CLUSTERING, SAMPLING AND FILTERING ULTRA-LARGE CHEMICAL DATABASES FOR VIRTUAL SCREENING: IT'S NOT JUST WHAT YOU DO, BUT THE WHEN THAT YOU DO IT.

Roger Sayle and John Mayfield

NextMove Software, Cambridge, UK

13th RDKit UGM, ETH, Zurich, Switzerland, 12th September 2024

OVERVIEW

- In cheminformatics, “ultra-large” refers to chemical databases of over 1 billion compounds.
- The immense size of these databases cause performance problems for traditional virtual screening workflows.
- This is more serious with the use of the cloud, where inefficiency becomes a high financial cost, and not just an inconvenient delay.
- Fortunately, things can often be improved by reordering steps and using efficient algorithms.

SAMPLING AND FILTERING.

- Normally, you should filter a database before performing clustering, i.e. PAINS filtering before MaxMin, or MW before docking.
- However, diversity selection for very small percentages is better done by random sampling, where it's then better to filter after/during the selection process.

Pre-sorting a database

- How to sort result set from a (SMARTS) substructure screen by Tanimoto/relevance?
- Simply pre-sort the database by the number of bits in each fingerprint, or highly correlated by molecular weight or heavy atom count.
- Results of searches now contain the “smallest” hit first, closest to the query, with results decreasing in Tanimoto/relevance such that the largest hits appear last.

RANDOMLY PERMUTING DATABASES

- How would you select 1000 random boronic acids from Enamine building blocks, or 100 random steroids from ChEMBL.
- One approach is to randomly permute the database in advance, and then retrieve the first N substructure hits.

THE PROBLEM WITH INFINITY

- Clustering an infinite data set (typically) produces an infinite number of clusters and an infinite number of singletons.

FALLING AT THE FIRST HURDLE

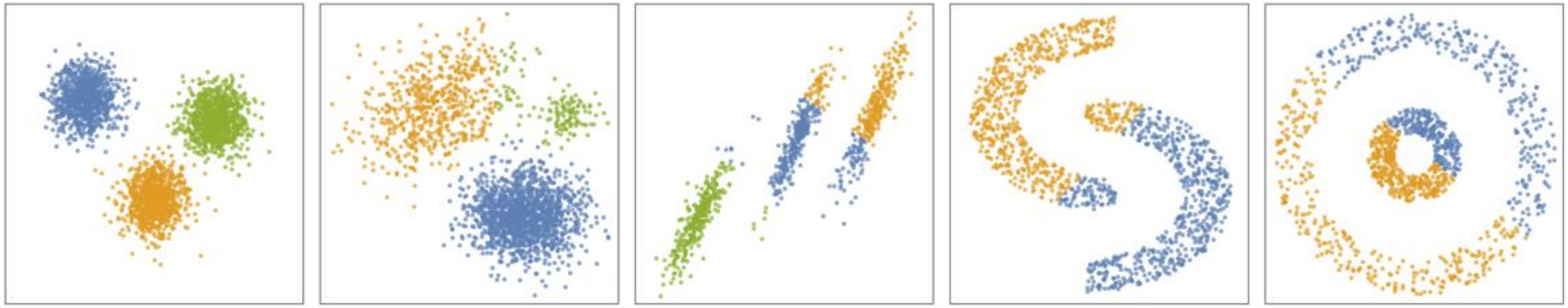
- Any clustering algorithm that starts with “Step 1. Calculate the full NxN distance matrix” is doomed to be inappropriate.
- Often using an NxN distance matrix as input is just a poor implementation rather than a requirement of the algorithm
  - RDKit’s legacy Butina clustering implementation (from 2008) had quadratic behavior and was obsoleted by the vastly more efficient LeaderPicker implementation in 2019.
    - Sayle, “2D similarity, Diversity and Clustering in RDKit”, presented at the 8th RDKit UGM, Hamburg, 2019.

LIMITATIONS OF K-MEANS CLUSTERING

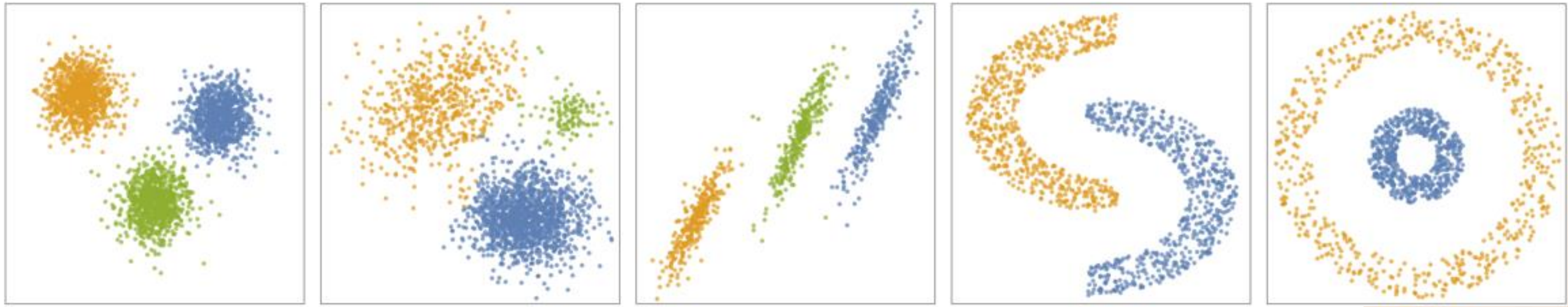
- An increasingly popular choice in cheminformatics is to use the K-means clustering, often in combination with Silhouette score to determine K, the number of clusters.
- These methods have the benefit of being (relatively) computationally efficient O(n), and the convenience of scikit-learn python implementations.
- Unfortunately, these approaches perform poorly on chemical data sets using standard similarity measures (such as fingerprint Tanimoto).

FROM WOLFRAM DOCUMENTATION...

- **KMeans**
- "KMeans" is a classic, simple, centroid-based clustering method. "KMeans" works when clusters have similar sizes and are locally and isotropically distributed around their centroid. When clusters have very different sizes, are anisotropic, are intertwined, or when outliers are present, it is likely that "KMeans" will give poor results.

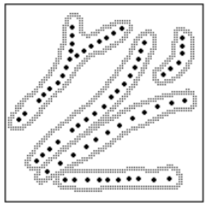


- **JarvisPatrick**

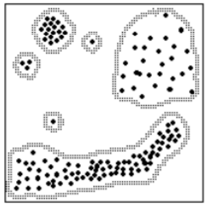


FROM DAYLIGHT DOCUMENTATION…

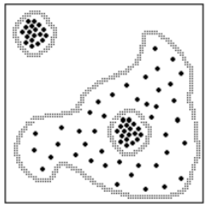
- Chemical structure clustering places extremely tough demands on a clustering algorithm.



- Many clustering algorithms are biased towards finding globular clusters. Such algorithms are not suitable for chemical clustering, where long "stringy" clusters are the rule, not the exception.



- To be effective for clustering chemical structures, a clustering algorithm must be self-scaling, since it is expected to find both straggly, diverse clusters (e.g. penicillins) and tight ones (e.g. PCB's).



- A chemical clustering method must be adjustable in some way that allows control of the tightness required to cluster (e.g., do the estradiols form a separate cluster within the steroid cluster or not?)

- Willett published several thorough analyses and comparisons of various clustering algorithms to which the user is referred for more information (see References). One of his conclusions was that the Jarvis-Patrick method performed best, but that it was computationally prohibitive for large data sets.

LIMITATIONS OF SILHOUETTE

- Silhouette score is a measure of how “well” a data set is clustered (for globular/spherical) clusters.

$$s(i) = \frac{average\ intra - cluster\ distance}{average\ distance\ to\ nearest\ cluster}$$

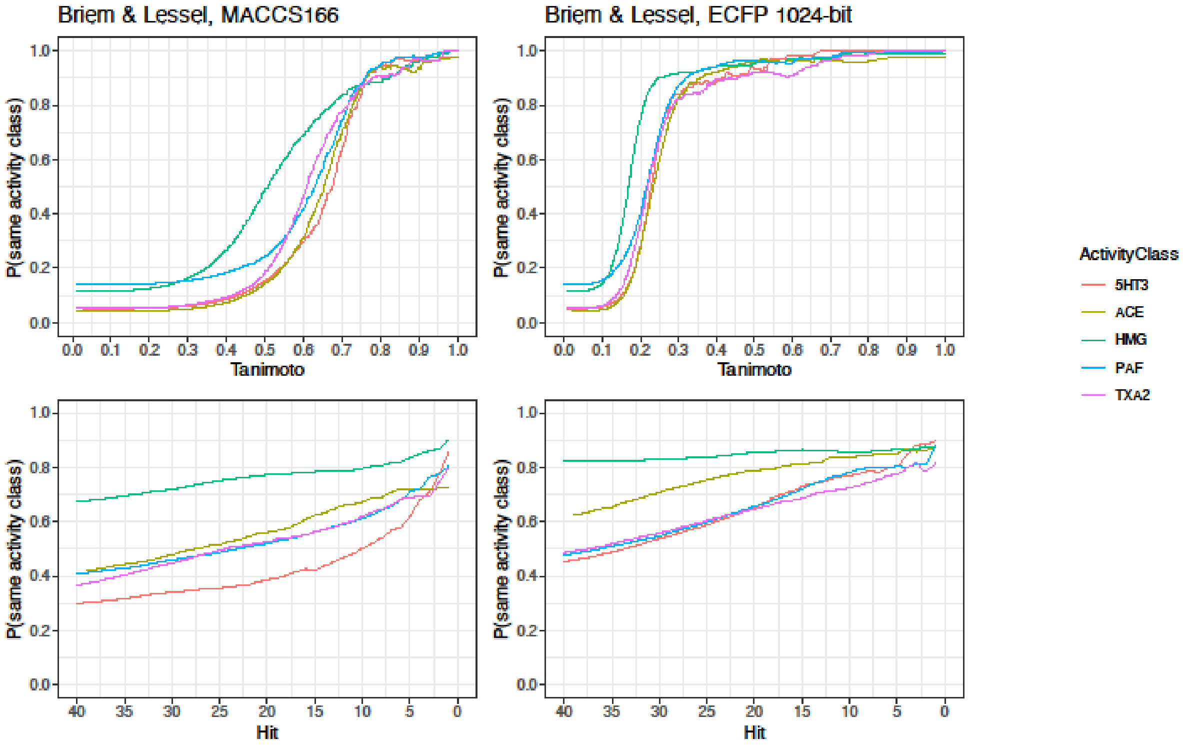
- To quote David Weininger, “Folks use the number of singletons to assess clustering methods; The number of singletons in your data is a property of your data set, not the quality of a clustering algorithm”.
- Silhouette score isn’t (just) how well your data is clustered, but a measures of how separably globular it is.

AVERAGE (MEAN) ISN'T USEFUL.

- The bulk behavior of binary fingerprints can be accurately predicted from bit frequencies alone, i.e. they behave like random independent variables.
- The average score is therefore more dependent upon the fingerprint method used, the FP length and the similarity metric than it is composition/content of the database.
- This is a fundamental challenge with high dimensional spaces, distances become more similar, called the “curse of dimensionality”.

THE SIMILARITY PRINCIPLE

- ECFP & Tanimoto are used for their ability to recognize actives



- The slope of this curve is important in classification, and not its location.
- The take home is that 0.35 is a reasonable threshold for ECFP4 (and 0.7 is not!).

GLOBULARITY

	ACE	TXA2	HMG	PAF	5HT3	ALL
ACE	0.106 0.301 0.863	0.047 0.160 0.314	0.043 0.135 0.329	0.02 0.144 0.318	0.055 0.160 0.341	0.02 0.146 1.0
TXA2	0.047 0.160 0.314	0.059 0.221 0.847	0.027 0.126 0.325	0.012 0.128 0.310	0.043 0.133 0.286	0.008 0.126 0.78
HMG COA	0.043 0.135 0.329	0.027 0.126 0.325	0.031 0.243 1.0	0.016 0.117 0.275	0.02 0.129 0.243	0.008 0.120 0.902
PAF	0.02 0.144 0.318	0.012 0.128 0.310	0.016 0.117 0.275	0.002 0.165 1.0	0.02 0.144 0.337	0.0 0.126 0.953
5HT	0.055 0.160 0.341	0.043 0.133 0.286	0.02 0.129 0.243	0.02 0.144 0.337	0.071 0.215 0.941	0.008 0.137 0.651
ALL	0.02 0.146 1.0	0.008 0.126 0.78	0.008 0.120 0.902	0.0 0.126 0.953	0.008 0.137 0.651	0.0 0.12 0.788

Min

Avg

Max

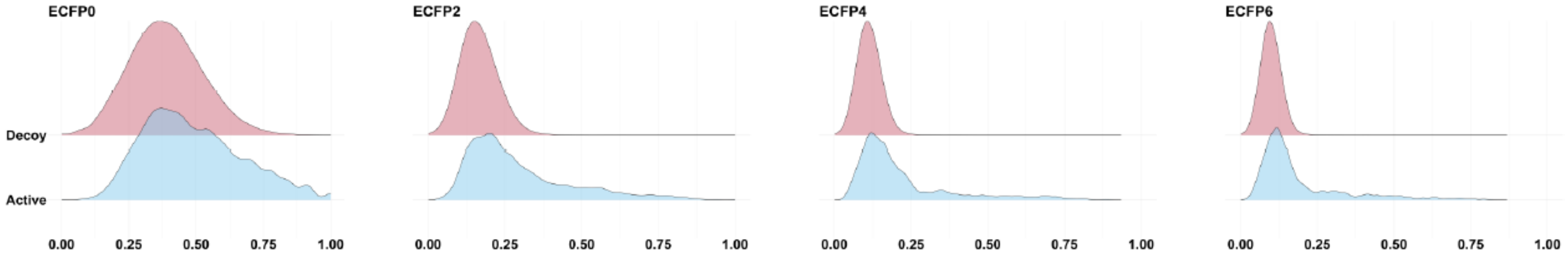
ECFP4

1024-bit

Tanimoto

THINKING VIRTUAL SCREENING DOESN'T WORK

- 2D similarity search is a myopic fish-eye lens.
- Things appear equally far as Tanimotos approach zero.
- Misunderstanding this using AUC, enrichment at N%, retrieval of 50% actives, etc. can lead to incorrect conclusions and poor protocols.



- Fig 1. from Venkatramen et al. (2024) “Do Molecular Fingerprints Identify Diverse Active Drugs in Large-Scale Virtual Screening? (No)”, Pharmaceticals,17:992.

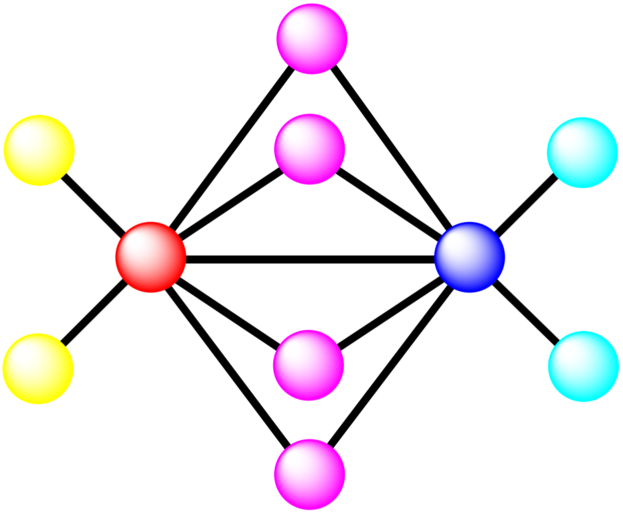
JARVIS-PATRICK CLUSTERING

- So obscure, it doesn't have a Wikipedia entry.
- A density-based clustering method base on shared nearest neighbors.
- Conceptually related to DBSCAN clustering.

- R.A. Jarvis and E.A. Patrick, "Clustering Using a Similarity Measure Based on Shared Near Neighbors" IEEE Trans. Comput. 22(11):1025-1034, Nov. 1973.

THE JARVIS-PATRICK CRITERION

- Jarvis-Patrick cluster has two parameters P and Q.
- Two items are in the same cluster if they are in each others' Q-nearest neighbor lists, and have P or more Q-nearest neighbors in common.



Red and Blue are in the same cluster for (P,Q) = (4,7) but not (directly) connected for (P,Q) = (5,7).

Jarvis-Patrick-Mayfield-Sayle (#1)

- It is possible to determine the Jarvis-Patrick cluster membership for the cluster containing X, by performing a graph traversal starting at X.
- Historically, this step has been done using a “Union-Join” algorithm to represent/find disjoint sets, to find all clusters.

Jarvis-Patrick-Mayfield-Sayle (#2)

Input: Database D, Query $q \in D$, P,Q:int

```
cluster = {q}
queue = {q}
nbors[q] ← getNeighbors(q,D,Q)
while (queue ≠ {})
    i = pop from q
    for j in nbors[i]
        if (nbors[j] = {})
            nbors[j] ← getNeighbors(j,D,Q)
        if $j \notin cluster \land i \in nbors[j] \land |nbors[i] \cap nbors[j]| \geq P$
            cluster ← cluster ∪ {j}
            push j on queue
return cluster
```

JARVIS-PATRICK IS HIERARCHICAL

- Clustering at (P,Q) implies clustering at (P-1,Q)

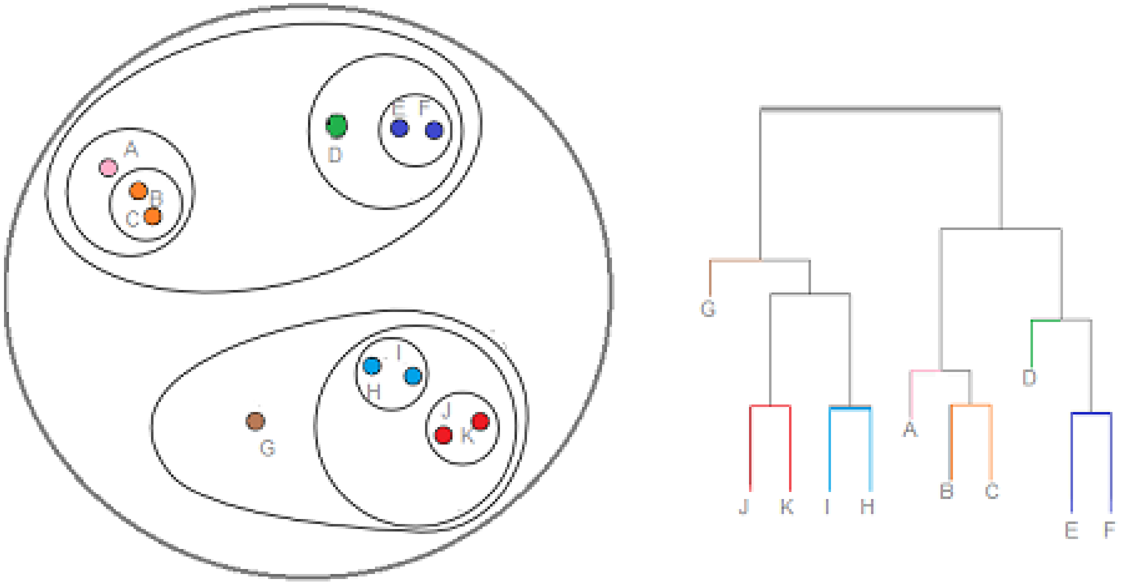


Image Credit: statisticshowto.com/hierarchical-clustering

- This can be used to create a non-parametric ordering
  - A's neighbors are {{B,C},{D,E,F},{G,H,I,J,K}}.

TURBO SIMILARITY SEARCHING

- Jarvis-Patrick ranking is related to the Willett group's "Turbo Similarity Searching" (TSS).
- TSS combines the results of a search with results of the searches of the top-Q neighbors.
- Named from the automotive "turbocharger" where the exhaust/result is used to improve performance.

1. Hert et al., "Enhancing the effectiveness of similarity-based virtual screening using nearest-neighbor information", J. Med. Chem, 48(22):7049-7054, 2005.
2. Hert et al., "New methods for ligand-based virtual screening: use of data fusion and machine learning to enhance the effectiveness of similarity searching", JCIM, 46(2):462-470, 2006.
3. https://www.daylight.com/meetings/emug05/Willet/Presentation.pdf

RESULTS: QUALITY

Method	Total	Recall
Tanimoto	79.89%	100%
Turbo (5)	83.89%	100%
Turbo (10)	83.84%	100%
Turbo (30)	83.79%	100%
J-P (5,10)	88.53%	69.26%
J-P (10,15)	92.08%	61.16%
J-P (10,20)	91.39%	84.08%
J-P (5,30)	83.42%	100%
J-P (0,25)	82.92%	100%

RESULTS: PERFORMANCE

Query	DB	(P,Q)	Size	Searchs	Time
Abilify	ChEMBL	(8,10)	1	11	0.012s
Abilify	ChEMBL	(6,10)	14	33	0.034s
Abilify	ChEMBL	(5,10)	44	69	0.034s
Abilify	ChEMBL	(4,10)	94	142	0.152s
Abilify	ChEMBL	(10,20)	161	225	0.260s
Celebrex	ChEMBL	(8,10)	2	11	0.010s
Celebrex	ChEMBL	(6,10)	3	14	0.016s
Celebrex	ChEMBL	(5,10)	12	33	0.034s
Celebrex	ChEMBL	(4,10)	49	80	0.111s
Celebrex	ChEMBL	(10,20)	84	183	0.197s

Times on M1 MacBook Pro (-j 10) vs. ChEMBL 34.

FURTHER PERFORMANCE ADVANTAGES

- The above algorithm batches together searches of a small number of highly similar queries.
- This improves access locality (e.g. in SmallWorld).
- These searches can be performed in a single scan.
- In the latest generation of search engines, such as NextMove Software’s Arthor, the similarity of queries can be used to reduce memory accesses and improve performance.

SUMMARY

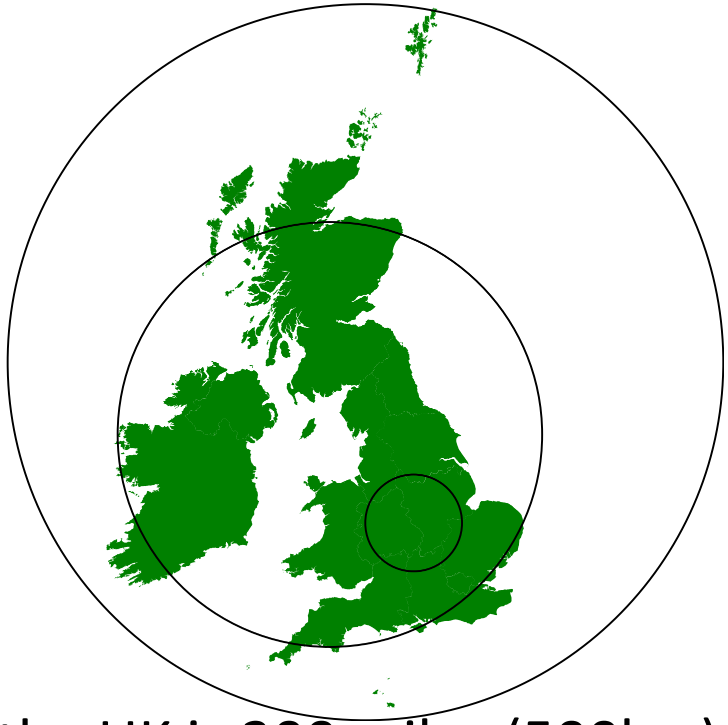
- Adapting traditional workflows to only cluster/filter what you need (when you need it) allows processing of much larger data sets than previously possible.

ACKNOWLEDGEMENTS

- The Team at NextMove Software
  - John Mayfield
  - Ingvar Lagerstedt
  - Rachel Pirie
  - Michael Blakey
  - Zayyan Masud

- David Weininger, Pat Walters and Greg Landrum.

GEOGRAPHICAL ANALOGY



- At its widest the UK is 300 miles (500km) across.
- From North Scotland to Southern Coast is 600 miles (1000km).
- No part is more than 75miles (120km) from the sea.