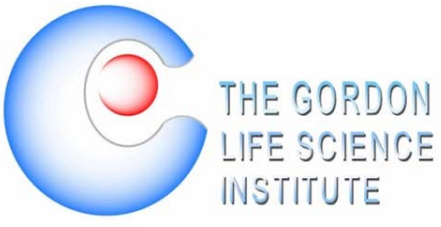
Pse-in-One: a web server for generating various modes of pseudo components of DNA, RNA, and protein sequences

Manual of stand-alone program of Pse-in-One

2014-1-5

Home-page: http://bioinformatics.hitsz.edu.cn/Pse-in-One/

Stand-alone source code repo: http://github.com/liufule12/Pse-in-One



Contents

1. Introduction of Pse-in-One

The **Pse-in-One** web server is able to generate totally 28 different features, including 14 features for DNA sequences (**Table 1**), 6 features for RNA sequences (**Table 2**), and 8 features for protein sequences (**Table 3**). All these features can be deemed as different pseudo components.

To the best of our knowledge, **Pse-in-One** is so far the first web server that can generate all the possible pseudo components for DNA, RNA, and protein sequences, and even those defined by users themselves, and hence it is extremely flexible.

In order to handle large dataset, the stand-alone program of **Pse-in-One** is given, which is more powerful than the Pse-in-One web server, and will be introduced in the following parts of this manual.

2. Installation

The **Pse-in-One** package can be run on Linux, Mac, and Windows systems.
Download the package from http://bioinformatics.hitsz.edu.cn/Pse-in-One/download and extract it to a directory, for example, "~/usr".
To execute the **Pse-in-One** in command line environment, navigate to the "~/usr/Pse-in-One-1.0/Pse-in-One" directory and you will find three python scripts, namely "kmer.py", "acc.py" and "pse.py". The "kmer.py" is used for calculating the features in the category nucleic acid composition or amino acid composition; The "acc.py" is used for calculating the features in autocorrelation category. The "pse.py" is used for calculating the features in the category pseudo nucleotide composition or pseudo amino acid composition.

3. Input/Output formats

3.1. Input format

The input file should be a valid FASTA format that consists of a single initial line beginning with a greater-than symbol (">") in the first column, followed by lines of sequence data. The words right after the ">" symbol in the single initial line are optional and only used for the purpose of identification and description.

3.2. Output format

The output file formats support three choices that are suitable for downstream computational analyses, such as machine learning. The first and the default choice is the tab format. In this format, all data is separated by TABs. The second one is the LIBSVM's sparse data format. For this format, each line contains an instance and is ended by a '\n' character, like <label> <index1>:<value1> <index2>:<value2> ... . The <label> is a category label of the sequence. The pair <index>:<value> gives a feature (attribute) value: <index> is an integer starting from 1 and <value> is a real number. The third output format is the csv format. This format is similar to the tab format. The only difference is the separation characters between data are commas.

3.3. Physicochemical Properties Selection

The Physicochemical Properties Selection file is a text file that contains a list of property names used for generating the features in categories: autocorrelation, pseudo nucleotide composition/ pseudo amino acid composition. For example, if you want to use the “Rise”, “Tilt” and “Shift” of DNA dinucleotide for calculating, the Physicochemical Properties Selection file should be written as follows:

```
Rise
Tilt
Shift
```

After saving this file as “propChosen.txt” and specifying it using the command “-i propChosen.txt”, or just “I propChosen.txt”, the above three properties will be used in calculations. Meanwhile, you can also use the command “-a True” to select all the built-in physicochemical properties for the corresponding sequence type, which can be selected by using parameter DNA, RNA or PROTEIN.

The complete lists of physicochemical properties for DNA, RNA and protein sequences used in the stand-alone program are provided in **Table 4-7**.

3.4. User-defined Physicochemical Properties

In the user-defined physicochemical index files, each index should be represented in three lines. The first line must start with a greater-than symbol (">") in the first column. The words right after the ">" symbol in the single initial line are optional and only used for the purpose of identification and description of the index. The second line lists the names of the sequence compositions (i.e. amino acids, nucleotides, dinucleotides, or trinucleotides, etc), which should be sorted in the alphabet order, such as 'A' 'C' ... 'AA' 'AC'. All the elements in this line should be separated by "tab". The corresponding values of these sequence compositions are listed in the third line, which are separated by TAB.

For example, if you defined a physicochemical property “user_property”, the user-defined physicochemical index file should be written as follows,

```
> user_property
A    C    …   AA AC …
0.21      0.12      …  0.37      0.15      …
```

After saving this file as “user_defined.txt” and specifying it using the command “-e user_defined.txt”, or just “E user_defined.txt”, the properties defined by user will be used in calculations.

4. Commands

4.1 Command line parameters for kmer.py

Options	Interpretations
inputfile	The input file in FASTA format.
outputfile	The output file stored results.
{1,2,3,4,5,6}	The k value of kmer.

{DNA, RNA, PROTEIN}	The sequence type.
-h, --help	show this help message and exit.
-r {1,0}	Whether consider the reverse complement or not. 1 means True, 0 means False. (default = 0)
-f {tab, svm, csv}	The output format. (default = tab) tab -- Simple format, delimited by TAB. svm -- The LIBSVM training data format. csv -- The format that can be loaded into a spreadsheet program.

4.2 Command line parameters for acc.py

Options	Interpretations
inputfile	The input file, in FASTA format.
outputfile	The output file stored results.
lag	The value of lag.
{DNA, RNA, PROTEIN}	The sequence type.
method	The method name of autocorrelation.
-h, --help	show this help message and exit.
-i I	The index file user chosen.
-e E	The user-defined index file.
-a {True,False}	Choose all physicochemical indices or not. (default = False)
-f {tab,svm,csv}	The output format (default = tab). tab -- Simple format, delimited by TAB. svm -- The LIBSVM format. csv -- The format that can be loaded into a spreadsheet program.

4.3 Command line parameters for pse.py

Options	Interpretations
inputfile	The input file, in valid FASTA format.
outputfile	The outputfile stored results.
lamada	The value of lambda.
w	The value of weight.
{DNA, RNA, PROTEIN}	The sequence type.
method	The method name of pseudo components.
-h, --help	show this help message and exit.
-i I	The index file user chosen.
-k K	The value of kmer, it works only with PseKNC method.
-e E	The user-defined index file, this parameter only needs to be set for PC-PseDNC-General, PC-PseTNC-General, SC-PseDNC-General, SC-PseTNC-General, PC-PseAAC-General or SC-PseAAC-General.
-a {True, False}	Choose all physicochemical indices or not. (default =

	False)
-f {tab, svm, csv}	The output format (default = tab). tab -- Simple format, delimited by TAB. svm -- The LIBSVM format. csv -- The format that can be loaded into a spreadsheet program.

4.4 Examples

For user's convenience, some examples of how to process a query sequence using command line are given below.

Example 1: Calculate the kmer composition feature vector of the query sequence and output the results in LIBSVM format.

```
kmer.py test.txt output_kmer.txt 2 DNA -f svm
```

After running the above command, the following results will be found in "output_kmer.txt" file.

```
0 1:0.023 2:0.034 3:0.053 4:0.023 5:0.045 6:0.086 7:0.143 8:0.06 9:0.049 10:0.15
11:0.124 12:0.049 13:0.015 14:0.064 15:0.053 16:0.03
```

Example 2: Calculate the auto covariance feature vector of the query sequence and output the results in LIBSVM format.

```
acc.py test.txt output_acc.txt 3 DNA TAC -a True -f svm
```

After running the above command, the following results will be found in "output_acc.txt" file.

```
0 1:-0.057 2:0.057 3:0.647 4:0.381 5:0.057 6:0.057 7:-0.051 8:-0.06 9:0.021
10:0.021 11:0.379 12:0.374 13:0.033 14:-0.011 15:0.413 16:0.019 17:-0.009
18:-0.009 19:-0.024 20:0.032 21:0.105 22:0.105 23:0.021 24:0.024 25:-0.008
26:-0.056 27:0.09 28:-0.088 29:-0.056 30:-0.056 31:-0.011 32:-0.008 33:-0.002
34:-0.002 35:-0.087 36:-0.085
```

Example 3: Calculate the PseDNC feature vector of the query sequence and output the results in CSV format.

```
pse.py test.txt output_pse.csv 3 0.2 DNA PseDNC
```

After running the above command, the following results will be found in "output_pse.csv" file.

```
0.01,0.016,0.024,0.01,0.021,0.04,0.066,0.028,0.023,0.069,0.057,0.023,0.007,0.02
9,0.024,0.014,0.217,0.152,0.17
```

Example 4: Calculate the PC-PseDNC-General feature vector of the query sequence using user-defined physicochemical index file and output the results in the CSV

format.

```
pse.py test.txt output_pse2.csv 3 0.2 DNA PC-PseDNC-General -e user_indices.txt -f csv
```

After running the above command, the following results will be found in "outut_pse2.csv" file.

```
0.011,0.016,0.025,0.011,0.021,0.041,0.068,0.028,0.023,0.071,0.059,0.023,0.007,
0.03,0.025,0.014,0.213,0.153,0.161
```

The content of the file "test.txt" is listed as follow:

```
>misc_ppid_8090
CTTCGCCAGCCACTCTTAGTCCGCCAGCGCGTGCGGCGGAGGCCGAGC
GTCTCTATGATCCTGGCTTCTGGCAACGTCATCGTCACGCGCCGGATCC
AACCCCCAACCACTTTAGCCAGCTCTAGAGGCGCGCGTGGCCGGGACG
GAAGTGCGCGCGGGTGTCGCCGGGAGTGCGCGCTCCTCTGGCTGACG
GGCGGGCCGGGCATGCGCCGCGGGCGTTTTGGCGGGAAGCGCGGGGC
GGGCCGGACAATGAGAGTGTCCGCCTCC
```

The content of the file "user_indices.txt" is listed as follow:

```
>user_defined_property
AA  AC  AG  AT  CA  CC  CG  CT  GA  GC  GG  GT  TA  TC  TG  TT
0.063   1.502   0.783   1.071   -1.376  0.063   -1.664  0.783   -
0.081   -0.081  0.063   1.502   -1.233  -0.081  -1.376  0.063
```

Table 1. 14 features of DNA sequences calculated by **PseDAC-General.**

Category	Feature	Description
Nucleic acid Composition	Kmer	Basic kmer (1)
	RevKmer	Reverse compliment kmer (2,3)
Autocorrelation	DAC	Dinucleotide-based auto covariance (4,5)
	DCC	Dinucleotide-based cross covariance (4,5)
	DACC	Dinucleotide-based auto-cross covariance (4,5)
	TAC	Trinucleotide-based auto covariance (5)
	TCC	Trinucleotide-based cross covariance (5)
	TACC	Trinucleotide-based auto-cross covariance (5)
Pseudo nucleotide composition	PseDNC	Pseudo dinucleotide composition (6)
	PseKNC	Pseudo k-tupler nucleotide composition (7,8)
	PC-PseDNC-General	General parallel correlation pseudo dinucleotide composition (9)
	PC-PseTNC-General	General parallel correlation pseudo trinucleotide composition (9)
	SC-PseDNC-General	General series correlation pseudo dinucleotide composition (9)
	SC-PseTNC-General	General series correlation pseudo trinucleotide composition (9)

Table 2. 6 features of RNA sequences calculated by **PseRAC-General.**

Category	Feature	Description
Nucleic acid composition	Kmer	Basic kmer (10)
Autocorrelation	DAC	Dinucleotide-based auto covariance (4,5,11)
	DCC	Dinucleotide-based cross covariance (4,5,11)
	DACC	Dinucleotide-based auto-cross covariance (4,5,11)
Pseudo nucleotide	PC-PseDNC-	General parallel correlation pseudo dinucleotide

composition	General SC-PseDNC-General	composition (4,12) General series correlation pseudo dinucleotide composition (4,12)

Table 3. 8 features of protein sequences calculated by **PseAAC-General.**

Category	Feature	Description
Amino acid composition	Kmer	Basic kmer (13)
Autocorrelation	AC	Auto covariance (5,11)
	CC	Cross covariance (5,11)
	ACC	Auto-cross covariance (5,11)
Pseudo amino acid composition	PC-PseAAC	Parallel correlation pseudo amino acid composition (14)
	SC-PseAAC	Series correlation pseudo amino acid composition (15)
	PC-PseAAC-General	General parallel correlation pseudo amino acid composition (14,16)
	SC-PseAAC-General	General series correlation pseudo amino acid composition (15,16)

Table 4. The names of the 148 physicochemical indices for dinucleotides.

Base stacking	Protein induced deformability	B-DNA twist
Propeller twist	Duplex stability:(freeenergy)	Duplex tability(disruptenergy)
Protein DNA twist	Stabilising energy of Z-DNA	Aida_BA_transition
Breslauer_dS	Electron_interaction	Hartman_trans_free_energy
Lisser_BZ_transition	Polar_interaction	SantaLucia_dG
Sarai_flexibility	Stability	Stacking_energy
Sugimoto_dS	Watson-Crick_interaction	Twist
Shift	Slide	Rise
Twist stiffness	Tilt stiffness	Shift_rise
Twist_shift	Enthalpy1	Twist_twist
Shift2	Tilt3	Tilt1
Slide (DNA-protein complex)1	Tilt_shift	Twist_tilt
Roll_rise	Stacking energy	Stacking energy1
Propeller Twist	Roll11	Rise (DNA-protein complex)
Roll2	Roll3	Roll1
Slide_slide	Enthalpy	Shift_shift
Flexibility_slide	Minor Groove Distance	Rise (DNA-protein complex)1
Roll (DNA-protein complex)1	Entropy	Cytosine content
Major Groove Distance	Twist (DNA-protein complex)	Purine (AG) content
Tilt_slide	Major Groove Width	Major Groove Depth
Free energy6	Free energy7	Free energy4
Free energy3	Free energy1	Twist_roll
Flexibility_shift	Shift (DNA-protein complex)1	Thymine content
Tip	Keto (GT) content	Roll stiffness
Entropy1	Roll_slide	Slide (DNA-protein complex)
Twist2	Twist5	Twist4
Tilt (DNA-protein complex)1	Twist_slide	Minor Groove Depth
Persistance Length	Rise3	Shift stiffness
Slide3	Slide2	Slide1
Rise1	Rise stiffness	Mobility to bend towards minor

		groove
Dinucleotide GC Content	A-philicity	Wedge
DNA denaturation	Bending stiffness	Free energy5
Breslauer_dG	Breslauer_dH	Shift (DNA-protein complex)
Helix-Coil_transition	Ivanov_BA_transition	Slide_rise
SantaLucia_dH	SantaLucia_dS	Minor Groove Width
Sugimoto_dG	Sugimoto_dH	Twist1
Tilt	Roll	Twist7
Clash Strength	Roll_roll	Roll (DNA-protein complex)
Adenine content	Direction	Probability contacting nucleosome core
Roll_shift	Shift_slide	Shift1
Tilt4	Tilt2	Free energy8
Twist (DNA-protein complex)1	Tilt_rise	Free energy2
Stacking energy2	Stacking energy3	Rise_rise
Tilt_tilt	Roll4	Tilt_roll
Minor Groove Size	GC content	Inclination
Slide stiffness	Melting Temperature1	Twist3
Tilt (DNA-protein complex)	Guanine content	Twist6
Major Groove Size	Twist_rise	Rise2
Melting Temperature	Free energy	Mobility to bend towards major groove
Bend		

Table 5. The names of the 12 physicochemical indices for trinucleotides.

Bendability (DNAse)	Bendability (consensus)	Trinucleotide GC Content
Consensus_roll	Consensus-Rigid	Dnase I
MW-Daltons	MW-kg	Nucleosome
Nucleosome positioning	Dnase I-Rigid	Nucleosome-Rigid

Table 6. The names of the 6 physicochemical indices for dinucleotides.

Twist	Tilt	Roll
Shift	Slide	Rise

Table 7. The names of the 22 physicochemical indices for dinucleotides.

Shift (RNA)	Hydrophilicity (RNA)
Hydrophilicity (RNA)	GC content
Purine (AG) content	Keto (GT) content
Adenine content	Guanine content
Cytosine content	Thymine content
Slide (RNA)	Rise (RNA)
Tilt (RNA)	Roll (RNA)
Twist (RNA)	Stacking energy (RNA)
Enthalpy (RNA)	Entropy (RNA)
Free energy (RNA)	Free energy (RNA)
Enthalpy (RNA)	Entropy (RNA)

References

1. Lee, D., Karchin, R. and Beer, M.A. (2011) Discriminative prediction of mammalian enhancers from DNA sequence. *Genome research*, **21**, 2167-2180.
2. Noble, W.S., Kuehn, S., Thurman, R., Yu, M. and Stamatoyannopoulos, J. (2005) Predicting the in vivo signature of human gene regulatory sequences. *Bioinformatics*, **21 Suppl 1**, i338-343.
3. Gupta, S., Dennis, J., Thurman, R.E., Kingston, R., Stamatoyannopoulos, J.A. and Noble, W.S. (2008) Predicting human nucleosome occupancy from primary sequence. *PLoS computational biology*, **4**, e1000134.
4. Friedel, M., Nikolajewa, S., Suhnel, J. and Wilhelm, T. (2008) DiProDB: a database for dinucleotide properties. *Nucleic Acids Res*, **37**, D37-D40.
5. Dong, Q., Zhou, S. and Guan, J. (2009) A new taxonomy-based protein fold recognition approach based on autocross-covariance transformation. *Bioinformatics*, **25**, 2655-2662.
6. Chen, W., Feng, P.M., Lin, H. and Chou, K.C. (2013) iRSpot-PseDNC: identify recombination spots with pseudo dinucleotide composition. *Nucleic Acids Res*, **41**, e68.
7. Guo, S.H., Deng, E.Z., Xu, L.Q., Ding, H., Lin, H., Chen, W. and Chou, K.C. (2014) iNuc-PseKNC: a sequence-based predictor for predicting nucleosome positioning in genomes with pseudo k-tuple nucleotide composition. *Bioinformatics*, **30**, 1522-1529.
8. Lin, H., Deng, E.-Z., Ding, H., Chen, W. and Chou, K.-C. (2014) iPro54-PseKNC: a sequence-based predictor for identifying sigma-54 promoters in prokaryote with pseudo k-tuple nucleotide composition. *Nucleic Acids Res*, **42**, 12961-12972.
9. Chen, W., Lei, T.Y., Jin, D.C., Lin, H. and Chou, K.C. (2014) PseKNC: a flexible web server for generating pseudo K-tuple nucleotide composition. *Analytical biochemistry*, **456**, 53-60.
10. Wei, L., Liao, M., Gao, Y., Ji, R., He, Z. and Zou, Q. (2014) Improved and Promising Identification of Human MicroRNAs by Incorporating a High-quality Negative Set. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* **11**, 192-201.
11. Guo, Y., Yu, L., Wen, Z. and Li, M. (2008) Using support vector machine combined with auto covariance to predict protein-protein interactions from protein sequences. *Nucleic Acids Research*, **36**, 3025-3030.
12. Chen, W., Zhang, X., Brooker, J., Lin, H., Zhang, L. and Chou, K.-C. (2014) PseKNC-General: a cross-platform package for generating various modes of pseudo nucleotide compositions. *Bioinformatics*, DOI: 10.1093/bioinformatics/btu1602.
13. Liu, B., Wang, X., Lin, L., Dong, Q. and Wang, X. (2008) A Discriminative Method for Protein Remote Homology Detection and Fold Recognition Combining Top-n-grams and Latent Semantic Analysis. *BMC Bioinformatics*, **9**, 510.
14. Chou, K.-C. (2001) Prediction of protein cellular attributes using pseudo-amino-acid-composition. *PROTEINS: Structure, Function, and Genetics*, **43**, 246-255.
15. Chou, K.-C. (2005) Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes. *Bioinformatics*, **21**, 10-19.
16. Kawashima, S., Pokarowski, P., Pokarowska, M., Kolinski, A., Katayama, T. and Kanehisa, M. (2008) AAindex: amino acid index database, progress report 2008. *Nucleic Acids Res*, **36**, D202-D205.