SDFファイルから特徴ベクトルを計算する

2020年9月1日

目 次

1 Quick start

この冊子では，化合物の特徴ベクトルを計算するプログラムについて説明する．入力の化合物は，SDFと呼ばれる標準フォーマットのファイルで与えられ，ひとつのSDFファイルに複数個の化合物が含まれでもよい．特徴ベクトルは，本プロジェクト独自のFVフォーマットで出力される．これらのフォーマットを含めて詳細を次節で説明することとし，ここではまず利用方法を示す．

- **環境確認**
  ISO C++ 2011標準に対応するC++コンパイラーがあれば問題ないと考えられる．Linux Mint 18 & 19, コンパイラー g++ ver 5 & 7で確認したが，g++がインストールされていない場合は，次のようにインストールできる．
  `$ sudo apt install g++`

- **コンパイル**
  `$ g++ -std=c++11 -o fv4_in_ex fv4_in_ex.cpp`
  (g++ 7の場合は `-std=c++11` を省略できる．)

- **実行**
  `$ ./fv4_in_ex input.sdf output.csv`
  `input.sdf` で入力のSDFファイル，`output.csv` で出力の特徴ベクトルファイルを指定する．両方ともテキストエディターで内容を確認できる．例えば，
  `$ ./fv4_in_ex sample1.sdf sample1.csv`

2 SDFフォーマット

本プログラムの入力ファイルは，SDF (Structure Data File) という業界標準的なフォーマットを採用している．`https://www.chem-station.com/blog/2012/04/sdf.html`などの解説が分かりやすい．さらに，正確な定義書として，公式資料 `http://help.accelrysonline.com/ulm/onelab/1.0/content/ulm_pdfs/direct/reference/ctfileformats2016.pdf` を参照するとよい．

例として，添付のsample1.sdf (`https://pubchem.ncbi.nlm.nih.gov/compound/128703`) を以下に示す．

SDFフォーマットファイルの例: sample1.sdf

```
128703
  -OEChem-02061913062D

 24 23  0     1  0  0  0  0  0999 V2000
    6.0010   -1.2500    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
    5.1350    1.2500    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
    6.8671    1.2500    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
    4.2690   -1.2500    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
```

```
   7.7331   -1.2500    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
   2.5369   -0.2500    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
   8.5991    0.2500    0.0000 N   0  0  0  0  0  0  0  0  0  0  0  0
  12.9292   -0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
  12.0632    0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
  13.7953    0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
  11.1972   -0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
  14.6613   -0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
  10.3312    0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
   6.0010   -0.2500    0.0000 C   0  0  1  0  0  0  0  0  0  0  0  0
  15.5273    0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
   5.1350    0.2500    0.0000 C   0  0  1  0  0  0  0  0  0  0  0  0
   6.8671    0.2500    0.0000 C   0  0  2  0  0  0  0  0  0  0  0  0
   9.4651   -0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
  16.3933   -0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
   4.2690   -0.2500    0.0000 C   0  0  1  0  0  0  0  0  0  0  0  0
   7.7331   -0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
  17.2594    0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
   3.4030    0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
  18.1254   -0.2500    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
14  1  1  1  0  0  0
16  2  1  1  0  0  0
17  3  1  1  0  0  0
20  4  1  6  0  0  0
 5 21  2  0  0  0  0
 6 23  1  0  0  0  0
 7 18  1  0  0  0  0
 7 21  1  0  0  0  0
 8  9  1  0  0  0  0
 8 10  1  0  0  0  0
 9 11  1  0  0  0  0
10 12  1  0  0  0  0
11 13  1  0  0  0  0
12 15  1  0  0  0  0
13 18  1  0  0  0  0
14 16  1  0  0  0  0
14 17  1  0  0  0  0
15 19  1  0  0  0  0
16 20  1  0  0  0  0
17 21  1  0  0  0  0
```

```
 19 22  1  0  0  0  0
 20 23  1  0  0  0  0
 22 24  1  0  0  0  0
M  END
```

3 FVフォーマット

本プログラムの出力ファイルは，独自の FV (Feature Vector, 特徴ベクトル) フォーマットを採用している．このテキストファイルは，カンマで区切った CSV ファイルと同様のフォーマットを持ち，拡張子を csv にすることによって Excel などのいわゆる表計算ソフトで開くことができる．具体的に，一行目には特徴ベクトルの構成要素を示し，二行目以降の各行には特徴ベクトルの数値データが記入されている．具体例として，`sample1.sdf` に対して計算して得られた `sample1.csv` を示す．それぞれの構成要素はそのあとに説明する．

FV ファイルの例: sample1.csv (注: \\は実際に改行しないことを示す．)

```
CID,n,M,C_in,C_ex,S_in,S_ex,N_in,N_ex,H,S1S_in,S1S_ex,C1S_in,C1S_ex,\\
C2S_in,C2S_ex,C1N_in,C1N_ex,#degree1_in,#degree1_ex,#degree2_in,\\
#degree2_ex,#degree3_in,#degree3_ex,#degree4_in,#degree4_ex,\\
#double_bond_in,#double_bond_ex,#triple_bond_in,#triple_bond_ex,\\
Diameter,Bc_121_in,Bc_121_ex,Bc_122_in,Bc_122_ex,Bc_123_in,Bc_123_ex,\\
Bc_131_in,Bc_131_ex,Bc_132_in,Bc_132_ex,Bc_141_in,Bc_141_ex,\\
Bc_221_in,Bc_221_ex,Bc_222_in,Bc_222_ex,Bc_223_in,Bc_223_ex,\\
Bc_231_in,Bc_231_ex,Bc_232_in,Bc_232_ex,Bc_241_in,Bc_241_ex,\\
Bc_331_in,Bc_331_ex,Bc_332_in,Bc_332_ex,Bc_341_in,Bc_341_ex,\\
Bc_441_in,Bc_441_ex,2-branch_height,2-branch_leaf_number
128703,24,130.833,14,3,0,6,1,0,0,0,5,0,1,2,0,12,3,0,7,10,2,5,0,0,0,0,1,0,0,\\
0.75,0,2,0,0,0,0,0,0,4,0,1,0,0,9,1,0,0,0,0,1,1,0,0,0,0,4,0,0,0,0,0,0,0,1,2
```

構成要素の説明

- `CID`
  PubChem (`https://pubchem.ncbi.nlm.nih.gov/`) における CID．例えば `sample1.sdf` にある化合物は，`https://pubchem.ncbi.nlm.nih.gov/compound/128703` になる．
- `n`
  原子の数，ただし水素 `H` を除く．
- `M`
  独自定義の平均分子質量 $M = \frac{1}{n}\sum_{\mathtt{a}} \lfloor 10 \cdot \mathrm{mass}(\mathtt{a}) \rfloor$.

- C_in,O_in,N_in
  それぞれ内部原子の数.
- C_ex,O_ex,N_ex
  それぞれ外部原子の数.
- H
  原子の数.
- C1O_in, C2O_in, C1N_in, C1C_in
  それぞれの内部パスの数. 例えば C1O_in は，C と　O の一重結合による内部パス，C2O_in は，C と O の二重結合による内部パスを表す.
- C1O_ex, C2O_ex, C1N_ex, C1C_ex
  それぞれの外部パスの数. 例えば C1O_in は，C と　O の一重結合による外部パス，C2O_in は，C と O の二重結合による外部パスを表す.
- #degree1_in,#degree2_in,#degree3_in,#degree4_in
  それぞれ次数 (価数) の内部原子の数.
- #degree1_ex,#degree2_ex,#degree3_ex,#degree4_ex
  それぞれ次数 (価数) の外部原子の数.
- #double_bond_in,#triple_bond_in
  それぞれ内部二重結合と内部三重結合の数.
- #double_bond_ex,#triple_bond_ex
  それぞれ外部二重結合と外部三重結合の数.
- Diameter
  直径/n.
- Bc_xyz_in
  内部次数構成 (x, y, z), ただし $x \leq y$ は z-重結合の両端点の次数を表す.
- Bc_xyz_ex
  外部次数構成 (x, y, z), ただし $x \leq y$ は z-重結合の両端点の次数を表す.
- 2-branch_height
  2-branch-height bh_2. 別途 module 1 の論文を参照されたい.
- 2-branch-leaf_number
  2-branch-leaf-number bl_2. 別途 module 1 の論文を参照されたい.

4　計算プログラム

計算プログラム fv4_in_ex は，入力の SDF ファイルに対して，計算された特徴ベクトルを FV フォーマットで出力する．詳しい紹介は，本マニュアルの範囲を超えているので，別途 module 1 の論文を参照されたい．ここではプログラム利用上の注意事項を示す.

1. 原子の質量は，プログラムの中にハードコード仕様になっている．執筆の時点では，次のようになっているが，必要の場合，追加してご利用下さい．

対応している原子の質量 (function init_MassMap())

```
M["B"]  = 108;
M["C"]  = 120;
M["O"]  = 160;
M["N"]  = 140;
M["F"]  = 190;
M["Si"] = 280;
M["P"]  = 310;
M["S"]  = 320;
M["Cl"] = 355;
M["V"]  = 510;
M["Br"] = 800;
M["Cd"] = 1124;
M["I"]  = 1270;
M["Hg"] = 2006;
M["Pb"] = 2072;
M["Al"] = 269;
```

2. デバッグ出力をみるには，bool debug = true; にしてコンパイルして下さい．