NAME

TopologicalPharmacophoreAtomPairsFingerprints.pl - Generate topological pharmacophore atom pairs fingerprints for SD files

SYNOPSIS

TopologicalPharmacophoreAtomPairsFingerprints.pl SDFile(s)...

TopologicalPharmacophoreAtomPairsFingerprints.pl [--AromaticityModel *AromaticityModelType*] [--AtomPairsSetSizeToUse *ArbitrarySize* | *FixedSize*] [-a, --AtomTypesToUse *"AtomType1, AtomType2..."*] [--AtomTypesWeight *"AtomType1, Weight1, AtomType2, Weight2..."*] [--CompoundID *DataFieldName or LabelPrefixString*] [--CompoundIDLabel *text*] [--CompoundIDMode] [--DataFields *"FieldLabel1, FieldLabel2,..."*] [-d, --DataFieldsMode *All* | *Common* | *Specify* | *CompoundID*] [-f, --Filter *Yes* | *No*] [--FingerprintsLabelMode *FingerprintsLabelOnly* | *FingerprintsLabelWithIDs*] [--FingerprintsLabel *text*] [--FuzzifyAtomPairsCount *Yes* | *No*] [--FuzzificationMode *FuzzyBinning* | *FuzzyBinSmoothing*] [--FuzzificationMethodology *FuzzyBinning* | *FuzzyBinSmoothing*] [--FuzzFactor *number*] [-h, --help] [-k, --KeepLargestComponent *Yes* | *No*] [--MinDistance *number*] [--MaxDistance *number*] [-n, --NormalizationMethodology *None* | *ByHeavyAtomsCount* | *ByAtomTypesCount*] [--OutDelim *comma* | *tab* | *semicolon*] [--output *SD* | *FP* | *text* | *all*] [-o, --overwrite] [-q, --quote *Yes* | *No*] [-r, --root *RootName*] [--ValuesPrecision *number*] [-v, --VectorStringFormat *ValuesString, IDsAndValuesString* | *IDsAndValuesPairsString* | *ValuesAndIDsString* | *ValuesAndIDsPairsString*] [-w, --WorkingDir dirname] SDFile(s)...

DESCRIPTION

Generate topological pharmacophore atom pairs fingerprints [Ref 60-62, Ref 65, Ref 68] for *SDFile(s)* and create appropriate SD, FP or CSV/TSV text file(s) containing fingerprints vector strings corresponding to molecular fingerprints.

Multiple SDFile names are separated by spaces. The valid file extensions are *.sdf* and *.sd*. All other file names are ignored. All the SD files in a current directory can be specified either by **.sdf* or the current directory name.

Based on the values specified for --AtomTypesToUse, pharmacophore atom types are assigned to all non-hydrogen atoms in a molecule and a distance matrix is generated. A pharmacophore atom pairs basis set is initialized for all unique possible pairs within --MinDistance and --MaxDistance range.

```
Let:

P = Valid pharmacophore atom type

Px = Pharmacophore atom type x
Py = Pharmacophore atom type y

Dmin = Minimum distance corresponding to number of bonds between
       two atoms
Dmax = Maximum distance corresponding to number of bonds between
       two atoms
D = Distance corresponding to number of bonds between two atoms

Px-Dn-Py = Pharmacophore atom pair ID for atom types Px and Py at
           distance Dn

P = Number of pharmacophore atom types to consider
PPDn = Number of possible unique pharmacophore atom pairs at a distance Dn

PPT = Total number of possible pharmacophore atom pairs at all distances
      between Dmin and Dmax

Then:

PPD =  (P * (P - 1))/2 + P

PPT = ((Dmax - Dmin) + 1) * ((P * (P - 1))/2 + P)
    = ((Dmax - Dmin) + 1) * PPD

So for default values of Dmin = 1, Dmax = 10 and P = 5,

PPD =  (5 * (5 - 1))/2 + 5 = 15
PPT = ((10 - 1) + 1) * 15 = 150

The pharmacophore atom pairs bais set includes 150 values.

The atom pair IDs correspond to:
```

```
Px-Dn-Py = Pharmacophore atom pair ID for atom types Px and Py at
           distance Dn

For example: H-D1-H, H-D2-HBA, PI-D5-PI and so on
```

Using distance matrix and pharmacohore atom types, occurrence of unique pharmacohore atom pairs is counted. The contribution of each atom type to atom pair interaction is optionally weighted by specified --AtomTypesWeight before assigning its count to appropriate distance bin. Based on --NormalizationMethodology option, pharmacophore atom pairs count is optionally normalized. Additionally, pharmacohore atom pairs count is optionally fuzzified before or after the normalization controlled by values of --FuzzifyAtomPairsCount, --FuzzificationMode, --FuzzificationMethodology and --FuzzFactor options.

The final pharmacophore atom pairs count along with atom pair identifiers involving all non-hydrogen atoms, with optional normalization and fuzzification, constitute pharmacophore topological atom pairs fingerprints of the molecule.

For *ArbitrarySize* value of --AtomPairsSetSizeToUse option, the fingerprint vector correspond to only those topological pharmacophore atom pairs which are present and have non-zero count. However, for *FixedSize* value of --AtomPairsSetSizeToUse option, the fingerprint vector contains all possible valid topological pharmacophore atom pairs with both zero and non-zero count values.

Example of *SD* file containing topological pharmacophore atom pairs fingerprints string data:

```
... ...
... ...
$$$$
... ...
... ...
... ...
41 44  0  0  0  0  0  0  0  0999 V2000
 -3.3652    1.4499    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
... ...
2  3  1  0  0  0  0
... ...
M  END
>  <CmpdID>
Cmpd1

>  <TopologicalPharmacophoreAtomPairsFingerprints>
FingerprintsVector;TopologicalPharmacophoreAtomPairs:ArbitrarySize:Min
Distance1:MaxDistance10;54;NumericalValues;IDsAndValuesString;H-D1-H H
-D1-NI HBA-D1-NI HBD-D1-NI H-D2-H H-D2-HBA H-D2-HBD HBA-D2-HBA HBA-D2-
HBD H-D3-H H-D3-HBA H-D3-HBD H-D3-NI HBA-D3-NI HBD-D3-NI H-D4-H H-D...;
18 1 2 1 22 12 8 1 2 18 6 3 1 1 1 22 13 6 5 7 2 28 9 5 1 1 1 36 16 10 3
4 1 37 10 8 1 35 10 9 3 3 1 28 7 7 4 18 16 12 5 1 2 1

$$$$
... ...
... ...
```

Example of *FP* file containing topological pharmacophore atom pairs fingerprints string data:

```
#
# Package = MayaChemTools 7.4
# Release Date = Oct 21, 2010
#
# TimeStamp = Fri Mar 11 15:32:48 2011
#
# FingerprintsStringType = FingerprintsVector
#
# Description = TopologicalPharmacophoreAtomPairs:ArbitrarySize:MinDistance1:MaxDistance10
# VectorStringFormat = IDsAndValuesString
# VectorValuesType = NumericalValues
#
Cmpd1 54;H-D1-H H-D1-NI HBA-D1-NI HBD-D1-NI H-D2-H H-D2-HBA...;18 1 2...
Cmpd2 61;H-D1-H H-D1-NI HBA-D1-NI HBD-D1-NI H-D2-H H-D2-HBA...;5 1 2 ...
... ...
... ..
```

Example of CSV *Text* file containing topological pharmacophore atom pairs fingerprints string data:

```
"CompoundID","TopologicalPharmacophoreAtomPairsFingerprints"
"Cmpd1","FingerprintsVector;TopologicalPharmacophoreAtomPairs:Arbitrary
```

```
Size:MinDistance1:MaxDistance10;54;NumericalValues;IDsAndValuesString;H
-D1-H H-D1-NI HBA-D1-NI HBD-D1-NI H-D2-H H-D2-HBA H-D2-HBD HBA-D2-HBA H
BA-D2-HBD H-D3-H H-D3-HBA H-D3-HBD H-D3-NI HBA-D3-NI HBD-D3-NI H-D4...;
18 1 2 1 22 12 8 1 2 18 6 3 1 1 1 22 13 6 5 7 2 28 9 5 1 1 1 36 16 10 3
4 1 37 10 8 1 35 10 9 3 3 1 28 7 7 4 18 16 12 5 1 2 1"
... ...
... ...
```

The current release of MayaChemTools generates the following types of topological pharmacophore atom pairs fingerprints vector strings:

```
FingerprintsVector;TopologicalPharmacophoreAtomPairs:ArbitrarySize:Min
Distance1:MaxDistance10;54;NumericalValues;IDsAndValuesString;H-D1-H H
-D1-NI HBA-D1-NI HBD-D1-NI H-D2-H H-D2-HBA H-D2-HBD HBA-D2-HBA HBA-D2-
HBD H-D3-H H-D3-HBA H-D3-HBD H-D3-NI HBA-D3-NI HBD-D3-NI H-D4-H H-D4-H
BA H-D4-HBD HBA-D4-HBA HBA-D4-HBD HBD-D4-HBD H-D5-H H-D5-HBA H-D5-...;
18 1 2 1 22 12 8 1 2 18 6 3 1 1 1 22 13 6 5 7 2 28 9 5 1 1 1 36 16 10
3 4 1 37 10 8 1 35 10 9 3 3 1 28 7 7 4 18 16 12 5 1 2 1
```

```
FingerprintsVector;TopologicalPharmacophoreAtomPairs:FixedSize:MinDist
ance1:MaxDistance10;150;OrderedNumericalValues;ValuesString;18 0 0 1 0
0 0 2 0 0 1 0 0 0 0 22 12 8 0 0 1 2 0 0 0 0 0 0 0 0 18 6 3 1 0 0 0 1
0 0 1 0 0 0 0 22 13 6 0 0 5 7 0 0 2 0 0 0 0 0 28 9 5 1 0 0 0 1 0 0 1 0
0 0 0 36 16 10 0 0 3 4 0 0 1 0 0 0 0 0 37 10 8 0 0 0 0 1 0 0 0 0 0 0
0 35 10 9 0 0 3 3 0 0 1 0 0 0 0 0 28 7 7 4 0 0 0 0 0 0 0 0 0 0 0 18...
```

```
FingerprintsVector;TopologicalPharmacophoreAtomPairs:FixedSize:MinDist
ance1:MaxDistance10;150;OrderedNumericalValues;IDsAndValuesString;H-D1
-H H-D1-HBA H-D1-HBD H-D1-NI H-D1-PI HBA-D1-HBA HBA-D1-HBD HBA-D1-NI H
BA-D1-PI HBD-D1-HBD HBD-D1-NI HBD-D1-PI NI-D1-NI NI-D1-PI PI-D1-PI H-D
2-H H-D2-HBA H-D2-HBD H-D2-NI H-D2-PI HBA-D2-HBA HBA-D2-HBD HBA-D2...;
18 0 0 1 0 0 0 2 0 0 1 0 0 0 0 22 12 8 0 0 1 2 0 0 0 0 0 0 0 0 18 6 3
1 0 0 0 1 0 0 1 0 0 0 0 22 13 6 0 0 5 7 0 0 2 0 0 0 0 0 28 9 5 1 0 0 0
1 0 0 1 0 0 0 0 36 16 10 0 0 3 4 0 0 1 0 0 0 0
```

OPTIONS

--AromaticityModel *MDLAromaticityModel* | *TriposAromaticityModel* | *MMFFAromaticityModel* | *ChemAxonBasicAromaticityModel* | *ChemAxonGeneralAromaticityModel* | *DaylightAromaticityModel* | *MayaChemToolsAromaticityModel*

Specify aromaticity model to use during detection of aromaticity. Possible values in the current release are: *MDLAromaticityModel, TriposAromaticityModel, MMFFAromaticityModel, ChemAxonBasicAromaticityModel, ChemAxonGeneralAromaticityModel, DaylightAromaticityModel or MayaChemToolsAromaticityModel*. Default value: *MayaChemToolsAromaticityModel*.

The supported aromaticity model names along with model specific control parameters are defined in AromaticityModelsData.csv, which is distributed with the current release and is available under lib/data directory. Molecule.pm module retrieves data from this file during class instantiation and makes it available to method DetectAromaticity for detecting aromaticity corresponding to a specific model.

--AtomPairsSetSizeToUse *ArbitrarySize* | *FixedSize*

Atom pairs set size to use during generation of topological pharmacophore atom pairs fingerprints.

Possible values: *ArbitrarySize* | *FixedSize*; Default value: *ArbitrarySize*.

For *ArbitrarySize* value of --AtomPairsSetSizeToUse option, the fingerprint vector correspond to only those topological pharmacophore atom pairs which are present and have non-zero count. However, for *FixedSize* value of --AtomPairsSetSizeToUse option, the fingerprint vector contains all possible valid topological pharmacophore atom pairs with both zero and non-zero count values.

-a, --AtomTypesToUse *"AtomType1,AtomType2,..."*

Pharmacophore atom types to use during generation of topological phramacophore atom pairs. It's a list of comma separated valid pharmacophore atom types.

Possible values for pharmacophore atom types are: *Ar, CA, H, HBA, HBD, Hal, NI, PI, RA*. Default value [Ref 60-62] : *HBD,HBA,PI,NI,H*.

The pharmacophore atom types abbreviations correspond to:

```
HBD: HydrogenBondDonor
HBA: HydrogenBondAcceptor
PI :  PositivelyIonizable
NI : NegativelyIonizable
Ar : Aromatic
Hal : Halogen
H : Hydrophobic
```

```
RA : RingAtom
CA : ChainAtom
```

AtomTypes::FunctionalClassAtomTypes module is used to assign pharmacophore atom types. It uses following definitions [Ref 60-61, Ref 65-66]:

```
HydrogenBondDonor: NH, NH2, OH
HydrogenBondAcceptor: N[!H], O
PositivelyIonizable: +, NH2
NegativelyIonizable: -, C(=O)OH, S(=O)OH, P(=O)OH
```

--AtomTypesWeight *"AtomType1,Weight1,AtomType2,Weight2..."*

Weights of specified pharmacophore atom types to use during calculation of their contribution to atom pair count. Default value: *None*. Valid values: real numbers greater than 0. In general it's comma delimited list of valid atom type and its weight.

The weight values allow to increase the importance of specific pharmacophore atom type in the generated fingerprints. A weight value of 0 for an atom type eliminates its contribution to atom pair count where as weight value of 2 doubles its contribution.

--CompoundID *DataFieldName or LabelPrefixString*

This value is --CompoundIDMode specific and indicates how compound ID is generated.

For *DataField* value of --CompoundIDMode option, it corresponds to datafield label name whose value is used as compound ID; otherwise, it's a prefix string used for generating compound IDs like LabelPrefixString<Number>. Default value, *Cmpd*, generates compound IDs which look like Cmpd<Number>.

Examples for *DataField* value of --CompoundIDMode:

```
MolID
ExtReg
```

Examples for *LabelPrefix* or *MolNameOrLabelPrefix* value of --CompoundIDMode:

```
Compound
```

The value specified above generates compound IDs which correspond to Compound<Number> instead of default value of Cmpd<Number>.

--CompoundIDLabel *text*

Specify compound ID column label for CSV/TSV text file(s) used during *CompoundID* value of --DataFieldsMode option. Default value: *CompoundID*.

--CompoundIDMode *DataField* | *MolName* | *LabelPrefix* | *MolNameOrLabelPrefix*

Specify how to generate compound IDs and write to FP or CSV/TSV text file(s) along with generated fingerprints for *FP* | *text* | *all* values of --output option: use a *SDFile(s)* datafield value; use molname line from *SDFile(s)*; generate a sequential ID with specific prefix; use combination of both MolName and LabelPrefix with usage of LabelPrefix values for empty molname lines.

Possible values: *DataField* | *MolName* | *LabelPrefix* | *MolNameOrLabelPrefix*. Default value: *LabelPrefix*.

For *MolNameAndLabelPrefix* value of --CompoundIDMode, molname line in *SDFile(s)* takes precedence over sequential compound IDs generated using *LabelPrefix* and only empty molname values are replaced with sequential compound IDs.

This is only used for *CompoundID* value of --DataFieldsMode option.

--DataFields *"FieldLabel1,FieldLabel2,..."*

Comma delimited list of *SDFiles(s)* data fields to extract and write to CSV/TSV text file(s) along with generated fingerprints for *text* | *all* values of --output option.

This is only used for *Specify* value of --DataFieldsMode option.

Examples:

```
Extreg
MolID,CompoundName
```

-d, --DataFieldsMode *All* | *Common* | *Specify* | *CompoundID*

Specify how data fields in *SDFile(s)* are transferred to output CSV/TSV text file(s) along with generated fingerprints for *text* | *all* values of --output option: transfer all SD data field; transfer SD data files common to all compounds; extract specified data fields; generate a compound ID using molname line, a compound prefix, or a combination of both. Possible values: *All* | *Common* | *specify* | *CompoundID*. Default value: *CompoundID*.

-f, --Filter *Yes* | *No*

Specify whether to check and filter compound data in SDFile(s). Possible values: *Yes or No*. Default value: *Yes*.

By default, compound data is checked before calculating fingerprints and compounds containing atom data corresponding to non-element symbols or no atom data are ignored.

--FingerprintsLabelMode *FingerprintsLabelOnly* | *FingerprintsLabelWithIDs*

Specify how fingerprints label is generated in conjunction with --FingerprintsLabel option value: use fingerprints label generated only by --FingerprintsLabel option value or append topological atom pair count value IDs to --FingerprintsLabel option value.

Possible values: *FingerprintsLabelOnly* | *FingerprintsLabelWithIDs*. Default value: *FingerprintsLabelOnly*.

Topological atom pairs IDs appended to --FingerprintsLabel value during *FingerprintsLabelWithIDs* values of --FingerprintsLabelMode correspond to atom pair count values in fingerprint vector string.

FingerprintsLabelWithIDs value of --FingerprintsLabelMode is ignored during *ArbitrarySize* value of --AtomPairsSetSizeToUse option and topological atom pairs IDs not appended to the label.

--FingerprintsLabel *text*

SD data label or text file column label to use for fingerprints string in output SD or CSV/TSV text file(s) specified by --output. Default value: *TopologicalPharmacophoreAtomPairsFingerprints*.

--FuzzifyAtomPairsCount *Yes* | *No*

To fuzzify or not to fuzzify atom pairs count. Possible values: *Yes or No*. Default value: *No*.

--FuzzificationMode *BeforeNormalization* | *AfterNormalization*

When to fuzzify atom pairs count. Possible values: *BeforeNormalization* | *AfterNormalizationYes*. Default value: *AfterNormalization*.

--FuzzificationMethodology *FuzzyBinning* | *FuzzyBinSmoothing*

How to fuzzify atom pairs count. Possible values: *FuzzyBinning* | *FuzzyBinSmoothing*. Default value: *FuzzyBinning*.

In conjunction with values for options --FuzzifyAtomPairsCount, --FuzzificationMode and --FuzzFactor, --FuzzificationMethodology option is used to fuzzify pharmacophore atom pairs count.

Let:

```
Px = Pharmacophore atom type x
Py = Pharmacophore atom type y
PPxy = Pharmacophore atom pair between atom type Px and Py

PPxyDn = Pharmacophore atom pairs count between atom type Px and Py
         at distance Dn
PPxyDn-1 = Pharmacophore atom pairs count between atom type Px and Py
           at distance Dn - 1
PPxyDn+1 = Pharmacophore atom pairs count between atom type Px and Py
           at distance Dn + 1

FF = FuzzFactor for FuzzyBinning and FuzzyBinSmoothing
```

Then:

For *FuzzyBinning*:

```
PPxyDn = PPxyDn (Unchanged)

PPxyDn-1 = PPxyDn-1 + PPxyDn * FF
PPxyDn+1 = PPxyDn+1 + PPxyDn * FF
```

For *FuzzyBinSmoothing*:

```
PPxyDn = PPxyDn - PPxyDn * 2FF for Dmin < Dn < Dmax
PPxyDn = PPxyDn - PPxyDn * FF for Dn = Dmin or Dmax

PPxyDn-1 = PPxyDn-1 + PPxyDn * FF
PPxyDn+1 = PPxyDn+1 + PPxyDn * FF
```

In both fuzzification schemes, a value of 0 for FF implies no fuzzification of occurrence counts. A value of 1 during *FuzzyBinning* corresponds to maximum fuzzification of occurrence counts; however, a value of 1 during *FuzzyBinSmoothing* ends up completely distributing the value over the previous and next distance bins.

So for default value of --FuzzFactor (FF) 0.15, the occurrence count of pharmacohore atom pairs at distance Dn during FuzzyBinning is left unchanged and the counts at distances Dn -1 and Dn + 1 are incremented by PPxyDn * 0.15.

And during *FuzzyBinSmoothing* the occurrence counts at Distance Dn is scaled back using multiplicative factor of (1 - 2*0.15) and the occurrence counts at distances Dn -1 and Dn + 1 are incremented by PPxyDn * 0.15. In otherwords, occurrence bin count is smoothed out by distributing it over the previous and next distance value.

--FuzzFactor *number*

Specify by how much to fuzzify atom pairs count. Default value: *0.15*. Valid values: For *FuzzyBinning* value of --FuzzificationMethodology option: *between 0 and 1.0*; For *FuzzyBinSmoothing* value of --FuzzificationMethodology option: *between 0 and 0.5*.

-h, --help

Print this help message.

-k, --KeepLargestComponent *Yes* | *No*

Generate fingerprints for only the largest component in molecule. Possible values: *Yes or No*. Default value: *Yes*.

For molecules containing multiple connected components, fingerprints can be generated in two different ways: use all connected components or just the largest connected component. By default, all atoms except for the largest connected component are deleted before generation of fingerprints.

--MinDistance *number*

Minimum bond distance between atom pairs for generating topological pharmacophore atom pairs. Default value: *1*. Valid values: positive integers including 0 and less than --MaxDistance.

--MaxDistance *number*

Maximum bond distance between atom pairs for generating topological pharmacophore atom pairs. Default value: *10*. Valid values: positive integers and greater than --MinDistance.

-n, --NormalizationMethodology *None* | *ByHeavyAtomsCount* | *ByAtomTypesCount*

Normalization methodology to use for scaling the occurrence count of pharmacophore atom pairs within specified distance range. Possible values: *None, ByHeavyAtomsCount or ByAtomTypesCount*. Default value: *None*.

--OutDelim *comma* | *tab* | *semicolon*

Delimiter for output CSV/TSV text file(s). Possible values: *comma, tab, or semicolon* Default value: *comma*.

--output *SD* | *FP* | *text* | *all*

Type of output files to generate. Possible values: *SD, FP, text, or all*. Default value: *text*.

-o, --overwrite

Overwrite existing files.

-q, --quote *Yes* | *No*

Put quote around column values in output CSV/TSV text file(s). Possible values: *Yes or No*. Default value: *Yes*

-r, --root *RootName*

New file name is generated using the root: <Root>.<Ext>. Default for new file names: <SDFileName><TopologicalPharmacophoreAtomPairsFP>.<Ext>. The file type determines <Ext> value. The sdf, fpf, csv, and tsv <Ext> values are used for SD, FP, comma/semicolon, and tab delimited text files, respectively.This option is ignored for multiple input files.

--ValuesPrecision *number*

Precision of atom pairs count real values which might be generated after normalization or fuzzification. Default value: up to *2* decimal places. Valid values: positive integers.

-v, --VectorStringFormat *ValuesString, IDsAndValuesString* | *IDsAndValuesPairsString* | *ValuesAndIDsString* | *ValuesAndIDsPairsString*

Format of fingerprints vector string data in output SD, FP or CSV/TSV text file(s) specified by --output option. Possible values: *ValuesString, IDsAndValuesString* | *IDsAndValuesPairsString* | *ValuesAndIDsString* | *ValuesAndIDsPairsString*.

Default value during *FixedSize* value of --AtomPairsSetSizeToUse option: *ValuesString*. Default value during *ArbitrarySize* value of --AtomPairsSetSizeToUse option: *IDsAndValuesString*.

ValuesString option value is not allowed for *ArbitrarySize* value of --AtomPairsSetSizeToUse option.

Examples:

```
FingerprintsVector;TopologicalPharmacophoreAtomPairs:ArbitrarySize:Min
Distance1:MaxDistance10;54;NumericalValues;IDsAndValuesString;H-D1-H H
-D1-NI HBA-D1-NI HBD-D1-NI H-D2-H H-D2-HBA H-D2-HBD HBA-D2-HBA HBA-D2-
HBD H-D3-H H-D3-HBA H-D3-HBD H-D3-NI HBA-D3-NI HBD-D3-NI H-D4-H H-D4-H
BA H-D4-HBD HBA-D4-HBA HBA-D4-HBD HBD-D4-HBD H-D5-H H-D5-HBA H-D5-...;
18 1 2 1 22 12 8 1 2 18 6 3 1 1 1 22 13 6 5 7 2 28 9 5 1 1 1 36 16 10
3 4 1 37 10 8 1 35 10 9 3 3 1 28 7 7 4 18 16 12 5 1 2 1

FingerprintsVector;TopologicalPharmacophoreAtomPairs:FixedSize:MinDist
ance1:MaxDistance10;150;OrderedNumericalValues;ValuesString;18 0 0 1 0
0 0 2 0 0 1 0 0 0 0 22 12 8 0 0 1 2 0 0 0 0 0 0 0 0 18 6 3 1 0 0 0 1
0 0 1 0 0 0 0 22 13 6 0 0 5 7 0 0 2 0 0 0 0 0 28 9 5 1 0 0 0 1 0 0 1 0
0 0 0 36 16 10 0 0 3 4 0 0 1 0 0 0 0 0 37 10 8 0 0 0 0 1 0 0 0 0 0 0
0 35 10 9 0 0 3 3 0 0 1 0 0 0 0 0 28 7 7 4 0 0 0 0 0 0 0 0 0 0 0 18...

FingerprintsVector;TopologicalPharmacophoreAtomPairs:FixedSize:MinDist
ance1:MaxDistance10;150;OrderedNumericalValues;IDsAndValuesString;H-D1
-H H-D1-HBA H-D1-HBD H-D1-NI H-D1-PI HBA-D1-HBA HBA-D1-HBD HBA-D1-NI H
BA-D1-PI HBD-D1-HBD HBD-D1-NI HBD-D1-PI NI-D1-NI NI-D1-PI PI-D1-PI H-D
```

```
2-H H-D2-HBA H-D2-HBD H-D2-NI H-D2-PI HBA-D2-HBA HBA-D2-HBD HBA-D2...;
18 0 0 1 0 0 0 2 0 0 1 0 0 0 0 22 12 8 0 0 1 2 0 0 0 0 0 0 0 0 18 6 3
1 0 0 0 1 0 0 1 0 0 0 0 22 13 6 0 0 5 7 0 0 2 0 0 0 0 0 28 9 5 1 0 0 0
1 0 0 1 0 0 0 0 36 16 10 0 0 3 4 0 0 1 0 0 0 0
```

-w, --WorkingDir *DirName*

Location of working directory. Default value: current directory.

EXAMPLES

To generate topological pharmacophore atom pairs fingerprints of arbitrary size corresponding to distances from 1 through 10 using default atom types with no weighting, normalization, and fuzzification of atom pairs count and create a SampleTPAPFP.csv file containing sequential compound IDs along with fingerprints vector strings data in ValuesString format, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl -r SampleTPAPFP
  -o Sample.sdf
```

To generate topological pharmacophore atom pairs fingerprints of fixed size corresponding to distances from 1 through 10 using default atom types with no weighting, normalization, and fuzzification of atom pairs count and create a SampleTPAPFP.csv file containing sequential compound IDs along with fingerprints vector strings data in ValuesString format, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl
   --AtomPairsSetSizeToUse FixedSize -r SampleTPAPFP-o Sample.sdf
```

To generate topological pharmacophore atom pairs fingerprints of arbitrary size corresponding to distances from 1 through 10 using default atom types with no weighting, normalization, and fuzzification of atom pairs count and create SampleTPAPFP.sdf, SampleTPAPFP.fpf and SampleTPAPFP.csv files containing sequential compound IDs in CSV file along with fingerprints vector strings data in ValuesString format, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl --output all
  -r SampleTPAPFP -o Sample.sdf
```

To generate topological pharmacophore atom pairs fingerprints of arbitrary size corresponding to distances from 1 through 10 using default atom types with no weighting, normalization, and fuzzification of atom pairs count and create a SampleTPAPFP.csv file containing sequential compound IDs along with fingerprints vector strings data in IDsAndValuesPairsString format, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl --VectorStringFormat
  IDsAndValuesPairsString -r SampleTPAPFP -o Sample.sdf
```

To generate topological pharmacophore atom pairs fingerprints of arbitrary size corresponding to distances from 1 through 6 using default atom types with no weighting, normalization, and fuzzification of atom pairs count and create a SampleTPAPFP.csv file containing sequential compound IDs along with fingerprints vector strings data in ValuesString format, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl --MinDistance 1
  -MaxDistance 6 -r SampleTPAPFP -o Sample.sdf
```

To generate topological pharmacophore atom pairs fingerprints of arbitrary size corresponding to distances from 1 through 10 using "HBD,HBA,PI,NI" atom types with double the weighting for "HBD,HBA" and normalization by HeavyAtomCount but no fuzzification of atom pairs count and create a SampleTPAPFP.csv file containing sequential compound IDs along with fingerprints vector strings data in ValuesString format, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl --MinDistance 1
  -MaxDistance 10  --AtomTypesToUse "HBD,HBA,PI, NI"  --AtomTypesWeight
  "HBD,2,HBA,2,PI,1,NI,1" --NormalizationMethodology ByHeavyAtomsCount
  --FuzzifyAtomPairsCount No -r SampleTPAPFP -o Sample.sdf
```

To generate topological pharmacophore atom pairs fingerprints of arbitrary size corresponding to distances from 1 through 10 using "HBD,HBA,PI,NI,H" atom types with no weighting of atom types and normalization but with fuzzification of atom pairs count using FuzzyBinning methodology with FuzzFactor value 0.15 and create a SampleTPAPFP.csv file containing sequential compound IDs along with fingerprints vector strings data in ValuesString format, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl --MinDistance 1
  --MaxDistance 10  --AtomTypesToUse "HBD,HBA,PI, NI,H"  --AtomTypesWeight
  "HBD,1,HBA,1,PI,1,NI,1,H,1" --NormalizationMethodology None
  --FuzzifyAtomPairsCount Yes --FuzzificationMethodology FuzzyBinning
  --FuzzFactor  0.5 -r SampleTPAPFP -o Sample.sdf
```

To generate topological pharmacophore atom pairs fingerprints of arbitrary size corresponding to distances distances from 1 through 10 using default atom types with no weighting, normalization, and fuzzification of atom pairs count and create a SampleTPAPFP.csv file containing compound ID from molecule name line along with fingerprints vector strings data, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl --DataFieldsMode
  CompoundID -CompoundIDMode MolName -r SampleTPAPFP -o Sample.sdf
```

To generate topological pharmacophore atom pairs fingerprints of arbitrary size corresponding to distances from 1 through 10 using default atom types with no weighting, normalization, and fuzzification of atom pairs count and create a SampleTPAPFP.csv file containing compound IDs using specified data field along with fingerprints vector strings data, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl --DataFieldsMode
  CompoundID -CompoundIDMode DataField --CompoundID Mol_ID
  -r SampleTPAPFP -o Sample.sdf
```

To generate topological pharmacophore atom pairs fingerprints of arbitrary size corresponding to distances from 1 through 10 using default atom types with no weighting, normalization, and fuzzification of atom pairs count and create a SampleTPAPFP.csv file containing compound ID using combination of molecule name line and an explicit compound prefix along with fingerprints vector strings data, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl --DataFieldsMode
  CompoundID -CompoundIDMode MolnameOrLabelPrefix
  --CompoundID Cmpd --CompoundIDLabel MolID -r SampleTPAPFP -o Sample.sdf
```

To generate topological pharmacophore atom pairs fingerprints of arbitrary size corresponding to distances from 1 through 10 using default atom types with no weighting, normalization, and fuzzification of atom pairs count and create a SampleTPAPFP.csv file containing specific data fields columns along with fingerprints vector strings data, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl --DataFieldsMode
  Specify --DataFields Mol_ID -r SampleTPAPFP -o Sample.sdf
```

To generate topological pharmacophore atom pairs fingerprints of arbitrary size corresponding to distances from 1 through 10 using default atom types with no weighting, normalization, and fuzzification of atom pairs count and create a SampleTPAPFP.csv file containing common data fields columns along with fingerprints vector strings data, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl --DataFieldsMode
  Common -r SampleTPAPFP -o Sample.sdf
```

To generate topological pharmacophore atom pairs fingerprints of arbitrary size corresponding to distances from 1 through 10 using default atom types with no weighting, normalization, and fuzzification of atom pairs count and create SampleTPAPFP.sdf, SampleTPAPFP.fpf, and SampleTPAPFP.csv files containing all data fields columns in CSV file along with fingerprints data, type:

```
% TopologicalPharmacophoreAtomPairsFingerprints.pl --DataFieldsMode
  All  --output all -r SampleTPAPFP -o Sample.sdf
```

AUTHOR

Manish Sud <msud@san.rr.com>

SEE ALSO

InfoFingerprintsFiles.pl, SimilarityMatricesFingerprints.pl, AtomNeighborhoodsFingerprints.pl, ExtendedConnectivityFingerprints.pl, MACCSKeysFingerprints.pl, PathLengthFingerprints.pl, TopologicalAtomPairsFingerprints.pl, TopologicalAtomTorsionsFingerprints.pl, TopologicalPharmacophoreAtomTripletsFingerprints.pl

COPYRIGHT