Method	Classifier	Features	ACC(%)	SE(%)	SP(%)	MCC
DNN[34]	-	200	91.01	89.89	92.13	0.820
CAMP[34]	RF	64	87.57	92.70	82.44	0.755
CAMP[34]	SVM	64	84.41	88.90	79.92	0.691
CAMP[34]	ANN	64	84.04	82.98	85.09	0.681
gkmSVM[34]	SVM	$l=9$, k=*6*	89.46	88.34	90.59	0.790
PeptiDesCal[†]	SVM	15	91.17	91.43	90.93	0.824

[†]10-fold test set validation is performed to eliminate possible dependence on test set selection

While direct comparisons of test set performance may not be made since the study by Veltri *et. al* [36] included sequences with more than 50 amino acids (approx. 15% of the unfiltered test dataset assuming equal training-test set distribution) and considered a 1-fold test set stratification (in place of 10-folds as in the present study), it may be inferred that the PeptiDesCalculator indices based model generally yield similar performance with the DNN and CAMP-RF classifiers which were the best models reported by Veltri *et. al*.

The PeptiDesCalculator SVM model was built using only 15 features, while the CAMP-RF classifier employed 64 features, highlighting the greater simplicity for the former notwithstanding the difference in test set size (data matrix of the 15 features employed to build the SVM classifier provided as supplementary information, SI2). It is indeed implausible that the number of features in the latter is four times that of the former only to achieve accurate prediction of approximately 15% of the unfiltered test (sequences with more than 50 amino acids). Models characterized by a lower degree of freedom are considered to be more robust and thus less prone to fortuitous correlation.

The comparable performance of the PeptiDesCalculator SVM classifier and the DNN model, echoes the need to reconsider the push to adopt sophisticated deep learning algorithms to solve classical modeling problems in detriment of the efforts to define more accurate and diverse approaches for codifying chemical structural information. For a given modeling task, the gains in accuracy using complex algorithms should be blatant for their use to be justified. In the study by Veltri *et. al* [36], although the DNN directly uses the peptide sequences to build an antimicrobial classifier, the subsequent model complexity due to the embedding, convolutional, max-pooling and Long Short Term Memory (LSTM) layers, prior to the output layer, outweighs the possible gains obtained from the direct use of the peptide sequences, since no major improvements in test set performance are observed. On the other hand, the PeptiDesCalculator indices employed in the SVM classifier are based on the application of simple mathematical operators on peptide sequences or vectors of amino acid physical, chemical or physicochemical properties which on one hand involve a rather low computational cost and are interpretable in physicochemical and/or chemical structural terms (the meaning of each of these features is available at https://www.genome.jp/aaindex/AAindex/list_of_indices).

4. CONCLUSION

The PeptiDesCalculator software provides a user-friendly platform for computing theoretical descriptors for peptide molecules. In light of the satisfactory performance of the models built with the PeptiDesCalculator indices, it may be inferred that these codify relevant peptide structural, chemical and physicochemical information, useful in the prediction of peptide bioactivity profiles. It is hoped that this computational program will facilitate the development of *in silico* models for the prediction of peptide bioactivity, pharmacokinetic and toxicological profiles and consequently guide the discovery, design and optimization of therapeutically interesting peptides. The PeptiDesCalculator software is available for academic use upon request at info@protoqsar.com.

Acknowledgement

SJB acknowledges the funding assistance from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 893810.

Associated Content

Supporting Information. Configuration of classification models; matrix of features comprising built classifiers for studied bioactivity profiles. This material is available free of charge via the Internet at

Conflict of interest: none declared.

‡Current address: Eurofins Agroscience Services Regulatory Spain SL, Sorolla Center, Av. Cortes Valencianas 58, Valencia 46015, Spain

5. REFERENCES

1. Henninot A, Collins JC, Nuss JM. The current state of peptide drug discovery: back to the future? *J Med Chem.* 2017;61(4):1382-1414.
2. Vlieghe P, Lisowski V, Martinez J, Khrestchatisky M. Synthetic therapeutic peptides: science and market. *Drug Discov Today.* 2010;15(1-2):40-56.
3. Lau JL, Dunn MK. Therapeutic peptides: Historical perspectives, current development trends, and future directions. *Bioorg Med Chem.* 2018;26(10):2700-2707.
4. Jorgensen WL. The many roles of computation in drug discovery. *Science.* 2004;303(5665):1813-1818.
5. Waghu FH, Gopi L, Barai RS, Ramteke P, Nizami B, Idicula-Thomas S. CAMP: Collection of sequences and structures of antimicrobial peptides. *Nucleic Acids Res.* 2014;42(D1):D1154-D1158.
6. Xiao X, Wang P, Lin W-Z, Jia J-H, Chou K-C. iAMP-2L: a two-level multi-label classifier for identifying antimicrobial peptides and their functional types. *Anal Biochem.* 2013;436(2):168-177.
7. Usmani SS, Bhalla S, Raghava GP. Prediction of antitubercular peptides from sequence information using ensemble classifier and hybrid features. *Front Pharmacol.* 2018;9:954.
8. Osorio D, Rondón-Villarrea P, Torres R. Peptides: a package for data mining of antimicrobial peptides. *R Journal.* 2015;7(1).
9. Meher PK, Sahu TK, Saini V, Rao AR. Predicting antimicrobial peptides with improved accuracy by incorporating the compositional, physico-chemical and structural features into Chou's general PseAAC. *Sci Rep.* 2017;7:42362.
10. Haney EF, Brito-Sánchez Y, Trimble MJ, Mansour SC, Cherkasov A, Hancock RE. Computer-aided discovery of peptides that specifically attack bacterial biofilms. *Sci Rep.* 2018;8(1):1871.
11. Guo H, Wang Y, He Q, et al. In silico rational design and virtual screening of antioxidant tripeptides based on 3D-QSAR modeling. *J Mol Struct.* 2019;1193:223-230.
12. Mathur D, Singh S, Mehta A, Agrawal P, Raghava GP. In silico approaches for predicting the half-life of natural and modified peptides in blood. *PloS ONE.* 2018;13(6):e0196829.
13. Gramatica P. Principles of QSAR models validation: internal and external. *QSAR Comb Sci.* 2007;26(5):694-701.
14. Barigye SJ, Marrero-Ponce Y, Alfonso-Reguera V, Pérez-Giménez F. Extended GT-STAF information indices based on Markov approximation models. *Chem Phys Lett.* 2013;570:147-152.
15. Barigye SJ, Marrero-Ponce Y, Martínez-López Y, et al. Event-Based Criteria in GT-STAF Information Indices: Theory, Exploratory Diversity Analysis and QSPR Applications. *SAR & QSAR Environ Res.* 2012:3-34.
16. Barigye SJ, Marrero-Ponce Y, Martínez-López Y, et al. Relations Frequency Hypermatrices in Mutual, Conditional and Joint Entropy-Based Information Indices. *J Comp Chem.* 2013;34(4):259-274.
17. Barigye SJ, Marrero-Ponce Y, Martínez-Santiago O, Martínez-López Y, Torrens F. Shannon's, Mutual, Conditional and Joint Entropy-Based Information Indices. Generalization of Global Indices Defined from Local Vertex Invariants *Curr Comput-Aided Drug Des* 2013;9:164-183.

18. Barigye SJ, Marrero-Ponce Y, Zupan J, Pérez-Giménez F, Freitas MP. Structural and physicochemical interpretation of GT-STAF information theory-based indices. *Bull Chem Soc Jpn.* 2014;88(1):97-109.
19. Marrero-Ponce Y, Santiago OM, López YM, Barigye SJ, Torrens F. Derivatives in discrete mathematics: a novel graph-theoretical invariant for generating new 2/3D molecular descriptors. I. Theory and QSPR application. *J Comput Aided Mol Des.* 2012;26(11):1229-1246.
20. Mauri A, Ballabio D, Consonni V, Manganaro A, Todeschini R. Peptides multivariate characterisation using a molecular descriptor based approach. *Match Commun Math Comput Chem.* 2008;60:671-690.
21. Kawashima S, Pokarowski P, Pokarowska M, Kolinski A, Katayama T, Kanehisa M. AAindex: amino acid index database, progress report 2008. *Nucleic Acids Res.* 2007;36(suppl_1):D202-D205.
22. Dubchak I, Muchnik I, Holbrook SR, Kim S-H. Prediction of protein folding class using global description of amino acid sequence. *Proc Natl Acad Sci USA.* 1995;92(19):8700-8704.
23. Shen J, Zhang J, Luo X, et al. Predicting protein–protein interactions based only on sequences information. *Proc Natl Acad Sci USA.* 2007;104(11):4337-4341.
24. Chou K-C. Prediction of protein subcellular locations by incorporating quasi-sequence-order effect. *Biochem Biophys Res Commun.* 2000;278(2):477-483.
25. Chou KC. Prediction of protein cellular attributes using pseudo-amino acid composition. *Proteins: Struct, Funct, Bioinf.* 2001;43(3):246-255.
26. Schneider G, Wrede P. The rational design of amino acid sequences by artificial neural networks and simulated molecular evolution: de novo design of an idealized leader peptidase cleavage site. *Biophys J.* 1994;66(2):335-344.
27. Moreau G, Broto P. Auto-correlation of molecular-structures, application to sar studies. *Nour J Chim.* 1980;4(12):757-764.
28. Moran PA. Notes on continuous stochastic phenomena. *Biometrika.* 1950;37(1/2):17-23.
29. Geary RC. The contiguity ratio and statistical mapping. *The Incorporated Statistician.* 1954;5(3):115-146.
30. Su Y, Zhou A, Xia X, Li W, Sun Z. Quantitative prediction of protein–protein binding affinity with a potential of mean force considering volume correction. *Protein Sci.* 2009;18(12):2550-2558.
31. Martínez-López Y, Marrero-Ponce Y, Barigye SJ, et al. When global and local molecular descriptors are more than the sum of its parts: Simple, But Not Simpler? *Mol Divers.* 2019:1-20.
32. Terán JE, Marrero-Ponce Y, Contreras-Torres E, et al. Tensor Algebra-based Geometrical (3D) Biomacro-Molecular Descriptors for Protein Research: Theory, Applications and Comparison with other Methods. *Sci Rep.* 2019;9(1):1-15.
33. Valdés-Martiní JR, Marrero-Ponce Y, García-Jacas CR, et al. QuBiLS-MAS, open source multi-platform software for atom-and bond-based topological (2D) and chiral (2.5 D) algebraic molecular descriptors computations. *J Cheminformatics.* 2017;9(1):35.
34. Aguilera-Mendoza L, Marrero-Ponce Y, Beltran JA, Tellez Ibarra R, Guillen-Ramirez HA, Brizuela CA. Graph-based data integration from bioactive peptide databases of pharmaceutical interest: toward an organized collection enabling visual network analysis. *Bioinformatics.* 2019;35(22):4739-4747.
35. Eriksson L, Jaworska J, Worth AP, Cronin MT, McDowell RM, Gramatica P. Methods for reliability and uncertainty assessment and for applicability evaluations of classification-and regression-based QSARs. *Environ Health Perspect.* 2003;111(10):1361-1375.
36. Veltri D, Kamath U, Shehu A. Deep learning improves antimicrobial peptide recognition. *Bioinformatics.* 2018;34(16):2740-2747.
37. Thomas S, Karnik S, Barai RS, Jayaraman VK, Idicula-Thomas S. CAMP: a useful resource for research on antimicrobial peptides. *Nucleic Acids Res.* 2009;38(suppl_1):D774-D780.

38. Ghandi M, Lee D, Mohammad-Noori M, Beer MA. Enhanced regulatory sequence prediction using gapped k-mer features. *PLoS Comput Biol.* 2014;10(7):e1003711.