PS-COILS

Piero Fariselli

April 6, 2008

1 PSCOILS program

PSCOILS is a simple evolution of COILS [1] and PCOILS [2] programs. It uses the same parameters that were developed for COILS and exploits both sequence and evolutionary information (in the form of sequence profiles). For the details of the parameter construction please see [1, 3].

2 Background

Here we summarize the basic algorithms behind COILS, PCOILS and PSCOILS in order to highlight the basic differences.

2.1 COILS

The coils program is based on scoring tables $S^h(a)$ that are used to compute the probability score for each segment of a protein sequence (see [1]), and some parameters that define Gaussian probabilities. Then the main parameters are:

- μ_{cc}, μ_g the average scoring values of the coiled-coil and globular protein sets;
- σ_{cc}, σ_g the standard deviation of the scoring values for the coiled-coil and globular protein sets;
- $S^h(a)$ = the score for the residue type a in the heptad position h (from 1 to 7).

So that the probability of a coiled-coil segment of length W starting at position i in a given sequence is computed as

$$Pr_i = \frac{G_{cc}}{G_{cc} + c \cdot G_g} \tag{1}$$

where c is the bias for the most abundant globular class (g) and G_{cc} and G_g are defined as

$$G_{cc} = \frac{1}{\sqrt{2\sigma_{cc}}} e^{-\frac{(x_i-\mu_{cc})^2}{\sigma_{cc}^2}} \tag{2}$$

$$G_g = \frac{1}{\sqrt{2\sigma_g}} e^{-\frac{(x_i-\mu_g)^2}{\sigma_g^2}} \tag{3}$$

The score x_i is computed using the matrix $S^h(a)$ ([1]) along the segment W starting at position i as

$$x_i = (\prod_{h=1}^{W} f(a_{i+h}, h)^{e_h})^{1/N} \tag{4}$$

where e_h is the exponential weight of the position h (if not weighted is simply $e_h = 1$) and N is the normalization factor $N = \sum_{h=1}^{W} e_h$. The function f is in the case of COILS program is simply

$$f(a_{i+h}, h) = S^h(a_{i+h}) \tag{5}$$

where $S^h(a_{i+h})$ is the element of the COILS scoring table accounting for the residue type a_{i+h} in the h^{th} heptad position.

2.2 PCOILS

For sake of clarity we have to mention that this is *our implementation* of PCOILS, and we cannot guarantee that the original PCOILS program works in the same way, since the authors does not show the explicit algorithm [2]. However, in case of our PCOILS, all the machinery described above still remains untouched, with the exception of function f (Eq. 5). Since we are dealing with evolutionary information computed from a given multiple alignment, instead of the single-sequence s we work with the profile $P_k(a)$, that represents the frequency of residue a in position k of the alignment. In this case the PCOILS score is still defined by an equation similar to Eq.4, but with the new function:

$$x_i = (\prod_{h=1}^{W} f(S, P, h)^{e_h})^{1/N} \tag{6}$$

and

$$f(S, P, h) =< S^h, P_{i+h} >= \sum_{a \in \{Residues\}} S^h(a) \cdot P_{i+h}(a) \tag{7}$$

2.3 PSCOILS

PSCOILS combines the sequence and the profile information using a linear weighting scheme, namely $\lambda COILS + (1-\lambda)PCOILS$ with λ in the range of $[0,1]$ the only variation with respect to the previous algorithm is again the f equation (5 and 7). We then have as before

$$x_i = (\prod_{h=1}^{W} f(S,P,h,\lambda)^{e_h})^{1/N} \tag{8}$$

and

$$f(S,P,h,\lambda) = \lambda S^h(a_{i+h}) + (1-\lambda) < S^h, P_{i+h} > \tag{9}$$

where the meaning of $S^h(a_{i+h})$ and $< S^h, P_{i+h} >$ are the same as in 5 and 7, respectively. No attempt have been made to optimize λ but in the current stage it has been set to $1/2$. In this case the sequence and the profile are equally weighted.

3 PSCOILS usage

The program `psCoils.py` is written in **python** and it is distributed under the *GPL License*,

```
psCoils.py  is free software and can be redistributed  and/or
modified under the terms of the GNU General Public License as
published by the Free Software Foundation.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.
```

`psCoils.py` incorporates both COILS and our version of PCOILS, since the original one [2] were not explicitly described. Just typing the program name `psCoils.py` on the standard output will appear

```
USAGE:  ./psCoils.py -f fasta  -p profile [options]
Options:
 -W 14/21/28 # one of the possible windows (default 21)
 -w w/uw  # weight or unweight default=uw
 -l T/F  # print prediction labels when set T (default) if P>0.5
 -L [0.0,1.0] lambda value. It will be used only if both -f and -p are set
```

The parameters are the standard one defined for COILS, such as:

- `-W`: the input window that is set to 21 as default;
- `-w`: can set the heptad weighted `w`, or unweighted `uw` periodicity
- `-l`: set the label printing in the output options (unless -l F is set)
- `-L`: this set the linear weighting scheme between sequence and profile, according to λ Sequence +(1-λ) Profile.

`psCoils.py` can be used as COILS, PCOILS or PSCOILS depending on the different input provided, such as:

- As **COILS**: use only `-f fasta` option.
- As **PCOILS**: use only `-p profile` option.
- As **PSCOILS**: use both `-f fasta -p profile` options.

A typical output is

```
 Pos A Hep Score   Prob    Gcc     Gg    Pred (Loop=L Coiledcoil=C)
   1 C a   0.480   0.000   0.000   0.927 L
   2 M a   0.804   0.000   0.001   4.596 L
   3 S b   0.804   0.000   0.001   4.596 L
..
```

where `Pos` is the sequence position, `A` contains the residue sequence (only an `x` if PCOILS is used), `Prob` is the computed probability as described above, `Gcc` and `Gg` are Gaussian values obtained as described above and `Pred` is the label associated at `Prob` if its value is greater of 0.5.

4 Availability

PSCOILS is available as stand-alone python program at http://www.biocomp.unibo.it/piero/PSCOILS/download/ or as python package as part of the `plone4bio` effort at http://www.plone4bio.org.

References

[1] Lupas AN, Van Dyke M and Stock J. (1991) "Predicting coiled coils from protein sequences". *Science*, 252, 1162-1164.

[2] Gruber M, Soding J and Lupas AN (2005) "REPPER-repeats and their periodicities in fibrous proteins", *Nucleic Acids Res.* 33:239243.

[3] Gruber M, Soding J and Lupas AN (2006) "Comparative analysis of coiled-coil prediction methods". *J Struct Biol*, 155:140-145.