

## NAME

SimilaritySearchingFingerprints.pl - Perform similarity search using fingerprints strings data in SD, FP and CSV/TSV text file(s)

## SYNOPSIS

SimilaritySearchingFingerprints.pl ReferenceFPFile DatabaseFPFile

SimilaritySearchingFingerprints.pl [--alpha *number*] [--beta *number*] [-b, --BitVectorComparisonMode *TanimotoSimilarity* | *TverskySimilarity* | ...] [--DatabaseColMode *ColNum* | *ColLabel*] [--DatabaseCompoundIDCol *col number* | *col name*] [--DatabaseCompoundIDPrefix *text*] [--DatabaseCompoundIDField *DataFieldName*] [--DatabaseCompoundIDMode *DataField* | *MolName* | *LabelPrefix* | *MolNameOrLabelPrefix*] [--DatabaseDataCols "*DataColNum1*, *DataColNum2*,... " | *DataColLabel1*, *DataColLabel2*,... "] [--DatabaseDataColsMode *All* | *Specify* | *CompoundID*] [--DatabaseDataFields "*FieldLabel1*, *FieldLabel2*,... "] [--DatabaseDataFieldsMode *All* | *Common* | *Specify* | *CompoundID*] [--DatabaseFingerprintsCol *col number* | *col name*] [--DatabaseFingerprintsField *FieldLabel*] [--DistanceCutoff *number*] [-d, --detail *InfoLevel*] [-f, --fast] [--FingerprintsMode *AutoDetect* | *FingerprintsBitVectorString* | *FingerprintsVectorString*] [-g, --GroupFusionRule *Max*, *Mean*, *Median*, *Min*, *Sum*, *Euclidean*] [--GroupFusionApplyCutoff *Yes* | *No*] [-h, --help] [--InDelim *comma* | *semicolon*] [-k, --KNN *all* | *number*] [-m, --mode *IndividualReference* | *MultipleReferences*] [-n, --NumOfSimilarMolecules *number*] [--OutDelim *comma* | *tab* | *semicolon*] [--output *SD* | *text* | *both*] [-o, --overwrite] [-p, --PercentSimilarMolecules *number*] [--precision *number*] [-q, --quote *Yes* | *No*] [--ReferenceColMode *ColNum* | *ColLabel*] [--ReferenceCompoundIDCol *col number* | *col name*] [--ReferenceCompoundIDPrefix *text*] [--ReferenceCompoundIDField *DataFieldName*] [--ReferenceCompoundIDMode *DataField* | *MolName* | *LabelPrefix* | *MolNameOrLabelPrefix*] [--ReferenceFingerprintsCol *col number* | *col name*] [--ReferenceFingerprintsField *FieldLabel*] [-r, --root *RootName*] [-s, --SearchMode *SimilaritySearch* | *DissimilaritySearch*] [--SimilarCountMode *NumOfSimilar* | *PercentSimilar*] [--SimilarityCutoff *number*] [-v, --VectorComparisonMode *TanimotoSimilarity* | ... | *ManhattanDistance* | ...] [--VectorComparisonFormalism *AlgebraicForm* | *BinaryForm* | *SetTheoreticForm*] [-w, --WorkingDir *dirname*] ReferenceFingerprintsFile DatabaseFingerprintsFile

## DESCRIPTION

Perform molecular similarity search [ Ref 94-113 ] using fingerprint bit-vector or vector strings data in *SD*, *FP*, or *CSV/TSV text* files corresponding to *ReferenceFingerprintsFile* and *DatabaseFingerprintsFile*, and generate *SD* and *CSV/TSV text* file(s) containing database molecules which are similar to reference molecule(s). The reference molecules are also referred to as query or seed molecules and database molecules as target molecules in the literature.

The current release of MayaChemTools supports two types of similarity search modes: *IndividualReference* or *MultipleReferences*. For default value of *MultipleReferences* for -m, --mode option, reference molecules are considered as a set and -g, --GroupFusionRule is used to calculate similarity of a database molecule against reference molecules set. The group fusion rule is also referred to as data fusion or consensus scoring in the literature. However, for *IndividualReference* value of -m, --mode option, reference molecules are treated as individual molecules and each reference molecule is compared against a database molecule by itself to identify similar molecules.

The molecular dissimilarity search can also be performed using *DissimilaritySearch* value for -s, --SearchMode option. During dissimilarity search or usage of distance comparison coefficient in similarity similarity search, the meaning of fingerprints comparison value is automatically reversed as shown below:

SeachMode	ComparisonCoefficient	ResultsSort	ComparisonValues
Similarity	SimilarityCoefficient	Descending	Higher value implies high similarity
Similarity	DistanceCoefficient	Ascending	Lower value implies high similarity
Dissimilarity	SimilarityCoefficient	Ascending	Lower value implies high dissimilarity
Dissimilarity	DistanceCoefficient	Descending	Higher value implies high dissimilarity

During *IndividualReference* value of -m, --Mode option for similarity search, fingerprints bit-vector or vector string of each reference molecule is compared with database molecules using specified similarity or distance coefficients to identify most similar molecules for each reference molecule. Based on value of --SimilarCountMode, up to --n,

--NumOfSimilarMolecules or -p, --PercentSimilarMolecules at specified --SimilarityCutoff or --DistanceCutoff are identified for each reference molecule.

During *MultipleReferences* value -m, --mode option for similarity search, all reference molecules are considered as a set and -g, --GroupFusionRule is used to calculate similarity of a database molecule against reference molecules set either using all reference molecules or number of k-nearest neighbors (k-NN) to a database molecule specified using -k, --kNN. The fingerprints bit-vector or vector string of each reference molecule in a set is compared with a database molecule using a similarity or distance coefficient specified via -b, --BitVectorComparisonMode or -v, --VectorComparisonMode. The reference molecules whose comparison values with a database molecule fall outside specified --SimilarityCutoff or --DistanceCutoff are ignored during *Yes* value of --GroupFusionApplyCutoff. The specified -g, --GroupFusionRule is applied to -k, --kNN reference molecules to calculate final similarity value between a database molecule and reference molecules set.

The input fingerprints *SD*, *FP*, or *Text (CSV/TSV)* files for *ReferenceFingerprintsFile* and *DatabaseTextFile* must contain valid fingerprint bit-vector or vector strings data corresponding to same type of fingerprints.

The valid fingerprints *SDF* file extensions are *.sdf* and *.sd*. The valid fingerprints *FP* file extensions are *.fpf* and *.fp*. The valid fingerprints *TextFile (CSV/TSV)* extensions are *.csv* and *.tsv* for comma/semicolon and tab delimited text files respectively. The --indelim option determines the format of *TextFile*. Any file which doesn't correspond to the format indicated by --indelim option is ignored.

Example of *FP* file containing fingerprints bit-vector string data:

```
#
# Package = MayaChemTools 7.4
# ReleaseDate = Oct 21, 2010
#
# TimeStamp = Mon Mar 7 15:14:01 2011
#
# FingerprintsStringType = FingerprintsBitVector
#
# Description = PathLengthBits:AtomicInvariantsAtomTypes:MinLength1:...
# Size = 1024
# BitStringFormat = HexadecimalString
# BitsOrder = Ascending
#
Cmpd1 9c8460989ec8a49913991a6603130b0a19e8051c89184414953800cc21510...
Cmpd2 000000249400840040100042011001001980410c000000001010088001120...
... ..
... ..
```

Example of *FP* file containing fingerprints vector string data:

```
#
# Package = MayaChemTools 7.4
# ReleaseDate = Oct 21, 2010
#
# TimeStamp = Mon Mar 7 15:14:01 2011
#
# FingerprintsStringType = FingerprintsVector
#
# Description = PathLengthBits:AtomicInvariantsAtomTypes:MinLength1:...
# VectorStringFormat = IDsAndValuesString
# VectorValueType = NumericalValues
#
Cmpd1 338;C F N O C:C C:N C=O CC CF CN CO C:C:C C:C:N C:CC C:CF C:CN C:
N:C C:NC CC:N CC=O CCC CCN CCO CNC NC=O O=CO C:C:C:C C:C:C:N C:C:CC...;
33 1 2 5 21 2 2 12 1 3 3 20 2 10 2 2 1 2 2 2 8 2 5 1 1 1 19 2 8 2 2 2
6 2 2 2 2 2 2 2 3 2 2 1 4 1 5 1 1 18 6 2 2 1 2 10 2 1 2 1 2 2 2 ...
Cmpd2 103;C N O C=N C=O CC CN CO CC=O CCC CCN CCO CNC N=CN NC=O NCN O=C
O C CC=O CCCC CCCN CCCO CCNC CNC=N CNC=O CNCN CCCC=O CCCCC CCCCN CC...;
15 4 4 1 2 13 5 2 2 15 5 3 2 2 1 1 2 17 7 6 5 1 1 1 2 15 8 5 7 2 2 2
1 2 1 1 3 15 7 6 8 3 4 4 3 2 2 1 2 3 14 2 4 7 4 4 4 1 1 1 2 1 1 1 ...
... ..
```

... ..

Example of *SD* file containing fingerprints bit-vector string data:

```
... ..
... ..
$$$$
... ..
... ..
... ..
41 44 0 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

> <PathLengthFingerprints>
FingerprintsBitVector;PathLengthBits:AtomicInvariantsAtomTypes:MinLength:MaxLength8;1024;HexadecimalString;Ascending;9c8460989ec8a49913991a6603130b0a19e8051c89184414953800cc2151082844a201042800130860308e8204d402800831048940e44281c00060449a5000ac80c894114e006321264401600846c05016446208190410805000304a10205b0100e04c0038ba0fad0209c0ca8b1200012268b61c0026a
aa0660a11014a011d46

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

Example of CSV *TextFile* containing fingerprints bit-vector string data:

```
"CompoundID", "PathLengthFingerprints"
"Cmpd1", "FingerprintsBitVector;PathLengthBits:AtomicInvariantsAtomTypes:MinLength:MaxLength8;1024;HexadecimalString;Ascending;9c8460989ec8a49913991a6603130b0a19e8051c89184414953800cc2151082844a201042800130860308e8204d402800831048940e44281c00060449a5000ac80c894114e006321264401..."
... ..
... ..
```

The current release of MayaChemTools supports the following types of fingerprint bit-vector and vector strings:

```
FingerprintsVector;AtomNeighborhoods:AtomicInvariantsAtomTypes:MinRadius0:MaxRadius2;41;AlphaNumericalValues;ValuesString;NR0-C.X1.BO1.H3-ATC1:NR1-C.X3.BO3.H1-ATC1:NR2-C.X1.BO1.H3-ATC1:NR2-C.X3.BO4-ATC1 NR0-C.X1.BO1.H3-ATC1:NR1-C.X3.BO3.H1-ATC1:NR2-C.X1.BO1.H3-ATC1:NR2-C.X3.BO4-ATC1 NR0-C.X2.BO2.H2-ATC1:NR1-C.X2.BO2.H2-ATC1:NR1-C.X3.BO3.H1-ATC1:NR2-C.X2.BO2.H2-ATC1:NR2-N.X3.BO3-ATC1:NR2-O.X1.BO1.H1-ATC1 NR0-C.X2.B...
```

```
FingerprintsVector;AtomTypesCount:AtomicInvariantsAtomTypes:ArbitrarySize;10;NumericalValues;IDsAndValuesString;C.X1.BO1.H3 C.X2.BO2.H2 C.X2.BO3.H1 C.X3.BO3.H1 C.X3.BO4 F.X1.BO1 N.X2.BO2.H1 N.X3.BO3 O.X1.BO1.H1 O.X1.BO2;2 4 14 3 10 1 1 3 2
```

```
FingerprintsVector;AtomTypesCount:SLogPAtomTypes:ArbitrarySize;16;NumericalValues;IDsAndValuesString;C1 C10 C11 C14 C18 C20 C21 C22 C5 CS F N11 N4 O10 O2 O9;5 1 1 1 14 4 2 1 2 2 1 1 1 1 3 1
```

```
FingerprintsVector;AtomTypesCount:SLogPAtomTypes:FixedSize;67;OrderedNumericalValues;IDsAndValuesString;C1 C2 C3 C4 C5 C6 C7 C8 C9 C10 C11 C
```





```
H-aaCH-aasC-aaCH aaCH-aaCH-aasC-aasC aaCH-aaCH-aasC-sF aaCH-aaCH-aasC-
ssNH aaCH-aasC-aasC-aasC aaCH-aasC-aasC-aasN aaCH-aasC-ssNH-dssC a...;
4 4 8 4 2 2 6 2 2 2 4 3 2 1 3 3 2 2 2 1 2 1 1 1 2 1 1 1 1 1 1 1 2 1 1 2
```

```
FingerprintsVector;TopologicalAtomTriplets:AtomicInvariantsAtomTypes:MinDistance1:MaxDistance10;3096;NumericalValues;IDsAndValuesString;C.X1
.BO1.H3-D1-C.X1.BO1.H3-D1-C.X3.BO3.H1-D2 C.X1.BO1.H3-D1-C.X2.BO2.H2-D1
0-C.X3.BO4-D9 C.X1.BO1.H3-D1-C.X2.BO2.H2-D3-N.X3.BO3-D4 C.X1.BO1.H3-D1
-C.X2.BO2.H2-D4-C.X2.BO2.H2-D5 C.X1.BO1.H3-D1-C.X2.BO2.H2-D6-C.X3...;
1 2 2 2 2 2 2 8 8 4 8 4 4 2 2 2 2 4 2 2 2 4 2 2 2 1 2 2 4 4 4 2 2
2 4 4 4 8 4 4 2 4 4 4 2 4 4 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 8...
```

```
FingerprintsVector;TopologicalAtomTriplets:SYBYLAtomTypes:MinDistance1:MaxDistance10;2332;NumericalValues;IDsAndValuesString;C.2-D1-C.2-D9-C
.3-D10 C.2-D1-C.2-D9-C.ar-D10 C.2-D1-C.3-D1-C.3-D2 C.2-D1-C.3-D10-C.3-
D9 C.2-D1-C.3-D2-C.3-D3 C.2-D1-C.3-D2-C.ar-D3 C.2-D1-C.3-D3-C.3-D4 C.2
-D1-C.3-D3-N.ar-D4 C.2-D1-C.3-D3-O.3-D2 C.2-D1-C.3-D4-C.3-D5 C.2-D1-C.
3-D5-C.3-D6 C.2-D1-C.3-D5-O.3-D4 C.2-D1-C.3-D6-C.3-D7 C.2-D1-C.3-D7...
```

```
FingerprintsVector;TopologicalPharmacophoreAtomPairs:ArbitrarySize: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 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:MinDistance1: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;TopologicalPharmacophoreAtomTriplets:ArbitrarySize:MinDistance1:MaxDistance10;696;NumericalValues;IDsAndValuesString;Ar1-
Ar1-Ar1 Ar1-Ar1-H1 Ar1-Ar1-HBA1 Ar1-Ar1-HBD1 Ar1-H1-H1 Ar1-H1-HBA1 Ar1
-H1-HBD1 Ar1-HBA1-HBD1 H1-H1-H1 H1-H1-HBA1 H1-H1-HBD1 H1-HBA1-HBA1 H1-
HBA1-HBD1 H1-HBA1-NI1 H1-HBD1-NI1 HBA1-HBA1-NI1 HBA1-HBD1-NI1 Ar1-...;
46 106 8 3 83 11 4 1 21 5 3 1 2 2 1 1 1 100 101 18 11 145 132 26 14 23
28 3 3 5 4 61 45 10 4 16 20 7 5 1 3 4 5 3 1 1 1 5 4 2 1 2 2 2 1 1 1
119 123 24 15 185 202 41 25 22 17 3 5 85 95 18 11 23 17 3 1 1 6 4 ...
```

```
FingerprintsVector;TopologicalPharmacophoreAtomTriplets:FixedSize:MinDistance1:MaxDistance10;2692;OrderedNumericalValues;ValuesString;46 106
8 3 0 0 83 11 4 0 0 0 1 0 0 0 0 0 0 0 0 21 5 3 0 0 1 2 2 0 0 1 0 0 0
0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 100 101 18 11 0 0 145 132 26
14 0 0 23 28 3 3 0 0 5 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 61 45 10 4 0
0 16 20 7 5 1 0 3 4 5 3 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 5 ...
```

## OPTIONS

### --alpha *number*

Value of alpha parameter for calculating *Tversky* similarity coefficient specified for -b, --BitVectorComparisonMode option. It corresponds to weights assigned for bits set to "1" in a pair of fingerprint bit-vectors during the calculation of similarity coefficient. Possible values: 0 to 1. Default value: <0.5>.

### --beta *number*

Value of beta parameter for calculating *WeightedTanimoto* and *WeightedTversky* similarity coefficients specified for -b, --BitVectorComparisonMode option. It is used to weight the contributions of bits set to "0" during

the calculation of similarity coefficients. Possible values: 0 to 1. Default value of <1> makes *WeightedTanimoto* and *WeightedTversky* equivalent to *Tanimoto* and *Tversky*.

-b, --BitVectorComparisonMode *TanimotoSimilarity* | *TverskySimilarity* | ...

Specify what similarity coefficient to use for calculating similarity between fingerprints bit-vector string data values in *ReferenceFingerprintsFile* and *DatabaseFingerprintsFile* during similarity search. Possible values: *TanimotoSimilarity* | *TverskySimilarity* | .... Default: *TanimotoSimilarity*

The current release supports the following similarity coefficients: *BaroniUrbaniSimilarity*, *BuserSimilarity*, *CosineSimilarity*, *DiceSimilarity*, *DennisSimilarity*, *ForbesSimilarity*, *FossumSimilarity*, *HamannSimilarity*, *JaccardSimilarity*, *Kulczynski1Similarity*, *Kulczynski2Similarity*, *MatchingSimilarity*, *McConnaugheySimilarity*, *OchiaiSimilarity*, *PearsonSimilarity*, *RogersTanimotoSimilarity*, *RussellRaoSimilarity*, *SimpsonSimilarity*, *SkoalSneath1Similarity*, *SkoalSneath2Similarity*, *SkoalSneath3Similarity*, *TanimotoSimilarity*, *TverskySimilarity*, *YuleSimilarity*, *WeightedTanimotoSimilarity*, *WeightedTverskySimilarity*. These similarity coefficients are described below.

For two fingerprint bit-vectors A and B of same size, let:

Na = Number of bits set to "1" in A  
 Nb = Number of bits set to "1" in B  
 Nc = Number of bits set to "1" in both A and B  
 Nd = Number of bits set to "0" in both A and B  
  
 Nt = Number of bits set to "1" or "0" in A or B (Size of A or B)  
 Nt = Na + Nb - Nc + Nd  
  
 Na - Nc = Number of bits set to "1" in A but not in B  
 Nb - Nc = Number of bits set to "1" in B but not in A

Then, various similarity coefficients [ Ref. 40 - 42 ] for a pair of bit-vectors A and B are defined as follows:

*BaroniUrbaniSimilarity*:  $(\text{SQRT}(Nc * Nd) + Nc) / (\text{SQRT}(Nc * Nd) + Nc + (Na - Nc) + (Nb - Nc))$  ( same as Buser )

*BuserSimilarity*:  $(\text{SQRT}(Nc * Nd) + Nc) / (\text{SQRT}(Nc * Nd) + Nc + (Na - Nc) + (Nb - Nc))$  ( same as BaroniUrbani )

*CosineSimilarity*:  $Nc / \text{SQRT}(Na * Nb)$  (same as Ochiai)

*DiceSimilarity*:  $(2 * Nc) / (Na + Nb)$

*DennisSimilarity*:  $(Nc * Nd - ((Na - Nc) * (Nb - Nc))) / \text{SQRT}(Nt * Na * Nb)$

*ForbesSimilarity*:  $(Nt * Nc) / (Na * Nb)$

*FossumSimilarity*:  $(Nt * ((Nc - 1/2) ** 2)) / (Na * Nb)$

*HamannSimilarity*:  $((Nc + Nd) - (Na - Nc) - (Nb - Nc)) / Nt$

*JaccardSimilarity*:  $Nc / ((Na - Nc) + (Nb - Nc) + Nc) = Nc / (Na + Nb - Nc)$  (same as Tanimoto)

*Kulczynski1Similarity*:  $Nc / ((Na - Nc) + (Nb - Nc)) = Nc / (Na + Nb - 2Nc)$

*Kulczynski2Similarity*:  $((Nc / 2) * (2 * Nc + (Na - Nc) + (Nb - Nc))) / ((Nc + (Na - Nc)) * (Nc + (Nb - Nc))) = 0.5 * (Nc / Na + Nc / Nb)$

*MatchingSimilarity*:  $(Nc + Nd) / Nt$

*McConnaugheySimilarity*:  $(Nc ** 2 - (Na - Nc) * (Nb - Nc)) / (Na * Nb)$

*OchiaiSimilarity*:  $Nc / \text{SQRT}(Na * Nb)$  (same as Cosine)

*PearsonSimilarity*:  $((Nc * Nd) - ((Na - Nc) * (Nb - Nc))) / \text{SQRT}(Na * Nb * (Na - Nc + Nd) * (Nb - Nc + Nd))$

*RogersTanimotoSimilarity*:  $(Nc + Nd) / ((Na - Nc) + (Nb - Nc) + Nt) = (Nc + Nd) / (Na + Nb - 2Nc + Nt)$

*RussellRaoSimilarity*:  $Nc / Nt$

*SimpsonSimilarity*:  $Nc / \text{MIN}(Na, Nb)$

*SkoalSneath1Similarity*:  $Nc / (Nc + 2 * (Na - Nc) + 2 * (Nb - Nc)) = Nc / (2 * Na + 2 * Nb - 3 * Nc)$

*SkoalSneath2Similarity*:  $(2 * Nc + 2 * Nd) / (Nc + Nd + Nt)$

*SkoalSneath3Similarity*:  $(Nc + Nd) / ((Na - Nc) + (Nb - Nc)) = (Nc + Nd) / (Na + Nb - 2 * Nc)$

*TanimotoSimilarity*:  $Nc / ((Na - Nc) + (Nb - Nc) + Nc) = Nc / (Na + Nb - Nc)$  (same as Jaccard)

*TverskySimilarity*:  $Nc / (\alpha * (Na - Nc) + (1 - \alpha) * (Nb - Nc) + Nc) = Nc / (\alpha * (Na - Nb) +$

$$\text{Nb}) \text{YuleSimilarity: } ((Nc * Nd) - ((Na - Nc) * (Nb - Nc))) / ((Nc * Nd) + ((Na - Nc) * (Nb - Nc)))$$

Values of Tanimoto/Jaccard and Tversky coefficients are dependent on only those bit which are set to "1" in both A and B. In order to take into account all bit positions, modified versions of Tanimoto [ Ref. 42 ] and Tversky [ Ref. 43 ] have been developed.

Let:

Na' = Number of bits set to "0" in A  
Nb' = Number of bits set to "0" in B  
Nc' = Number of bits set to "0" in both A and B

Tanimoto':  $Nc' / ((Na' - Nc') + (Nb' - Nc') + Nc') = Nc' / (Na' + Nb' - Nc')$

Tversky':  $Nc' / (\alpha * (Na' - Nc') + (1 - \alpha) * (Nb' - Nc') + Nc') = Nc' / (\alpha * (Na' - Nb') + Nb')$

Then:

*WeightedTanimotoSimilarity* =  $\beta * \text{Tanimoto} + (1 - \beta) * \text{Tanimoto}'$

*WeightedTverskySimilarity* =  $\beta * \text{Tversky} + (1 - \beta) * \text{Tversky}'$

--DatabaseColMode *ColNum* | *ColLabel*

Specify how columns are identified in database fingerprints *TextFile*: using column number or column label. Possible values: *ColNum* or *ColLabel*. Default value: *ColNum*.

--DatabaseCompoundIDCol *col number* | *col name*

This value is --DatabaseColMode mode specific. It specifies column to use for retrieving compound ID from database fingerprints *TextFile* during similarity and dissimilarity search for output SD and CSV/TSV text files. Possible values: *col number* or *col label*. Default value: *first column containing the word compoundID in its column label or sequentially generated IDs*.

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

--DatabaseCompoundIDPrefix *text*

Specify compound ID prefix to use during sequential generation of compound IDs for database fingerprints *SDFFile* and *TextFile*. Default value: *Cmpd*. The default value generates compound IDs which look like *Cmpd<Number>*.

For database fingerprints *SDFFile*, this value is only used during *LabelPrefix* | *MolNameOrLabelPrefix* values of --DatabaseCompoundIDMode option; otherwise, it's ignored.

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

Compound

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

--DatabaseCompoundIDField *DataFieldName*

Specify database fingerprints *SDFFile* datafield label for generating compound IDs. This value is only used during *DataField* value of --DatabaseCompoundIDMode option.

Examples for *DataField* value of --DatabaseCompoundIDMode:

MolID  
ExtReg

--DatabaseCompoundIDMode *DataField* | *MolName* | *LabelPrefix* | *MolNameOrLabelPrefix*

Specify how to generate compound IDs from database fingerprints *SDFFile* during similarity and dissimilarity search for output SD and CSV/TSV text files: use a *SDFFile* datafield value; use molname line from *SDFFile*; 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: *LabelPrefix*.

For *MolNameAndLabelPrefix* value of --DatabaseCompoundIDMode, molname line in *SDFFile* 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 --DatabaseDataFieldsMode option.



--DatabaseDataCols "*DataColNum1,DataColNum2,...* " | *DataColLabel1,DataColLabel2,...* "

This value is --DatabaseColMode mode specific. It is a comma delimited list of database fingerprints *TextFile* data column numbers or labels to extract and write to SD and CSV/TSV text files along with other information for *SD | text | both* values of --output option.

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

Examples:

```
1,2,3
CompoundName,MolWt
```

--DatabaseDataColsMode *All | Specify | CompoundID*

Specify how data columns from database fingerprints *TextFile* are transferred to output SD and CSV/TSV text files along with other information for *SD | text | both* values of --output option: transfer all data columns; extract specified data columns; generate a compound ID database compound prefix. Possible values: *All | Specify | CompoundID*. Default value: *CompoundID*.

--DatabaseDataFields "*FieldLabel1,FieldLabel2,...* "

Comma delimited list of database fingerprints *SDFFile* data fields to extract and write to SD and CSV/TSV text files along with other information for *SD | text | both* values of --output option.

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

Examples:

```
Extreg
MolID,CompoundName
```

--DatabaseDataFieldsMode *All | Common | Specify | CompoundID*

Specify how data fields from database fingerprints *SDFFile* are transferred to output SD and CSV/TSV text files along with other information for *SD | text | both* 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*.

--DatabaseFingerprintsCol *col number | col name*

This value is --DatabaseColMode specific. It specifies fingerprints column to use during similarity and dissimilarity search for database fingerprints *TextFile*. Possible values: *col number or col label*. Default value: *first column containing the word Fingerprints in its column label*.

--DatabaseFingerprintsField *FieldLabel*

Fingerprints field label to use during similarity and dissimilarity search for database fingerprints *SDFFile*. Default value: *first data field label containing the word Fingerprints in its label*

--DistanceCutoff *number*

Distance cutoff value to use during comparison of distance value between a pair of database and reference molecule calculated by distance comparison methods for fingerprints vector string data values. Possible values: *Any valid number*. Default value: *10*.

The comparison value between a pair of database and reference molecule must meet the cutoff criterion as shown below:

SeachMode	CutoffCriterion	ComparisonValues
Similarity	<=	Lower value implies high similarity
Dissimilarity	>=	Higher value implies high dissimilarity

This option is only used during distance coefficients values of -v, --VectorComparisonMode option.

This option is ignored during *No* value of --GroupFusionApplyCutoff for *MultipleReferences* -m, --mode.

-d, --detail *InfoLevel*

Level of information to print about lines being ignored. Default: *1*. Possible values: *1, 2 or 3*.

-f, --fast

In this mode, fingerprints columns specified using `--FingerprintsCol` for reference and database fingerprints *TextFile(s)*, and `--FingerprintsField` for reference and database fingerprints *SDFFile(s)* are assumed to contain valid fingerprints data and no checking is performed before performing similarity and dissimilarity search. By default, fingerprints data is validated before computing pairwise similarity and distance coefficients.

`--FingerprintsMode` *AutoDetect | FingerprintsBitVectorString | FingerprintsVectorString*

Format of fingerprint strings data in reference and database fingerprints *SD, FP, or Text (CSV/TSV)* files: automatically detect format of fingerprints string created by MayaChemTools fingerprints generation scripts or explicitly specify its format. Possible values: *AutoDetect | FingerprintsBitVectorString | FingerprintsVectorString*. Default value: *AutoDetect*.

`-g, --GroupFusionRule` *Max, Min, Mean, Median, Sum, Euclidean*

Specify what group fusion [ Ref 94-97, Ref 100, Ref 105 ] rule to use for calculating similarity of a database molecule against a set of reference molecules during *MultipleReferences* value of similarity search `-m, --mode`. Possible values: *Max, Min, Mean, Median, Sum, Euclidean*. Default value: *Max*. *Mean* value corresponds to average or arithmetic mean. The group fusion rule is also referred to as data fusion of consensus scoring in the literature.

For a reference molecules set and a database molecule, let:

$N$  = Number of reference molecules in a set

$i$  =  $i$ th reference reference molecule in a set

$n$  =  $N$ th reference reference molecule in a set

$d$  =  $d$ th database molecule

$Crd$  = Fingerprints comparison value between  $r$ th reference and  $d$ th database molecule - similarity/dissimilarity comparison using similarity or distance coefficient

Then, various group fusion rules to calculate fused similarity between a database molecule and reference molecules set are defined as follows:

Max:  $MAX ( C1d, C2d, \dots, Cid, \dots, Cnd )$

Min:  $MIN ( C1d, C2d, \dots, Cid, \dots, Cnd )$

Mean:  $SUM ( C1d, C2d, \dots, Cid, \dots, Cnd ) / N$

Median:  $MEDIAN ( C1d, C2d, \dots, Cid, \dots, Cnd )$

Sum:  $SUM ( C1d, C2d, \dots, Cid, \dots, Cnd )$

Euclidean:  $SQRT( SUM( C1d ** 2, C2d ** 2, \dots, Cid ** 2, \dots, Cnd ** 2 ) )$

The fingerprints bit-vector or vector string of each reference molecule in a set is compared with a database molecule using a similarity or distance coefficient specified via `-b, --BitVectorComparisonMode` or `-v, --VectorComparisonMode`. The reference molecules whose comparison values with a database molecule fall outside specified `--SimilarityCutoff` or `--DistanceCutoff` are ignored during *Yes* value of `--GroupFusionApplyCutoff`. The specified `-g, --GroupFusionRule` is applied to `-k, --kNN` reference molecules to calculate final fused similarity value between a database molecule and reference molecules set.

During dissimilarity search or usage of distance comparison coefficient in similarity search, the meaning of fingerprints comparison value is automatically reversed as shown below:

SeachMode	ComparisonCoefficient	ComparisonValues
Similarity	SimilarityCoefficient	Higher value impls high similarity
Similarity	DistanceCoefficient	Lower value implies high similarity
Dissimilarity	SimilarityCoefficient	Lower value implies high dissimilarity
Dissimilarity	DistanceCoefficient	Higher value implies high dissimilarity

Consequently, *Max* implies highest and lowest comparison value for usage of similarity and distance coefficient respectively during similarity search. And it corresponds to lowest and highest comparison value

for usage of similarity and distance coefficient respectively during dissimilarity search. During *Min* fusion rule, the highest and lowest comparison values are appropriately reversed.

**--GroupFusionApplyCutoff *Yes* / *No***

Specify whether to apply --SimilarityCutoff or --DistanceCutoff values during application of -g, --GroupFusionRule to reference molecules set. Possible values: *Yes* or *No*. Default value: *Yes*.

During *Yes* value of --GroupFusionApplyCutoff, the reference molecules whose comparison values with a database molecule fall outside specified --SimilarityCutoff or --DistanceCutoff are not used to calculate final fused similarity value between a database molecule and reference molecules set.

**-h, --help**

Print this help message.

**--InDelim *comma* / *semicolon***

Input delimiter for reference and database fingerprints CSV *TextFile(s)*. Possible values: *comma* or *semicolon*. Default value: *comma*. For TSV files, this option is ignored and *tab* is used as a delimiter.

**-k, --kNN *all* / *number***

Number of k-nearest neighbors (k-NN) reference molecules to use during -g, --GroupFusionRule for calculating similarity of a database molecule against a set of reference molecules. Possible values: *all* / *positive integers*. Default: *all*.

After ranking similarity values between a database molecule and reference molecules during *MultipleReferences* value of similarity search -m, --mode option, a top -k, --kNN reference molecule are selected and used during -g, --GroupFusionRule.

This option is -s, --SearchMode dependent: It corresponds to dissimilar molecules during *DissimilaritySearch* value of -s, --SearchMode option.

**-m, --mode *IndividualReference* / *MultipleReferences***

Specify how to treat reference molecules in *ReferenceFingerprintsFile* during similarity search: Treat each reference molecule individually during similarity search or perform similarity search by treating multiple reference molecules as a set. Possible values: *IndividualReference* / *MultipleReferences*. Default value: *MultipleReferences*.

During *IndividualReference* value of -m, --Mode for similarity search, fingerprints bit-vector or vector string of each reference molecule is compared with database molecules using specified similarity or distance coefficients to identify most similar molecules for each reference molecule. Based on value of --SimilarCountMode, upto --n, NumOfSimilarMolecules or -p, --PercentSimilarMolecules at specified <--SimilarityCutoff> or --DistanceCutoff are identified for each reference molecule.

During *MultipleReferences* value -m, --mode for similarity search, all reference molecules are considered as a set and -g, --GroupFusionRule is used to calculate similarity of a database molecule against reference molecules set either using all reference molecules or number of k-nearest neighbors (k-NN) to a database molecule specified using -k, --kNN. The fingerprints bit-vector or vector string of each reference molecule in a set is compared with a database molecule using a similarity or distance coefficient specified via -b, --BitVectorComparisonMode or -v, --VectorComparisonMode. The reference molecules whose comparison values with a database molecule fall outside specified --SimilarityCutoff or --DistanceCutoff are ignored. The specified -g, --GroupFusionRule is applied to rest of -k, --kNN reference molecules to calculate final similarity value between a database molecule and reference molecules set.

The meaning of similarity and distance is automatically reversed during *DissimilaritySearch* value of -s, --SearchMode along with appropriate handling of --SimilarityCutoff or --DistanceCutoff values.

**-n, --NumOfSimilarMolecules *number***

Maximum number of most similar database molecules to find for each reference molecule or set of reference molecules based on *IndividualReference* or *MultipleReferences* value of similarity search -m, --mode option. Default: *10*. Valid values: positive integers.

This option is ignored during *PercentSimilar* value of --SimilarCountMode option.

This option is -s, --SearchMode dependent: It corresponds to dissimilar molecules during *DissimilaritySearch* value of -s, --SearchMode option.

**--OutDelim *comma* / *tab* / *semicolon***

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

--output *SD | text | both*

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

-o, --overwrite

Overwrite existing files

-p, --PercentSimilarMolecules *number*

Maximum percent of mosy similar database molecules to find for each reference molecule or set of reference molecules based on *IndividualReference* or *MultipleReferences* value of similarity search -m, --mode option. Default: 1 percent of database molecules. Valid values: non-zero values in between 0 to 100.

This option is ignored during *NumOfSimilar* value of --SimilarCountMode option.

During *PercentSimilar* value of --SimilarCountMode option, the number of molecules in *DatabaseFingerprintsFile* is counted and number of similar molecules correspond to --PercentSimilarMolecules of the total number of database molecules.

This option is -s, --SearchMode dependent: It corresponds to dissimilar molecules during *DissimilaritySearch* value of -s, --SearchMode option.

--precision *number*

Precision of calculated similarity values for comparison and generating output files. Default: up to 2 decimal places. Valid values: positive integers.

-q, --quote *Yes | No*

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

--ReferenceColMode *ColNum | ColLabel*

Specify how columns are identified in reference fingerprints *TextFile*: using column number or column label. Possible values: *ColNum* or *ColLabel*. Default value: *ColNum*.

--ReferenceCompoundIDCol *col number | col name*

This value is --ReferenceColMode mode specific. It specifies column to use for retrieving compound ID from reference fingerprints *TextFile* during similarity and dissimilarity search for output SD and CSV/TSV text files. Possible values: *col number* or *col label*. Default value: *first column containing the word compoundID in its column label or sequentially generated IDs*.

--ReferenceCompoundIDPrefix *text*

Specify compound ID prefix to use during sequential generation of compound IDs for reference fingerprints *SDFFile* and *TextFile*. Default value: *Cmpd*. The default value generates compound IDs which looks like Cmpd<Number>.

For reference fingerprints *SDFFile*, this value is only used during *LabelPrefix | MolNameOrLabelPrefix* values of --ReferenceCompoundIDMode option; otherwise, it's ignored.

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

Compound

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

--ReferenceCompoundIDField *DataFieldName*

Specify reference fingerprints *SDFFile* datafield label for generating compound IDs. This value is only used during *DataField* value of --ReferenceCompoundIDMode option.

Examples for *DataField* value of --ReferenceCompoundIDMode:

MolID  
ExtReg

--ReferenceCompoundIDMode *DataField | MolName | LabelPrefix | MolNameOrLabelPrefix*

Specify how to generate compound IDs from reference fingerprints *SDFFile* during similarity and dissimilarity search for output SD and CSV/TSV text files: use a *SDFFile* datafield value; use molname line from *SDFFile*; 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: *LabelPrefix*.

For *MolNameAndLabelPrefix* value of --ReferenceCompoundIDMode, molname line in *SDFfiles* takes precedence over sequential compound IDs generated using *LabelPrefix* and only empty molname values are replaced with sequential compound IDs.

--ReferenceFingerprintsCol *col number* | *col name*

This value is --ReferenceColMode specific. It specifies fingerprints column to use during similarity and dissimilarity search for reference fingerprints *TextFile*. Possible values: *col number* or *col label*. Default value: *first column containing the word Fingerprints in its column label*.

--ReferenceFingerprintsField *FieldLabel*

Fingerprints field label to use during similarity and dissimilarity search for reference fingerprints *SDFFile*. Default value: *first data field label containing the word Fingerprints in its label*

-r, --root *RootName*

New file name is generated using the root: <Root>.<Ext>. Default for new file name: <ReferenceFileName>SimilaritySearching.<Ext>. The output file type determines <Ext> value. The sdf, csv, and tsv <Ext> values are used for SD, comma/semicolon, and tab delimited text files respectively.

-s, --SearchMode *SimilaritySearch* | *DissimilaritySearch*

Specify how to find molecules from database molecules for individual reference molecules or set of reference molecules: Find similar molecules or dissimilar molecules from database molecules. Possible values: *SimilaritySearch* | *DissimilaritySearch*. Default value: *SimilaritySearch*.

During *DissimilaritySearch* value of -s, --SearchMode option, the meaning of the following options is switched and they correspond to dissimilar molecules instead of similar molecules: --SimilarCountMode, -n, --NumOfSimilarMolecules, --PercentSimilarMolecules, -k, --kNN.

--SimilarCountMode *NumOfSimilar* | *PercentSimilar*

Specify method used to count similar molecules found from database molecules for individual reference molecules or set of reference molecules: Find number of similar molecules or percent of similar molecules from database molecules. Possible values: *NumOfSimilar* | *PercentSimilar*. Default value: *NumOfSimilar*.

The values for number of similar molecules and percent similar molecules are specified using options -n, NumOfSimilarMolecule and --PercentSimilarMolecules.

This option is -s, --SearchMode dependent: It corresponds to dissimilar molecules during *DissimilaritySearch* value of -s, --SearchMode option.

--SimilarityCutoff *number*

Similarity cutoff value to use during comparison of similarity value between a pair of database and reference molecules calculated by similarity comparison methods for fingerprints bit-vector vector strings data values. Possible values: *Any valid number*. Default value: *0.75*.

The comparison value between a pair of database and reference molecule must meet the cutoff criterion as shown below:

SeachMode	CutoffCriterion	ComparisonValues
Similarity	>=	Higher value implies high similarity
Dissimilarity	<=	Lower value implies high dissimilarity

This option is ignored during *No* value of --GroupFusionApplyCutoff for *MultipleReferences* -m, --mode.

This option is -s, --SearchMode dependent: It corresponds to dissimilar molecules during *DissimilaritySearch* value of -s, --SearchMode option.

-v, --VectorComparisonMode *SupportedSimilarityName* | *SupportedDistanceName*

Specify what similarity or distance coefficient to use for calculating similarity between fingerprint vector strings data values in *ReferenceFingerprintsFile* and *DatabaseFingerprintsFile* during similarity search. Possible values: *TanimotoSimilarity* | ... | *ManhattanDistance* | .... Default value: *TanimotoSimilarity*.

The value of -v, --VectorComparisonMode, in conjunction with --VectorComparisonFormulism, decides which type of similarity and distance coefficient formulism gets used.

The current releases supports the following similarity and distance coefficients: *CosineSimilarity*, *CzekanowskiSimilarity*, *DiceSimilarity*, *OchiaiSimilarity*, *JaccardSimilarity*, *SorensonSimilarity*, *TanimotoSimilarity*,

*CityBlockDistance*, *EuclideanDistance*, *HammingDistance*, *ManhattanDistance*, *SoergelDistance*. These similarity and distance coefficients are described below.

FingerprintsVector.pm module, used to calculate similarity and distance coefficients, provides support to perform comparison between vectors containing three different types of values:

Type I: OrderedNumericalValues

- . Size of two vectors are same
- . Vectors contain real values in a specific order. For example: MACCS keys count, Topological pharmacophore atom pairs and so on.

Type II: UnorderedNumericalValues

- . Size of two vectors might not be same
- . Vectors contain unordered real value identified by value IDs. For example: Topological atom pairs, Topological atom torsions and so on

Type III: AlphaNumericalValues

- . Size of two vectors might not be same
- . Vectors contain unordered alphanumerical values. For example: Extended connectivity fingerprints, atom neighborhood fingerprints.

Before performing similarity or distance calculations between vectors containing UnorderedNumericalValues or AlphaNumericalValues, the vectors are transformed into vectors containing unique OrderedNumericalValues using value IDs for UnorderedNumericalValues and values itself for AlphaNumericalValues.

Three forms of similarity and distance calculation between two vectors, specified using --VectorComparisonFormulism option, are supported: *AlgebraicForm*, *BinaryForm* or *SetTheoreticForm*.

For *BinaryForm*, the ordered list of processed final vector values containing the value or count of each unique value type is simply converted into a binary vector containing 1s and 0s corresponding to presence or absence of values before calculating similarity or distance between two vectors.

For two fingerprint vectors A and B of same size containing OrderedNumericalValues, let:

$N$  = Number values in A or B

$X_a$  = Values of vector A

$X_b$  = Values of vector B

$X_{ai}$  = Value of  $i$ th element in A

$X_{bi}$  = Value of  $i$ th element in B

$SUM$  = Sum of  $i$  over  $N$  values

For SetTheoreticForm of calculation between two vectors, let:

$SetIntersectionX_aX_b = SUM ( MIN ( X_{ai}, X_{bi} ) )$

$SetDifferenceX_aX_b = SUM ( X_{ai} ) + SUM ( X_{bi} ) - SUM ( MIN ( X_{ai}, X_{bi} ) )$

For BinaryForm of calculation between two vectors, let:

$N_a$  = Number of bits set to "1" in A =  $SUM ( X_{ai} )$

$N_b$  = Number of bits set to "1" in B =  $SUM ( X_{bi} )$

$N_c$  = Number of bits set to "1" in both A and B =  $SUM ( X_{ai} * X_{bi} )$

$N_d$  = Number of bits set to "0" in both A and B

$= SUM ( 1 - X_{ai} - X_{bi} + X_{ai} * X_{bi} )$

$N$  = Number of bits set to "1" or "0" in A or B = Size of A or B =  $N_a + N_b - N_c + N_d$

Additionally, for BinaryForm various values also correspond to:

$N_a = | X_a |$

$N_b = | X_b |$

$N_c = | SetIntersectionX_aX_b |$

$N_d = N - | SetDifferenceX_aX_b |$

$| SetDifferenceX_aX_b | = N - N_d = N_a + N_b - N_c + N_d - N_d = N_a + N_b - N_c$

$$= |X_a| + |X_b| - |\text{SetIntersection}X_aX_b|$$

Various similarity and distance coefficients [ Ref 40, Ref 62, Ref 64 ] for a pair of vectors A and B in *AlgebraicForm*, *BinaryForm* and *SetTheoreticForm* are defined as follows:

CityBlockDistance: ( same as HammingDistance and ManhattanDistance)

*AlgebraicForm*:  $\text{SUM} ( \text{ABS} ( X_{ai} - X_{bi} ) )$

*BinaryForm*:  $( N_a - N_c ) + ( N_b - N_c ) = N_a + N_b - 2 * N_c$

*SetTheoreticForm*:  $|\text{SetDifference}X_aX_b| - |\text{SetIntersection}X_aX_b| = \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) )$

CosineSimilarity: ( same as OchiaiSimilarityCoefficient)

*AlgebraicForm*:  $\text{SUM} ( X_{ai} * X_{bi} ) / \text{SQRT} ( \text{SUM} ( X_{ai} ** 2 ) * \text{SUM} ( X_{bi} ** 2 ) )$

*BinaryForm*:  $N_c / \text{SQRT} ( N_a * N_b )$

*SetTheoreticForm*:  $|\text{SetIntersection}X_aX_b| / \text{SQRT} ( |X_a| * |X_b| ) = \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) / \text{SQRT} ( \text{SUM} ( X_{ai} ) * \text{SUM} ( X_{bi} ) )$

CzekanowskiSimilarity: ( same as DiceSimilarity and SorensonSimilarity)

*AlgebraicForm*:  $( 2 * ( \text{SUM} ( X_{ai} * X_{bi} ) ) ) / ( \text{SUM} ( X_{ai} ** 2 ) + \text{SUM} ( X_{bi} ** 2 ) )$

*BinaryForm*:  $2 * N_c / ( N_a + N_b )$

*SetTheoreticForm*:  $2 * |\text{SetIntersection}X_aX_b| / ( |X_a| + |X_b| ) = 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) ) / ( \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) )$

DiceSimilarity: ( same as CzekanowskiSimilarity and SorensonSimilarity)

*AlgebraicForm*:  $( 2 * ( \text{SUM} ( X_{ai} * X_{bi} ) ) ) / ( \text{SUM} ( X_{ai} ** 2 ) + \text{SUM} ( X_{bi} ** 2 ) )$

*BinaryForm*:  $2 * N_c / ( N_a + N_b )$

*SetTheoreticForm*:  $2 * |\text{SetIntersection}X_aX_b| / ( |X_a| + |X_b| ) = 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) ) / ( \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) )$

EuclideanDistance:

*AlgebraicForm*:  $\text{SQRT} ( \text{SUM} ( ( X_{ai} - X_{bi} ) ** 2 ) )$

*BinaryForm*:  $\text{SQRT} ( ( N_a - N_c ) + ( N_b - N_c ) ) = \text{SQRT} ( N_a + N_b - 2 * N_c )$

*SetTheoreticForm*:  $\text{SQRT} ( |\text{SetDifference}X_aX_b| - |\text{SetIntersection}X_aX_b| ) = \text{SQRT} ( \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) ) )$

HammingDistance: ( same as CityBlockDistance and ManhattanDistance)

*AlgebraicForm*:  $\text{SUM} ( \text{ABS} ( X_{ai} - X_{bi} ) )$

*BinaryForm*:  $( N_a - N_c ) + ( N_b - N_c ) = N_a + N_b - 2 * N_c$

*SetTheoreticForm*:  $|\text{SetDifference}X_aX_b| - |\text{SetIntersection}X_aX_b| = \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) )$

JaccardSimilarity: ( same as TanimotoSimilarity)

*AlgebraicForm*:  $\text{SUM} ( X_{ai} * X_{bi} ) / ( \text{SUM} ( X_{ai} ** 2 ) + \text{SUM} ( X_{bi} ** 2 ) - \text{SUM} ( X_{ai} * X_{bi} ) )$

*BinaryForm*:  $N_c / ( ( N_a - N_c ) + ( N_b - N_c ) + N_c ) = N_c / ( N_a + N_b - N_c )$

*SetTheoreticForm*:  $|\text{SetIntersection}X_aX_b| / |\text{SetDifference}X_aX_b| = \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) / ( \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) )$

ManhattanDistance: ( same as CityBlockDistance and HammingDistance)

*AlgebraicForm*:  $\text{SUM} ( \text{ABS} ( X_{ai} - X_{bi} ) )$

*BinaryForm*:  $( N_a - N_c ) + ( N_b - N_c ) = N_a + N_b - 2 * N_c$

*SetTheoreticForm*:  $|\text{SetDifference}X_aX_b| - |\text{SetIntersection}X_aX_b| = \text{SUM} ( X_{ai} ) + \text{SUM} ( X_{bi} ) - 2 * ( \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) )$

OchiaiSimilarity: ( same as CosineSimilarity)

*AlgebraicForm*:  $\text{SUM} ( X_{ai} * X_{bi} ) / \text{SQRT} ( \text{SUM} ( X_{ai} ** 2 ) * \text{SUM} ( X_{bi} ** 2 ) )$

*BinaryForm*:  $N_c / \text{SQRT} ( N_a * N_b )$

*SetTheoreticForm*:  $|\text{SetIntersection}X_aX_b| / \text{SQRT} ( |X_a| * |X_b| ) = \text{SUM} ( \text{MIN} ( X_{ai}, X_{bi} ) ) / \text{SQRT} ( \text{SUM} ( X_{ai} ) * \text{SUM} ( X_{bi} ) )$

SorensonSimilarity: ( same as CzekanowskiSimilarity and DiceSimilarity)

*AlgebraicForm*:  $(2 * (\text{SUM} ( \text{Xai} * \text{Xbi} ) ) ) / ( \text{SUM} ( \text{Xai} ** 2 ) + \text{SUM} ( \text{Xbi} ** 2 ) )$

*BinaryForm*:  $2 * \text{Nc} / ( \text{Na} + \text{Nb} )$

*SetTheoreticForm*:  $2 * | \text{SetIntersectionXaXb} | / ( | \text{Xa} | + | \text{Xb} | ) = 2 * ( \text{SUM} ( \text{MIN} ( \text{Xai}, \text{Xbi} ) ) ) / ( \text{SUM} ( \text{Xai} ) + \text{SUM} ( \text{Xbi} ) )$

SoergelDistance:

*AlgebraicForm*:  $\text{SUM} ( \text{ABS} ( \text{Xai} - \text{Xbi} ) ) / \text{SUM} ( \text{MAX} ( \text{Xai}, \text{Xbi} ) )$

*BinaryForm*:  $1 - \text{Nc} / ( \text{Na} + \text{Nb} - \text{Nc} ) = ( \text{Na} + \text{Nb} - 2 * \text{Nc} ) / ( \text{Na} + \text{Nb} - \text{Nc} )$

*SetTheoreticForm*:  $( | \text{SetDifferenceXaXb} | - | \text{SetIntersectionXaXb} | ) / | \text{SetDifferenceXaXb} | = ( \text{SUM} ( \text{Xai} ) + \text{SUM} ( \text{Xbi} ) - 2 * ( \text{SUM} ( \text{MIN} ( \text{Xai}, \text{Xbi} ) ) ) ) / ( \text{SUM} ( \text{Xai} ) + \text{SUM} ( \text{Xbi} ) - \text{SUM} ( \text{MIN} ( \text{Xai}, \text{Xbi} ) ) )$

TanimotoSimilarity: ( same as JaccardSimilarity)

*AlgebraicForm*:  $\text{SUM} ( \text{Xai} * \text{Xbi} ) / ( \text{SUM} ( \text{Xai} ** 2 ) + \text{SUM} ( \text{Xbi} ** 2 ) - \text{SUM} ( \text{Xai} * \text{Xbi} ) )$

*BinaryForm*:  $\text{Nc} / ( ( \text{Na} - \text{Nc} ) + ( \text{Nb} - \text{Nc} ) + \text{Nc} ) = \text{Nc} / ( \text{Na} + \text{Nb} - \text{Nc} )$

*SetTheoreticForm*:  $| \text{SetIntersectionXaXb} | / | \text{SetDifferenceXaXb} | = \text{SUM} ( \text{MIN} ( \text{Xai}, \text{Xbi} ) ) / ( \text{SUM} ( \text{Xai} ) + \text{SUM} ( \text{Xbi} ) - \text{SUM} ( \text{MIN} ( \text{Xai}, \text{Xbi} ) ) )$

--VectorComparisonFormulism *AlgebraicForm* | *BinaryForm* | *SetTheoreticForm*

Specify fingerprints vector comparison formulism to use for calculation similarity and distance coefficients during -v, --VectorComparisonMode. Possible values: *AlgebraicForm* | *BinaryForm* | *SetTheoreticForm*. Default value: *AlgebraicForm*.

For fingerprint vector strings containing AlphaNumericalValues data values -

ExtendedConnectivityFingerprints, AtomNeighborhoodsFingerprints and so on - all three formulism result in same value during similarity and distance calculations.

-w, --WorkingDir *DirName*

Location of working directory. Default: current directory.

## EXAMPLES

To perform similarity search using Tanimoto coefficient by treating all reference molecules as a set to find 10 most similar database molecules with application of Max group fusion rule and similarity cutoff to supported fingerprints strings data in SD fingerprints files present in a data fields with Fingerprint substring in their labels, and create a ReferenceFPHexSimilaritySearching.csv file containing sequentially generated database compound IDs with Cmpd prefix, type:

```
% SimilaritySearchingFingerprints.pl -o ReferenceSampleFPHex.sdf
DatabaseSampleFPHex.sdf
```

To perform similarity search using Tanimoto coefficient by treating all reference molecules as a set to find 10 most similar database molecules with application of Max group fusion rule and similarity cutoff to supported fingerprints strings data in FP fingerprints files, and create a SimilaritySearchResults.csv file containing database compound IDs retrieved from FP file, type:

```
% SimilaritySearchingFingerprints.pl -r SimilaritySearchResults -o
ReferenceSampleFPBin.fpf DatabaseSampleFPBin.fpf
```

To perform similarity search using Tanimoto coefficient by treating all reference molecules as a set to find 10 most similar database molecules with application of Max group fusion rule and similarity cutoff to supported fingerprints strings data in text fingerprints files present in a column names containing Fingerprint substring in their names, and create a ReferenceFPHexSimilaritySearching.csv file containing database compound IDs retrieved column name containing CompoundID substring or sequentially generated compound IDs, type:

```
% SimilaritySearchingFingerprints.pl -o ReferenceSampleFPCount.csv
DatabaseSampleFPCount.csv
```

To perform similarity search using Tanimoto coefficient by treating reference molecules as individual molecules to find 10 most similar database molecules for each reference molecule with application of similarity cutoff to supported fingerprints strings data in SD fingerprints files present in a data fields with Fingerprint substring in their labels, and create a ReferenceFPHexSimilaritySearching.csv file containing sequentially generated reference and



database compound IDs with Cmpd prefix, type:

```
% SimilaritySearchingFingerprints.pl -mode IndividualReference -o  
ReferenceSampleFPHex.sdf DatabaseSampleFPHex.sdf
```

To perform similarity search using Tanimoto coefficient by treating reference molecules as individual molecules to find 10 most similar database molecules for each reference molecule with application of similarity cutoff to supported fingerprints strings data in FP fingerprints files, and create a ReferenceFPHexSimilaritySearching.csv file containing references and database compound IDs retrieved from FP file, type:

```
% SimilaritySearchingFingerprints.pl -mode IndividualReference -o  
ReferenceSampleFPHex.fpf DatabaseSampleFPHex.fpf
```

To perform similarity search using Tanimoto coefficient by treating reference molecules as individual molecules to find 10 most similar database molecules for each reference molecule with application of similarity cutoff to supported fingerprints strings data in text fingerprints files present in a column names containing Fingerprint substring in their names, and create a ReferenceFPHexSimilaritySearching.csv file containing reference and database compound IDs retrieved column name containing CompoundID substring or sequentially generated compound IDs, type:

```
% SimilaritySearchingFingerprints.pl -mode IndividualReference -o  
ReferenceSampleFPHex.csv DatabaseSampleFPHex.csv
```

To perform dissimilarity search using Tanimoto coefficient by treating all reference molecules as a set to find 10 most dissimilar database molecules with application of Max group fusion rule and similarity cutoff to supported fingerprints strings data in SD fingerprints files present in a data fields with Fingerprint substring in their labels, and create a ReferenceFPHexSimilaritySearching.csv file containing sequentially generated database compound IDs with Cmpd prefix, type:

```
% SimilaritySearchingFingerprints.pl --mode MultipleReferences --SearchMode  
DissimilaritySearch -o ReferenceSampleFPHex.sdf DatabaseSampleFPHex.sdf
```

To perform similarity search using CityBlock distance by treating reference molecules as individual molecules to find 10 most similar database molecules for each reference molecule with application of distance cutoff to supported vector fingerprints strings data in SD fingerprints files present in a data fields with Fingerprint substring in their labels, and create a ReferenceFPHexSimilaritySearching.csv file containing sequentially generated reference and database compound IDs with Cmpd prefix, type:

```
% SimilaritySearchingFingerprints.pl -mode IndividualReference  
--VectorComparisonMode CityBlockDistance --VectorComparisonFormulism  
AlgebraicForm --DistanceCutoff 10 -o  
ReferenceSampleFPCount.sdf DatabaseSampleFPCount.sdf
```

To perform similarity search using Tanimoto coefficient by treating all reference molecules as a set to find 100 most similar database molecules with application of Mean group fusion rule to top 10 reference molecules with in similarity cutoff of 0.75 to supported fingerprints strings data in FP fingerprints files, and create a ReferenceFPHexSimilaritySearching.csv file containing database compound IDs retrieved from FP file, type:

```
% SimilaritySearchingFingerprints.pl --mode MultipleReferences --SearchMode  
SimilaritySearch --BitVectorComparisonMode TanimotoSimilarity  
--GroupFusionRule Mean --GroupFusionApplyCutoff Yes --kNN 10  
--SimilarityCutoff 0.75 --SimilarCountMode NumOfSimilar  
--NumOfSimilarMolecules 100 -o  
ReferenceSampleFPHex.fpf DatabaseSampleFPHex.fpf
```

To perform similarity search using Tanimoto coefficient by treating reference molecules as individual molecules to find 2 percent of most similar database molecules for each reference molecule with application of similarity cutoff of 0.85 to supported fingerprints strings data in text fingerprints files present in specific columns and create a ReferenceFPHexSimilaritySearching.csv file containing reference and database compoundIDs retrieved from specific columns, type:

```
% SimilaritySearchingFingerprints.pl --mode IndividualReference --SearchMode  
SimilaritySearch --BitVectorComparisonMode TanimotoSimilarity
```

```
--ReferenceColMode Collabel --ReferenceFingerprintsCol Fingerprints
--ReferenceCompoundIDCol CompoundID --DatabaseColMode Collabel
--DatabaseCompoundIDCol CompoundID --DatabaseFingerprintsCol
Fingerprints --SimilarityCutoff 0.85 --SimilarCountMode PercentSimilar
--PercentSimilarMolecules 2 -o
ReferenceSampleFPHex.csv DatabaseSampleFPHex.csv
```

To perform similarity search using Tanimoto coefficient by treating reference molecules as individual molecules to find top 50 most similar database molecules for each reference molecule with application of similarity cutoff of 0.85 to supported fingerprints strings data in SD fingerprints files present in specific data fields and create both ReferenceFPHexSimilaritySearching.csv and ReferenceFPHexSimilaritySearching.sdf files containing reference and database compoundIDs retrieved from specific data fields, type:

```
% SimilaritySearchingFingerprints.pl --mode IndividualReference --SearchMode
SimilaritySearch --BitVectorComparisonMode TanimotoSimilarity
--ReferenceFingerprintsField Fingerprints
--DatabaseFingerprintsField Fingerprints
--ReferenceCompoundIDMode DataField --ReferenceCompoundIDField CmpdID
--DatabaseCompoundIDMode DataField --DatabaseCompoundIDField CmpdID
--SimilarityCutoff 0.85 --SimilarCountMode NumOfSimilar
--NumOfSimilarMolecules 50 --output both -o
ReferenceSampleFPHex.sdf DatabaseSampleFPHex.sdf
```

To perform similarity search using Tanimoto coefficient by treating reference molecules as individual molecules to find 1 percent of most similar database molecules for each reference molecule with application of similarity cutoff to supported fingerprints strings data in SD fingerprints files present in specific data field labels, and create both ReferenceFPHexSimilaritySearching.csv ReferenceFPHexSimilaritySearching.sdf files containing reference and database compound IDs retrieved from specific data field labels along with other specific data for database molecules, type:

```
% SimilaritySearchingFingerprints.pl --mode IndividualReference --SearchMode
SimilaritySearch --BitVectorComparisonMode TanimotoSimilarity
--ReferenceFingerprintsField Fingerprints
--DatabaseFingerprintsField Fingerprints
--ReferenceCompoundIDMode DataField --ReferenceCompoundIDField CmpdID
--DatabaseCompoundIDMode DataField --DatabaseCompoundIDField CmpdID
--DatabaseDataFieldsMode Specify --DatabaseDataFields "TPSA,SLogP"
--SimilarityCutoff 0.75 --SimilarCountMode PercentSimilar
--PercentSimilarMolecules 1 --output both --OutDelim comma --quote Yes
--precision 3 -o ReferenceSampleFPHex.sdf DatabaseSampleFPHex.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, TopologicalPharmacophoreAtomPairsFingerprints.pl, TopologicalPharmacophoreAtomTripletsFingerprints.pl

## COPYRIGHT

Copyright (C) 2018 Manish Sud. All rights reserved.

This file is part of MayaChemTools.

MayaChemTools is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.