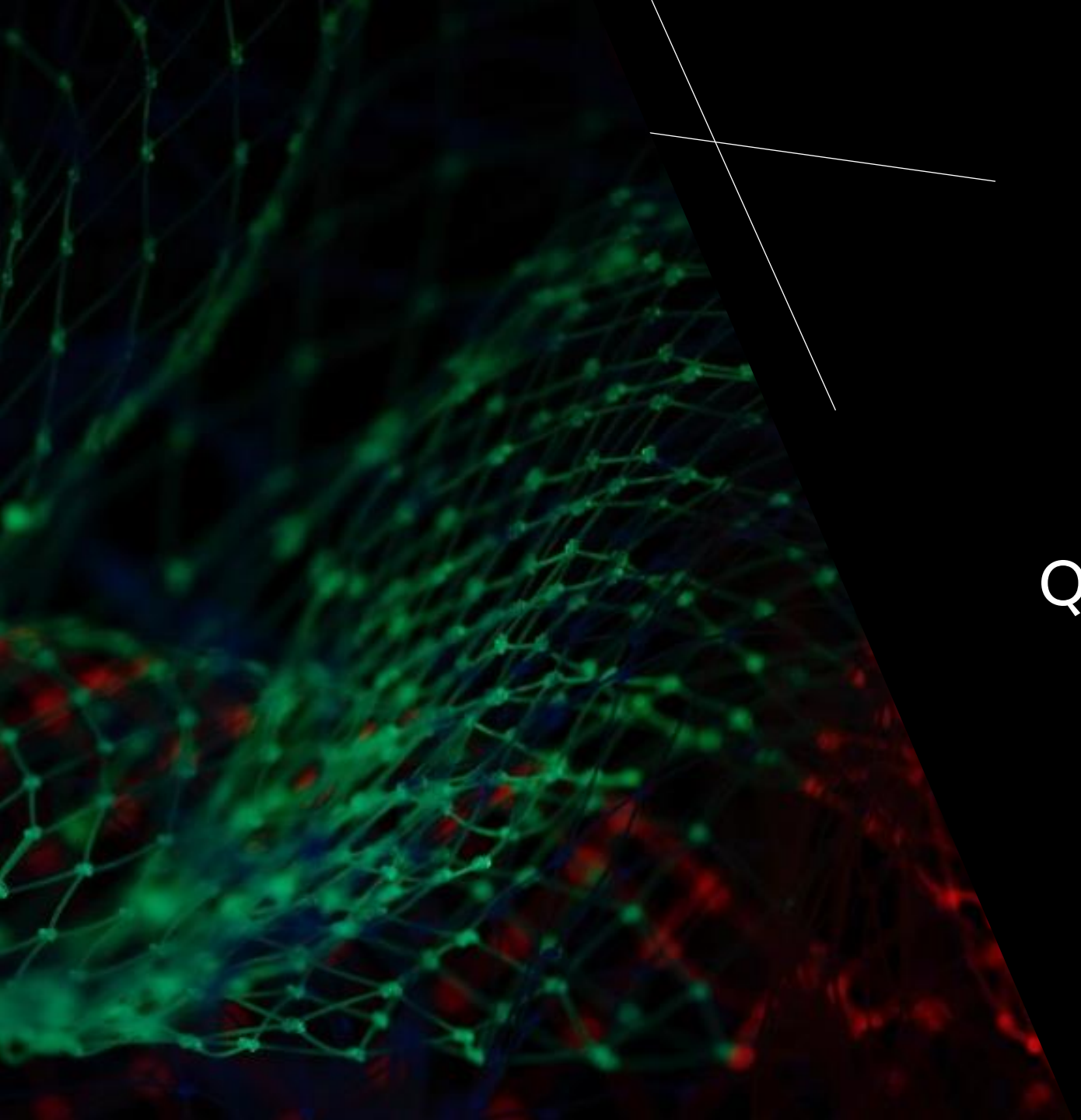
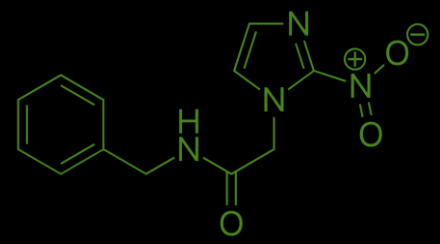




# QUIMIOINFORMÁTICA

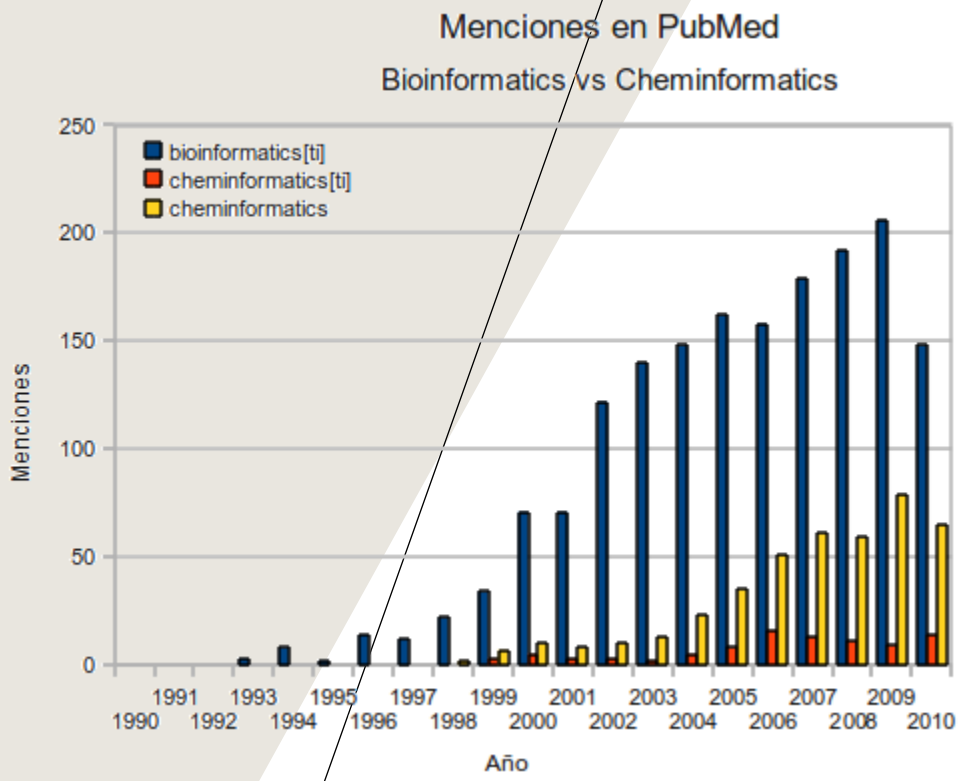
Fernán Agüero
Instituto de Investigaciones Biotecnológicas, UNSAM

# INTRODUCTION TO CHEMINFORMATICS

Cheminformatics is a relatively new field of information technology that focuses on the collection, storage, analysis, and manipulation of chemical data. The chemical data of interest typically includes information on small molecule formulas, structures, properties, spectra, and activities (biological or industrial). Cheminformatics originally emerged as a vehicle to help the drug discovery and development process, however cheminformatics now plays an increasingly important role in many areas of biology, chemistry, and biochemistry. The intent of this unit is to give readers some introduction into the field of cheminformatics and to show how cheminformatics not only shares many similarities with the field of bioinformatics, but that it can also enhance much of what is currently done in bioinformatics.

-- David Wishart

# CHEMINFORMATICS – QUÉ ES?




*"The application of computational techniques to the discovery, management, interpretation and manipulation of chemical information and data extracted therefrom".*
Chemistry plans a structural overhaul.
Nature 419:4-7 (2002)

Se la conoce como:

- Computational chemistry
- Theoretical chemistry
- Molecular modeling

Nace con el desarrollo de la mecánica cuántica a principios del siglo XX

Parece haber pasado desapercibida en la revolución "omica"

En activo desarrollo y expansión a partir de la introducción de las computadoras

# CHEMINFORMATICS EN LA LITERATURA

| Term | *Google* | *Google Scholar* | *Web of Knowledge* | *Scopus* |
|---|---|---|---|---|
| Chemical documentation | 695,000 | 66 | 1 | 34 |
| Chemical informatics | 50,400 | 129 | 20 | 39 |
| Chemical information management | 978 | 42 | 4 | 28 |
| Chemical information science | 779 | 17 | 2 | 5 |
| Chemiinformatics | 2,230 | 2 | 2 | 2 |
| Cheminformatics | 320,000 | 447 | 83 | 250 |
| Chemoinformatics | 191,000 | 5636 | 99 | 473 |

Table 1. Occurrences of search terms in *Google*, *Google Scholar*, the *Web of Knowledge* and *Scopus*

## Google:

- "Bioinformatics" (2023): ~ 240 millones de páginas
- "Cheminformatics" (2023): ~ 1.6 millones de páginas

Willett P (2007). *A bibliometric analysis of the literature of chemoinformatics.* **Aslib Proceedings, 60:** 4-17

# CUESTIONES QUÍMICAS

## La química se ocupa de esto

**estructura** → **propiedades**

### Compuestos

- Propiedades Físicas (➔Energía)
- Propiedades Químicas (Estructura, Reactividad)
- Propiedades Biológicas (➔Actividad)
- Separaciones de mezclas de compuestos
- **Aspectos estáticos**

### Transformaciones

- Reacciones químicas
- **Aspectos dinámicos**

## Y tiene estos desafíos

**propiedades** → **estructura**

### Inferencia

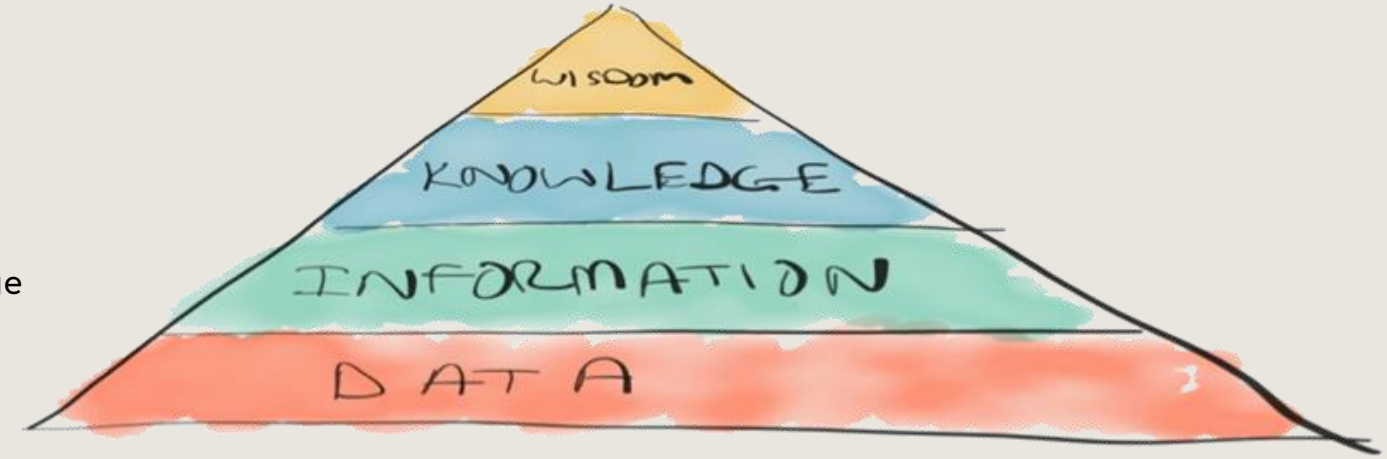
- Qué compuestos (estructuras) van a mostrar una determinada propiedad?
  - Inhibición de una actividad enzimática X (ej. drogas)
  - Propiedades mecánicas y elásticas definidas (ej. polímeros)
- Definir caminos óptimos para la síntesis de compuestos
  - Reacciones
  - Materiales iniciales
- Predecir estructuras
  - A partir de datos experimentales (ej NMR)
  - Compuestos desconocidos

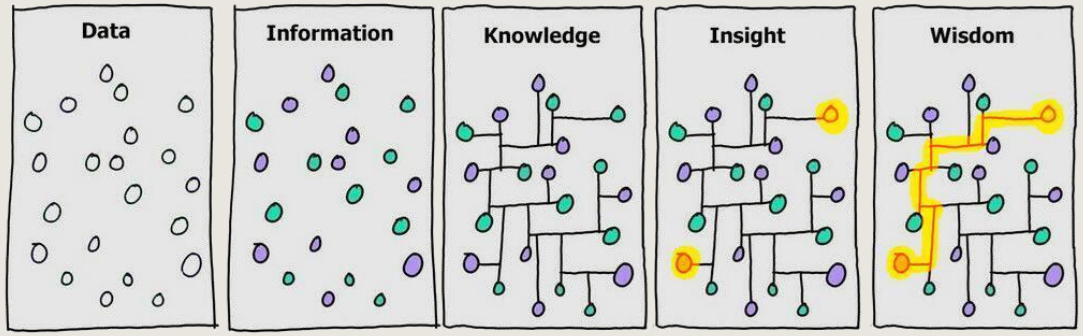
Síntesis, Abstracciones, Predicciones — Insight, Wisdom

Utilización de información para aplicaciones (memorización de datos) — Knowledge

Datos ordenados, refinados y puestos en contexto — Information

Experimentos, Mediciones — Data




**Predecir**

# EL DESAFÍO DE LA QUIMIOINFORMÁTICA

## Transformar datos en conocimiento

- El curso de una reacción química en un solvente determinado, a una temperatura dada y usando un catalizador definido
- La actividad biológica de un compuesto X contra una proteína target Y

# TEORÍA VS MODELOS

## En esencia son lo mismo, pero

Una **teoría** suele ser general

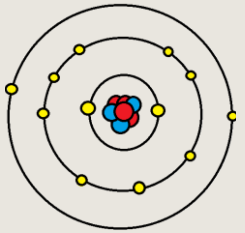
Mientras que los **modelos** introducen particularidades para facilitar la interpretación y el entendimiento

En algunos casos los **modelos** son aproximaciones, con error medible.

## Mecánica cuántica

**Teoría** fundamental de la química

Permite describir un sistema (por ej una molécula) en forma completa, usando funciones de onda, formación y ruptura de enlaces, reacciones químicas, etc.



## Modelo de valencia, capas de electrones y repulsión

Todos estudiamos este modelo en cursos básicos de química

Es un modelo o aproximación

Permite entender los mismos sistemas fácilmente

Pero tiene problemas para describir comportamientos de algunos sistemas químicos

# REPRESENTACIÓN DE COMPUESTOS QUÍMICOS

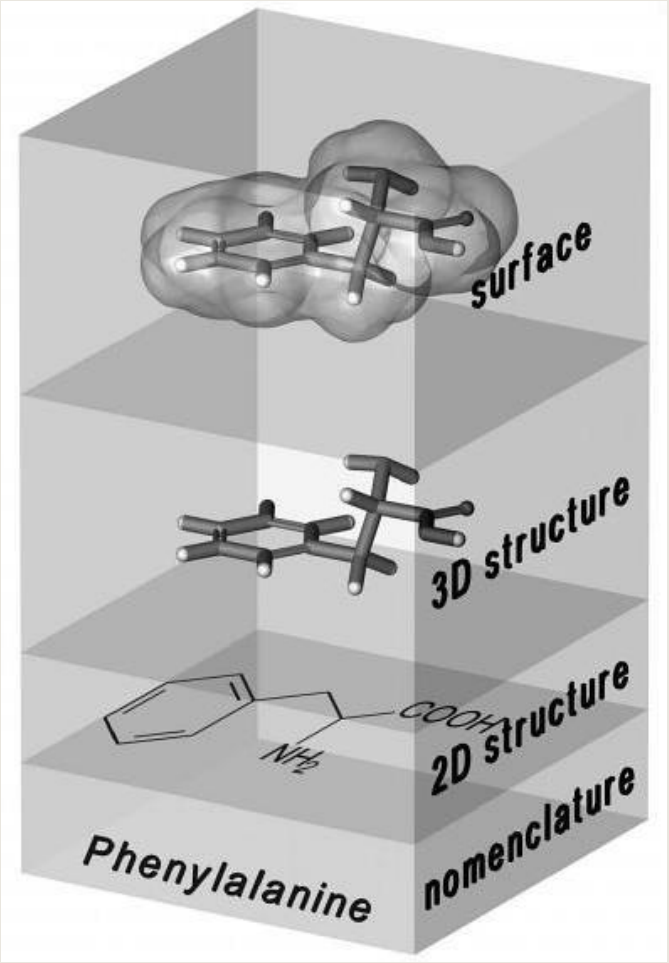
## 2D Structure vs 3D Structure

### 2D: Lenguaje natural “universal” entre químicos

- Explica la topología de una molécula
- Qué átomos están conectados mediante qué enlaces
- No explica el arreglo tridimensional de los átomos

### 3D: Requiere datos adicionales

- Posición de los átomos en el espacio
- Ángulos y distancias de los enlaces

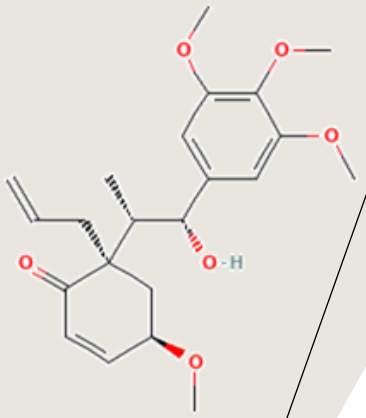


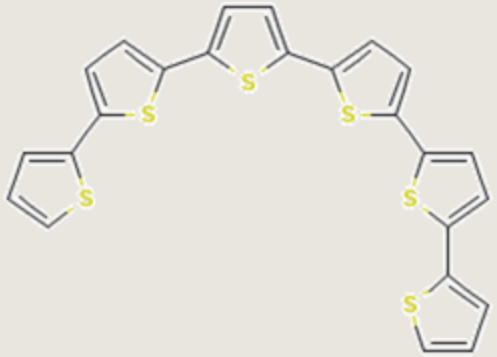

Hierarchical scheme for representations of a molecule with different content of structural information.
Tomado de J Gasteiger & T Engel (2003).

# NOMENCLATURA QUÍMICA

**Moléculas con nombres populares raros:**

Traumatic acid
Erotic acid
Commic acid
Diabolic acid
Megaphone
Sexitiophene





**Histórica**

aqua fortis (nitric acid)
oil of vitriol (sulfuric acid)
sweet oil of vitriol (diethyl ether)

**Trivial**

Fenilalanina
Ibuprofeno

Popular, pero díficil de sistematizar

**IUPAC**

2-amino-3-phenylpropanoic acid
2-[4-(2-methylpropyl)phenyl]propanoic acid

Sistemático, pero los nombres pueden ser largos!

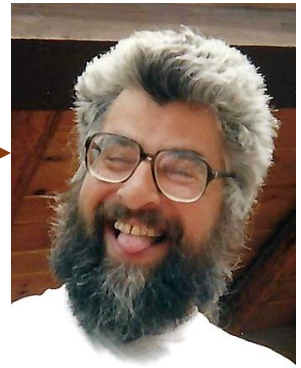
**Fórmula empírica**

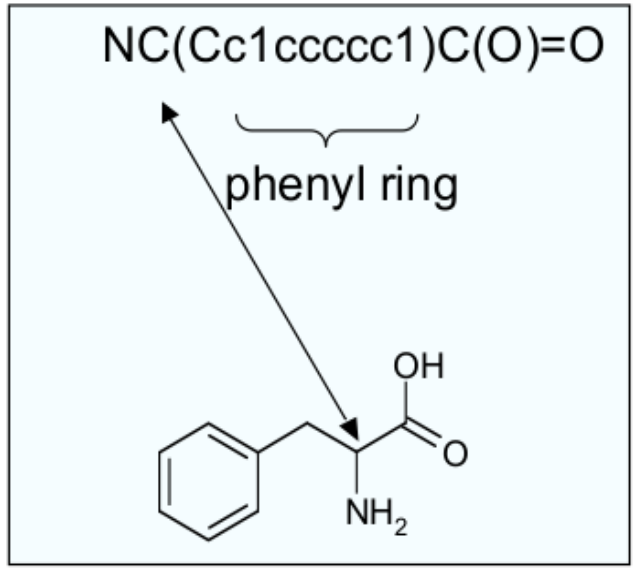
$C_9H_{11}NO_2$
$C_{13}H_{18}O_2$

Ambiguo: varios compuestos pueden tener la misma fórmula

# REPRESENTACIÓN DE COMPUESTOS QUÍMICOS: SMILES

**SMILES (*S*implified *M*olecular *I*nput *L*ine *E*ntry *S*ystem)**
Introducido en 1986 por David Weininger
Representa moléculas en forma lexicográfica
**Usa conceptos de *grafos* |** *Nodos* conectados a través de *aristas* o *arcos*






**Reglas:**
Los átomos se representan con sus respectivos símbolos:
C, N, Br, Na, Cl, O, F
MAYUSCULAS ➔ alifáticos; minúsculas ➔ aromáticos
Los hidrógenos son implícitos
Los átomos vecinos aparecen juntos
Se usan paréntesis cuando hay más de un vecino: ramificaciones
Enlaces dobles se representan usando '='
Enlaces triples se representan usando '#'

Quiralidad: '@' (contrario a las agujas del reloj)
'@@' (en el sentido de las agujas del reloj)

Anillos: números a continuación de los átomos que abren/cierran el ciclo

**Más información y reglas en:**
https://www.daylight.com/dayhtml/doc/theory/theory.smiles.html
https://en.wikipedia.org/wiki/Simplified_molecular-input_line-entry_system

**Otros ejemplos:**
Ciclohexano: C1CCCCC1
Benceno: C1=CC=CC=C1 (Kekulé)
Benceno: c1ccccc1
Etanol: CCO
Piridina: C1=CC=NC=C1 (Kekulé)
Piridina: c1ccncc1
Acido acético: CC(=O)O
Acido cianhídrico: C#N
L-alanina: N[C@@H](C)C(=O)O
L-alanina (sin especificar quiralidad): N[CH](C)C(=O)O
Cloruro de Sodio: [Na+].[Cl-]

# ANILLOS EN SMILES

## Linealizar y Etiquetar

**Linealizar el anillo en cualquier parte**

Benceno: cccccc (C=CC=CC=C)

Dioxano: occocc, ccocco, coccoc

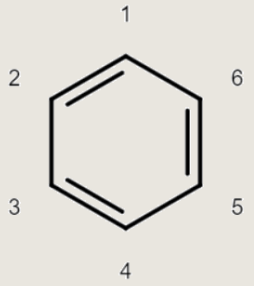
**Agregar etiquetas numéricas para indicar el inicio y cierre del anillo**

Benceno: c1ccccc1

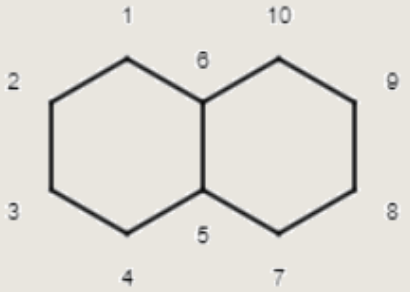
Dioxano: O1CCOCC1, C1COCCO1, C1OCCOC1

*Las etiquetas numericas pueden empezar en cero (0)* ***pero*** *rara vez se usa*

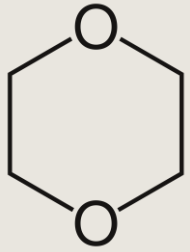
Decalin: C1CCCC2C1CCCC2, C1CCCC2CCCCC12




benceno




decalin

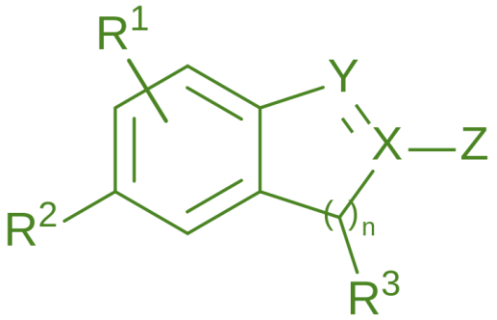



dioxano

# REPRESENTACIÓN DE PATRONES EN MOLECULAS

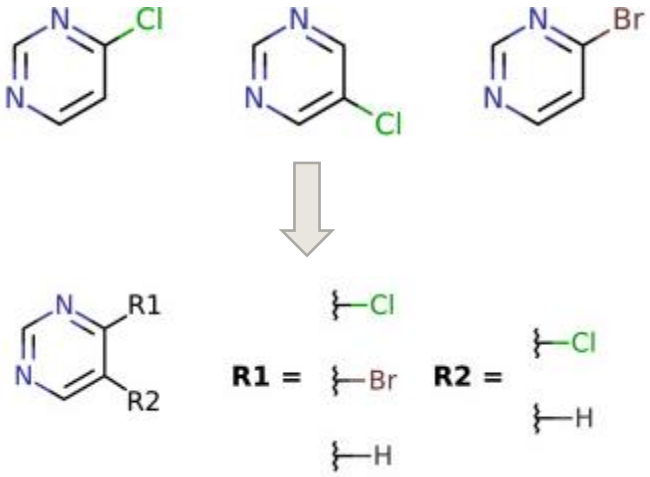
Al principio hubo **Markush structures**:

- Representan varias estructuras posibles
- Grupos R variables
- Descripción general de una molécula con **ambigüedad** en algunas posiciones




https://es.wikipedia.org/wiki/Estructura_de_Markush

Son comunes en patentes, y en libros de texto.




Eugene A. Markush

# REPRESENTACIÓN DE PATRONES: SMARTS

***SMARTS - A Language for Describing Molecular Patterns***

Representación lexicográfica de partes de una molécula

Es una extensión de **SMILES**

Concepto similar al de **expresiones regulares** (regex) en texto.

https://en.wikipedia.org/wiki/Regular_expression

**Reglas (las mismas que SMILES), y además:**

Representción de patrones para átomos:

* cualquier átomo

a aromático

A alifático  ... hay más reglas para átomos

Representación de patrones para enlaces:

~ cualquier enlace

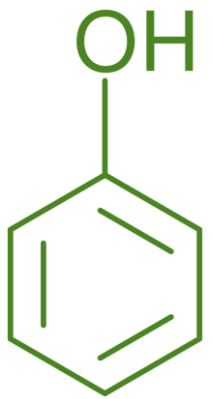
@ cualquier enlace en un anillo

/ enlace dirigido “arriba”

\ enlace dirigido “abajo”

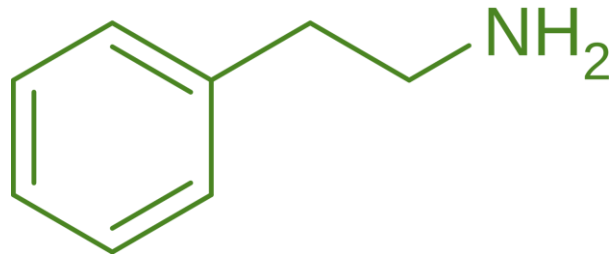
**Más información y reglas en:**

https://www.daylight.com/dayhtml/doc/theory/theory.smarts.html




**SMARTS: [OH]c1ccccc1**

hydroxyl-group attached to 6 aromatic carbons in a ring

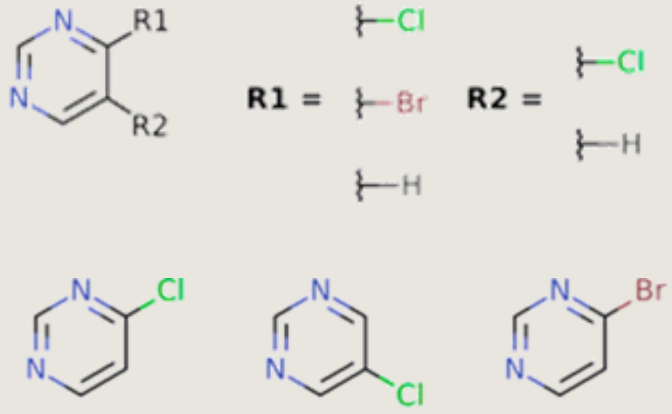



**SMARTS: NCCc1ccccc1**

Aliphatic nitrogen attached to 2 aliphatic carbons attached to 6 aromatic carbons in a ring

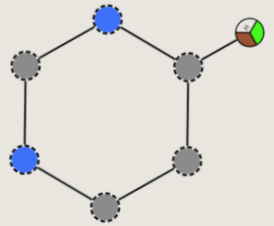
# FROM MARKUSH TO SMARTS

## Original molecules

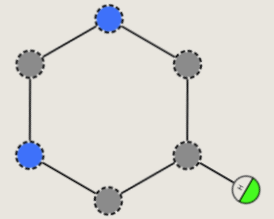



## SMARTS PATTERNS

n1cnc([Cl,Br,H])cc1



n1cncc([Cl,H])c1



SMARTS.PLUS SmartView: https://smarts.plus/

# SMARTS – A LANGUAGE FOR DESCRIBING MOLECULAR PATTERNS

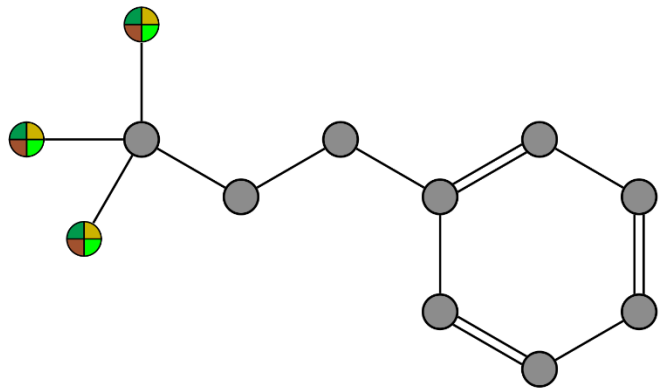
Una representación SMILES es un patrón SMARTS válido
[OH]c1ccccc1 (phenol)

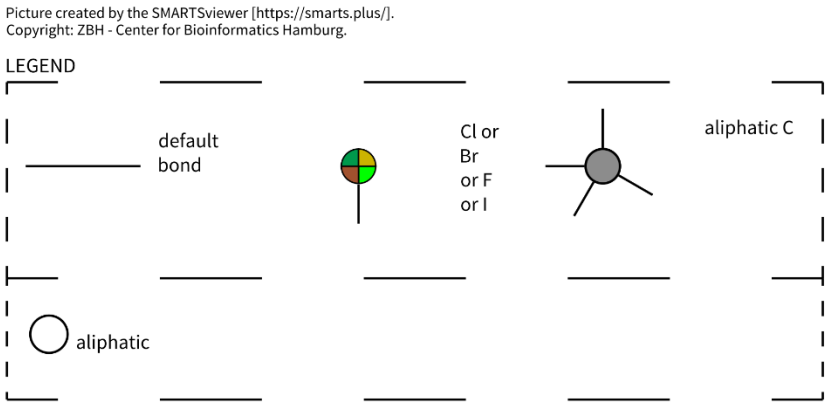
Patrones SMARTS simples
[C,N]1CCCCC1

[Cl,Br,F,I]C([Cl,Br,F,I])([Cl,Br,F,I])CCC1=CC=CC=C1

C-C=C-C=C~*~[++]

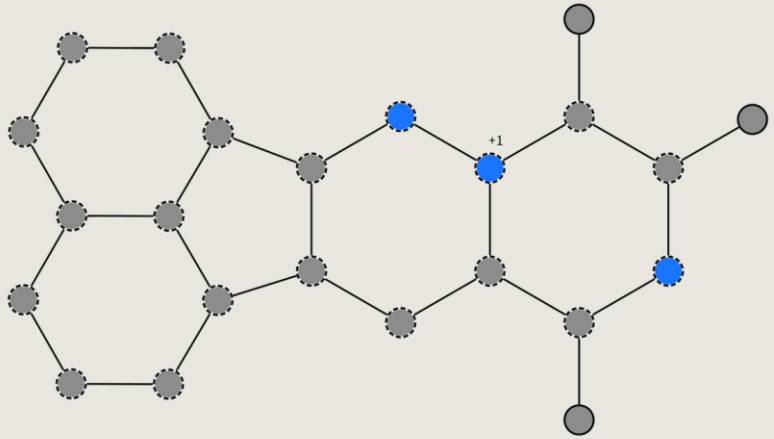
[Cl,Br,F,I]C([Cl,Br,F,I])([Cl,Br,F,I])CCC1=CC=CC=C1

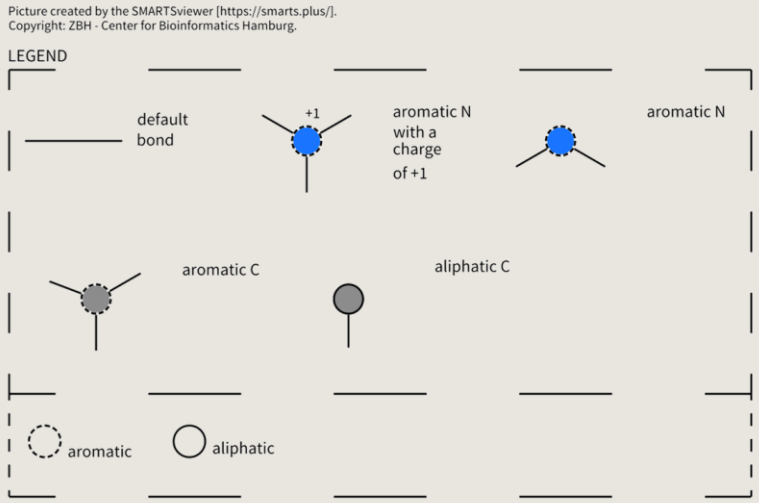





Referencias
https://www.daylight.com/dayhtml/doc/theory/theory.smarts.html
https://smarts.plus/

# PATRONES SMARTS PARA BÚSQUEDAS




Referencias

https://www.daylight.com/dayhtml/doc/theory/theory.smarts.html

https://smarts.plus/

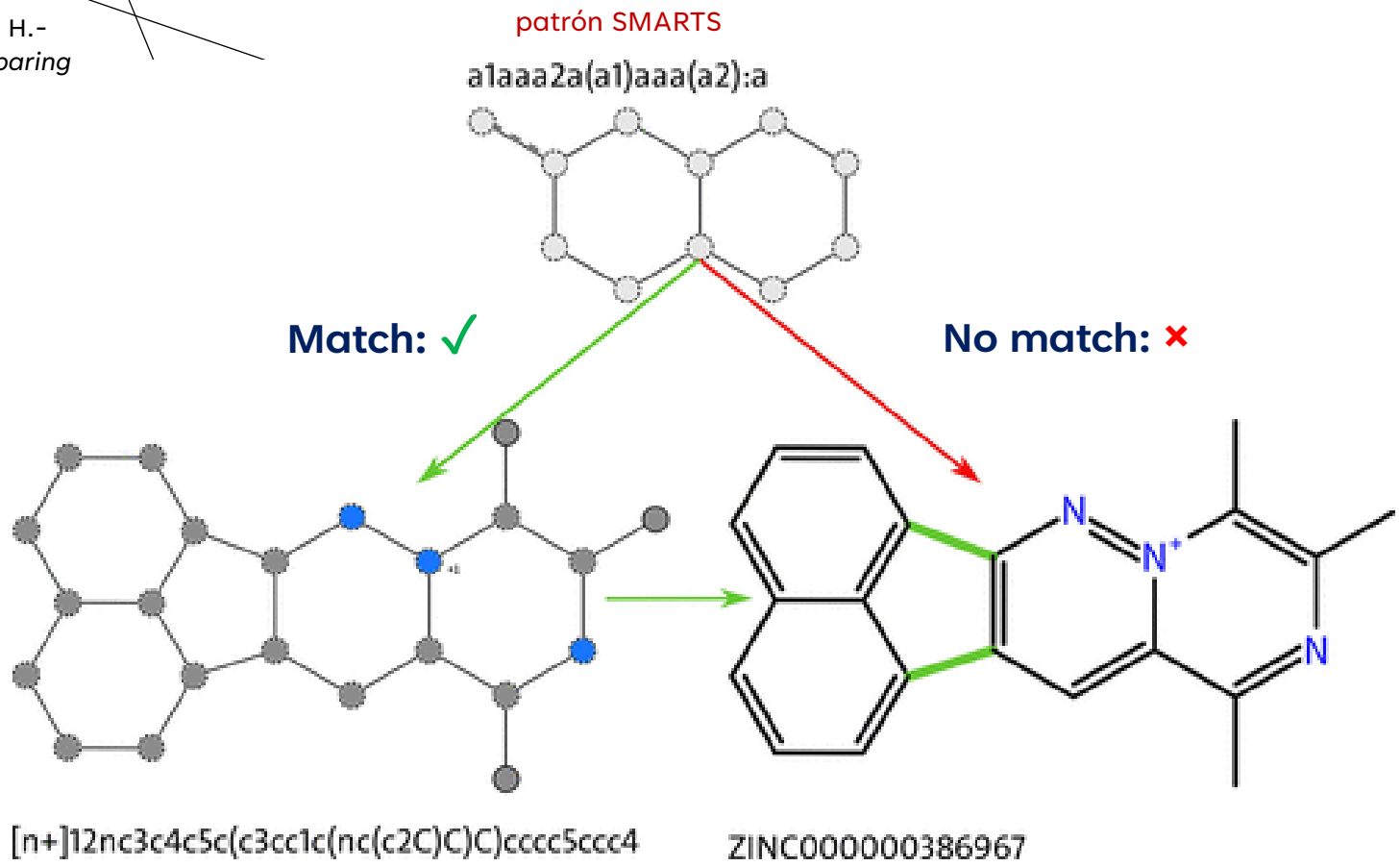
[n+]12nc3c4c5c(c3cc1c(nc(c2C)C)C)cccc5ccc4

C = carbono alifático

c = carbono aromático

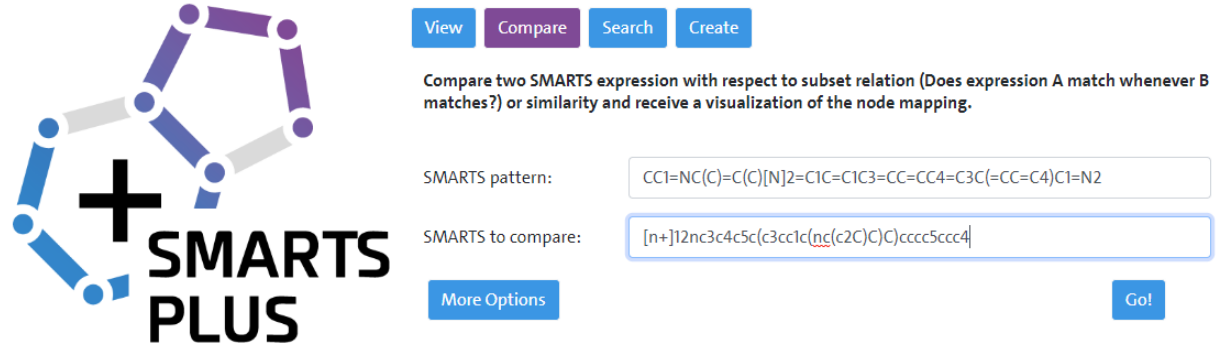
# MATCHING SMARTS PATTERNS

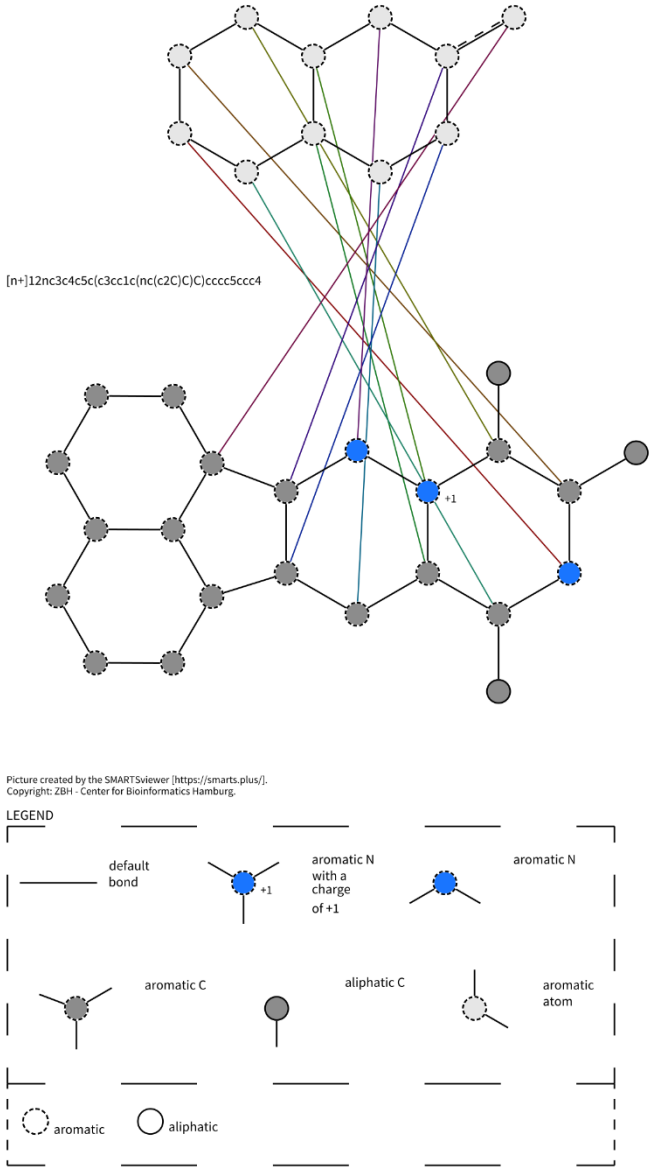
Schmidt, R., Ehmki, E. S. R., Ohm, F., Ehrlich, H.-C., Mashychev, A., & Rarey, M. (2019). *Comparing Molecular Patterns Using the Example of SMARTS: Theory and Algorithms. Journal of Chemical Information and Modeling.* doi:10.1021/acs.jcim.9b00250

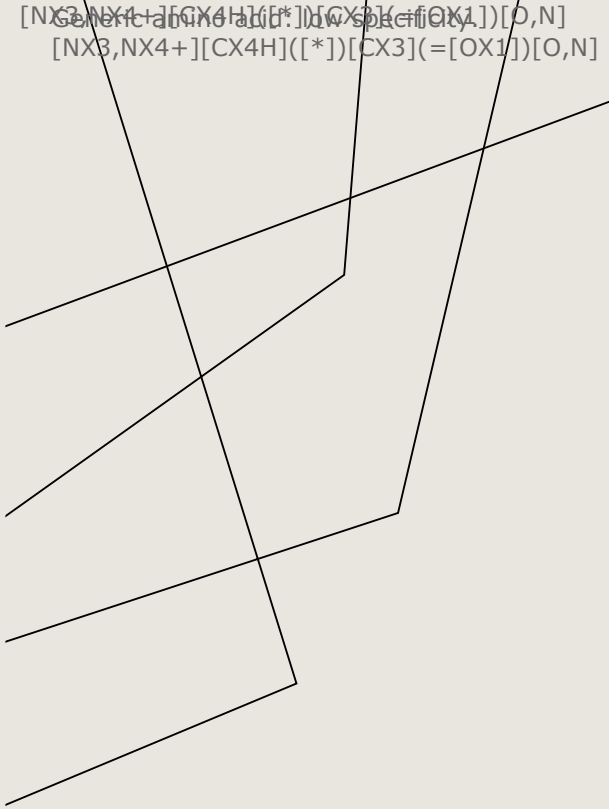


# TESTING AT SMARTS.PLUS

https://smarts.plus/






# SMARTS EXAMPLES

## Amino Acids

Generic amino acid: low specificity:

[NX3,NX4+][CX4H]([*])[CX3](=[OX1])[O,N]

## Other interesting examples

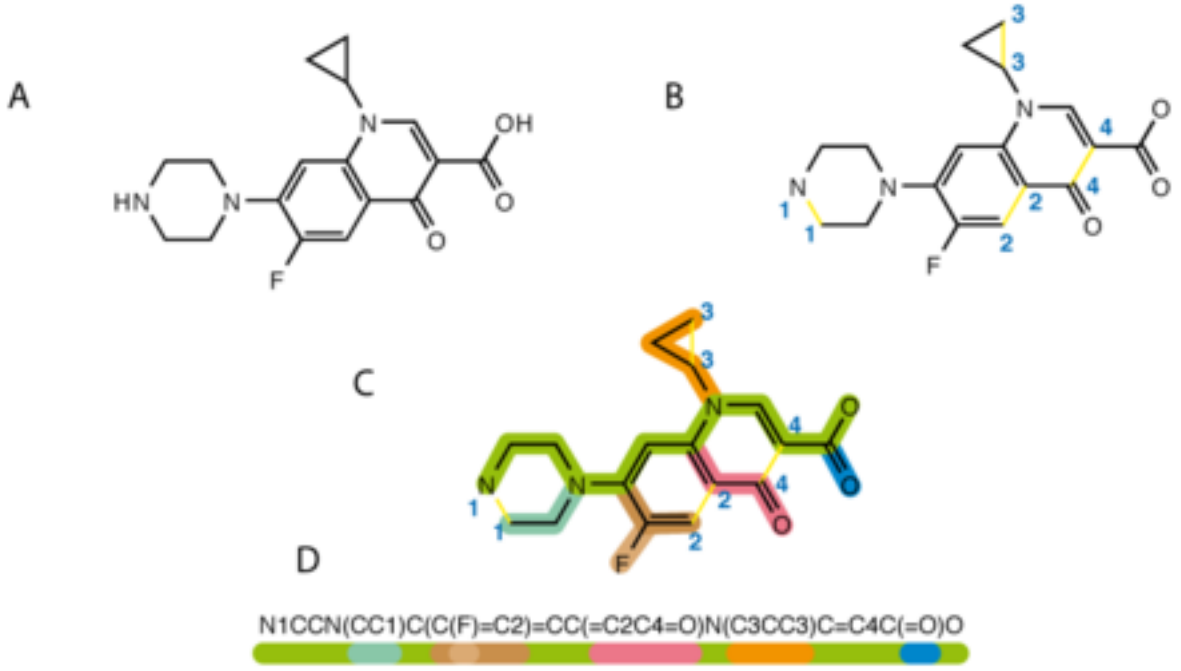
https://www.daylight.com/dayhtml_tutorials/languages/smarts/smarts_examples.html

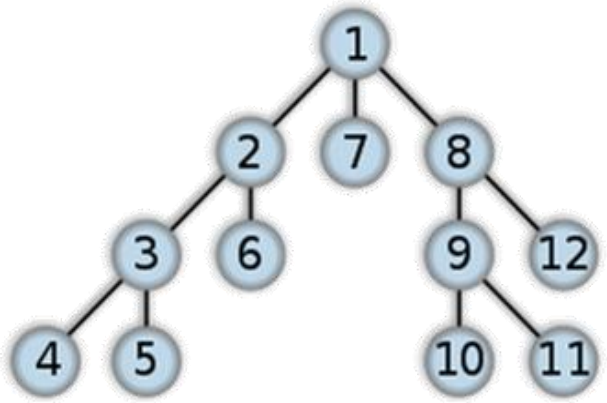
# REPRESENTACIÓN DE COMPUESTOS QUÍMICOS: SMILES

SMILES, relación con Teoría de Grafos

SMILES es una cadena de texto (ASCII)

Es el producto de escribir los símbolos (átomos) a medida que se recorre el grafo químico (la molécula) de modo *depth-first*




Order in which the nodes are expanded

| | |
|---|---|
| **Class** | Search algorithm |
| **Data structure** | Graph |
| **Worst case performance** | $O(\|V\| + \|E\|)$ for explicit graphs traversed without repetition, $O(b^d)$ for implicit graphs with branching factor $b$ searched to depth $d$ |
| **Worst case space complexity** | $O(\|V\|)$ if entire graph is traversed without repetition, O(longest path length searched) for implicit graphs without elimination of duplicate nodes |

**Depth-first Tree/Graph Traversal:**
http://en.wikipedia.org/wiki/Depth-first_search

# CANONIZACIÓN DE MOLÉCULAS: ALGORITMO DE MORGAN

Canonización: Representar la conectividad de una molécula de manera uniforme

Una estructura con ***n*** átomos puede ser descripta de ***n!*** maneras diferentes

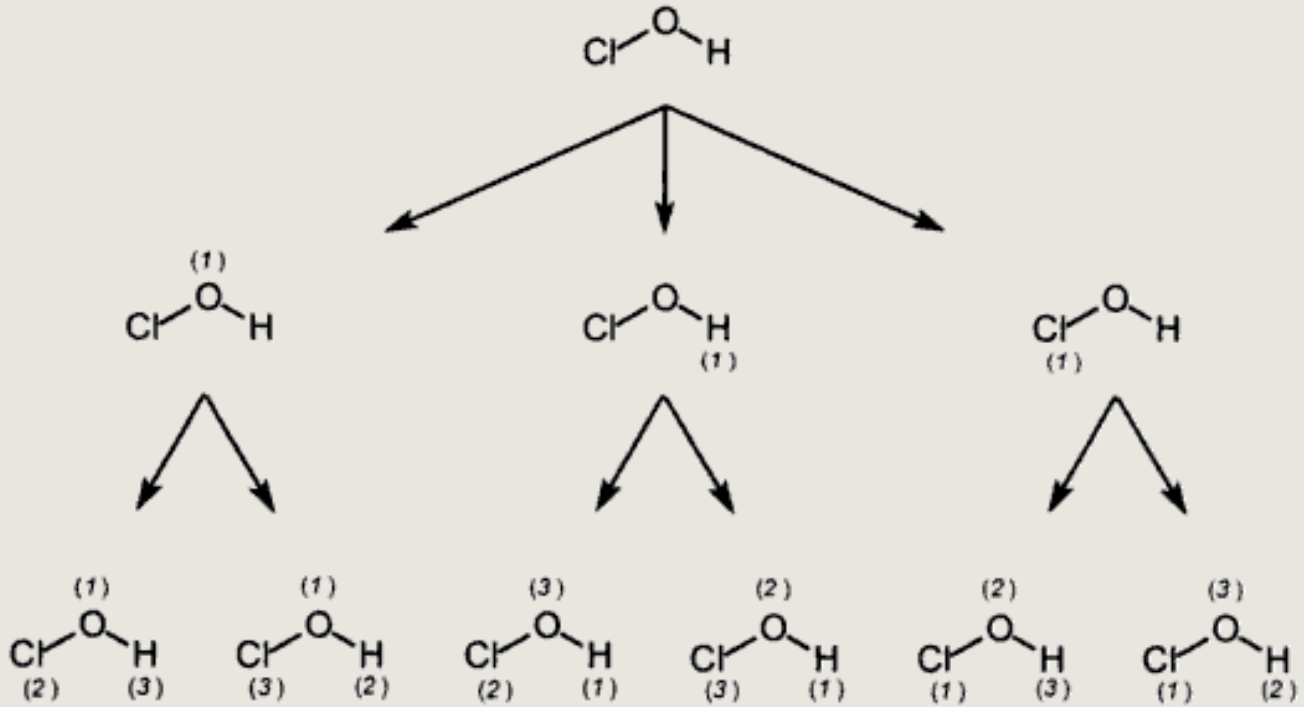


**Figure 2-41.** Six different possibilities for numbering the atoms in a hypochlorous acid molecule.

El algoritmo de Morgan es viejo pero lo vamos a usar para aprender el concepto de **canonización**!

Hay variantes nuevas!

Schneider N, Sayle RA, Landrum GA. Get Your Atoms in Order--An Open-Source Implementation of a Novel and Robust Molecular Canonicalization Algorithm. J Chem Inf Model. 2015 Oct 26;55(10):2111-20. doi: 10.1021/acs.jcim.5b00543. Epub 2015 Oct 15. PMID: 26441310.

# CANONIZACIÓN: ALGORITMO DE MORGAN

**Paso 1:** clasificar átomos de acuerdo a conectividad (vecindad)

Estructuras conteniendo C, N, O, H y halógenos se clasifican en cuatro categorías dependiendo del número de enlaces (no H)

**Paso 2:** Iteraciones

En una segunda iteración los valores de conectividad de cada átomo se incrementan de acuerdo al de los vecinos siguiendo una serie de reglas:

- Sumas (atomos internos) o transferencia de valores (atomos terminales)
- *Extended connectivity*

Las iteraciones siguen hasta que los valores de EC son iguales o menores a los de la iteración anterior

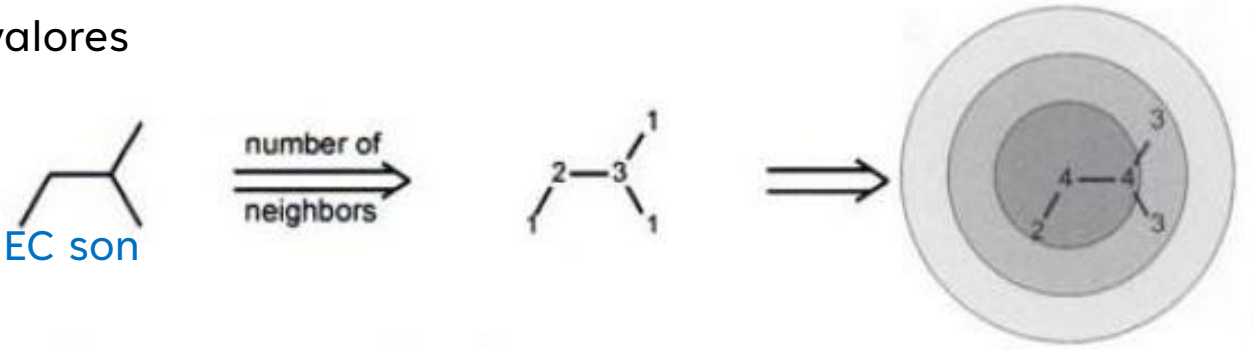



**Figure 2-43.** The EC value or the atom classification of each atom, respectively, is calculated by summing the EC values of the directly connected neighboring atoms of the former sphere (relaxation process).

# CANONIZACIÓN: ALGORITMO DE MORGAN

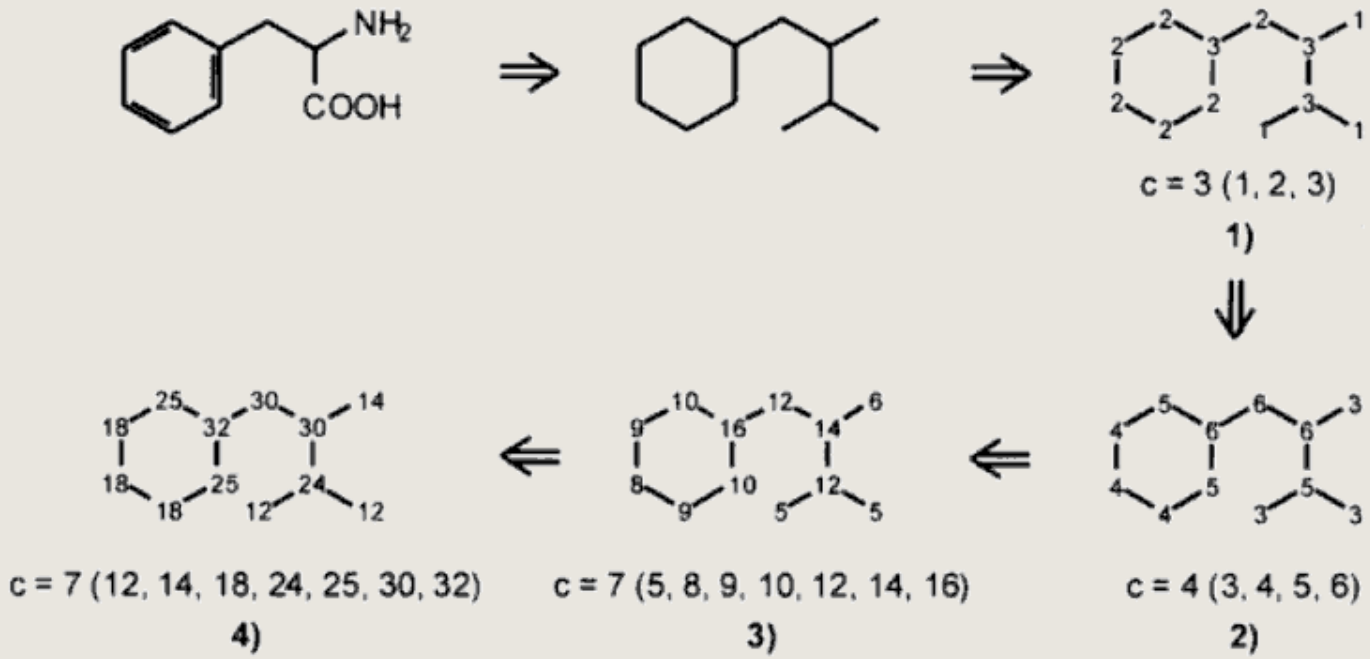



**Figure 2-44.** The EC values of the atoms of phenylalanine (without hydrogens) are calculated by considering the class values of the neighboring atoms. After each relaxation process, *c*, the number of equivalent classes (different EC values), is determined.

The process is repeated until the number of different EC values is lower than or equal to the number of different EC values in the previous iteration.

# CANONIZACIÓN: ALGORITMO DE MORGAN

Paso 3: Asignación de números de átomos únicos

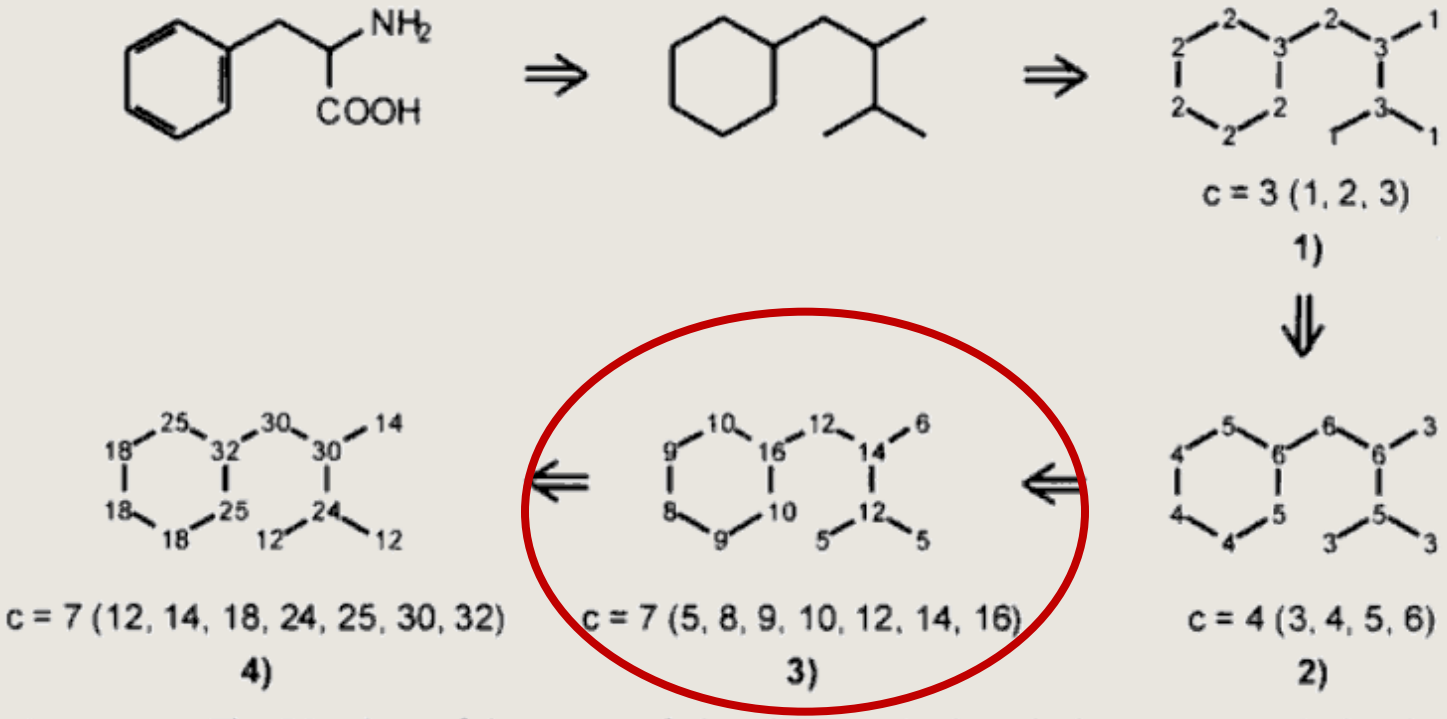



**Figure 2-44.** The EC values of the atoms of phenylalanine (without hydrogens) are calculated by considering the class values of the neighboring atoms. After each relaxation process, *c*, the number of equivalent classes (different EC values), is determined.

Se comienza por el paso en el que se obtiene el mayor EC por primera vez.

El atomo número 1 es el que tiene el mayor valor de EC en este paso.

El atomo 2 es el que sigue en la secuencia de valores EC.

# CANONIZACIÓN DE MOLECULAS

El algoritmo es de 1965!  Es viejo!

Hay moléculas problemáticas que no son fáciles de canonizar.

El problema general que intenta resolver es el de **Canonizacion de Grafos**

- Es un problema computacional complejo
- Relacionado con problemas de isomorfismo de grafos
- Hay muchas otras maneras (algoritmos) de resolverlos: http://en.wikipedia.org/wiki/Graph_canonization

En resumen:
Después de aplicar un método de canonización

El que implementa RDKit (Python)

Schneider N, Sayle RA, Landrum GA. Get Your Atoms in Order--An Open-Source Implementation of a Novel and Robust Molecular Canonicalization Algorithm. J Chem Inf Model. 2015 Oct 26;55(10):2111-20. doi: 10.1021/acs.jcim.5b00543. Epub 2015 Oct 15. PMID: 26441310.

Krotko DG. Atomic ring invariant and Modified CANON extended connectivity algorithm for symmetry perception in molecular graphs and rigorous canonicalization of SMILES. J Cheminform. 2020 Aug 20;12(1):48. doi: 10.1186/s13321-020-00453-4. PMID: 33431026; PMCID: PMC7439248.

# REPRESENTACIÓN DE COMPUESTOS QUÍMICOS: INCHI

InChI –***In***ternational ***Ch***emical ***I***dentifier

Introducido recientemente (2005) por IUPAC (International Union of Pure and Applied Chemistry)

**Objetivos**

Establecer un identificador (nomenclatura, etiqueta) ***único*** y ***no propietario*** para cada molécula

Que pueda ser utilizado tanto en medios impresos como electrónicos y que facilite la búsqueda de compuestos

Heller SR, McNaught A, Pletnev I, Stein S, Tchekhovskoi D. InChI, the IUPAC International Chemical Identifier. J Cheminform. 2015 May 30;7:23. doi: 10.1186/s13321-015-0068-4. PMID: 26136848; PMCID: PMC4486400.

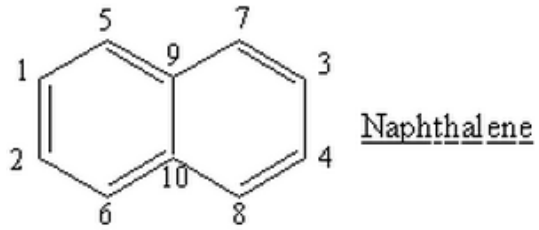
# REPRESENTACIÓN DE COMPUESTOS QUÍMICOS: INCHI

Formato de un identificador InChI

Es una cadena de texto (ASCII) compuesta por *segmentos* (layers) separada por *delimitadores* (/)

Cada capa contiene distintos tipos de información estructural

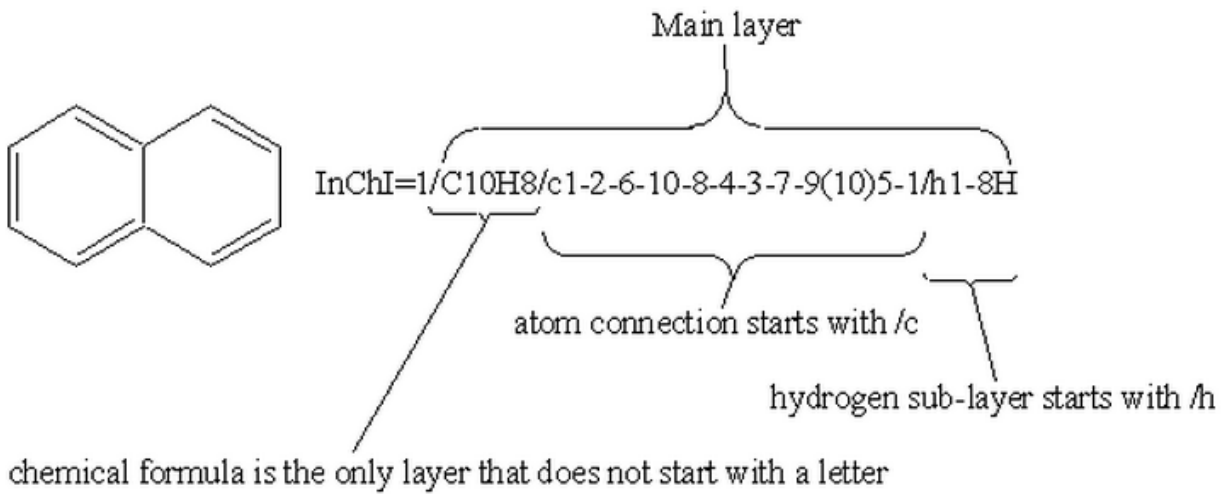
Los números dentro de una capa representan la numeración canónica de los átomos de la primera capa (fórmula) excepto los hidrógenos.




**Ejemplos:**

Agua: InChI = 1/H2O/h1H2

Benceno: InChI = 1/C6H6/c1-2-4-6-5-3-1/h1-6H



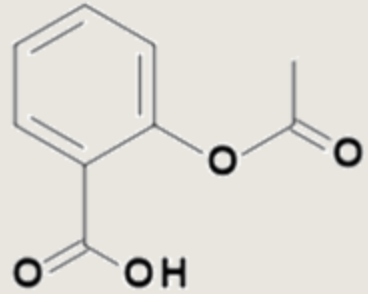
# INCHI IDENTIFIER: MAIN LAYER

Conectividad

InChI = 1S/C9H8O4/c1-6(10)13-8-5-3-2-4-7(8)9(11)12/h2-5H,1H3,(H,11,12)

Fórmula empírica

Hidrógenos




Aspirin

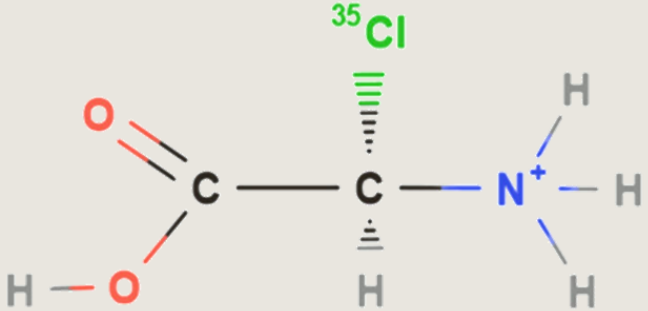
InChI = 1S/H2O/h1H2

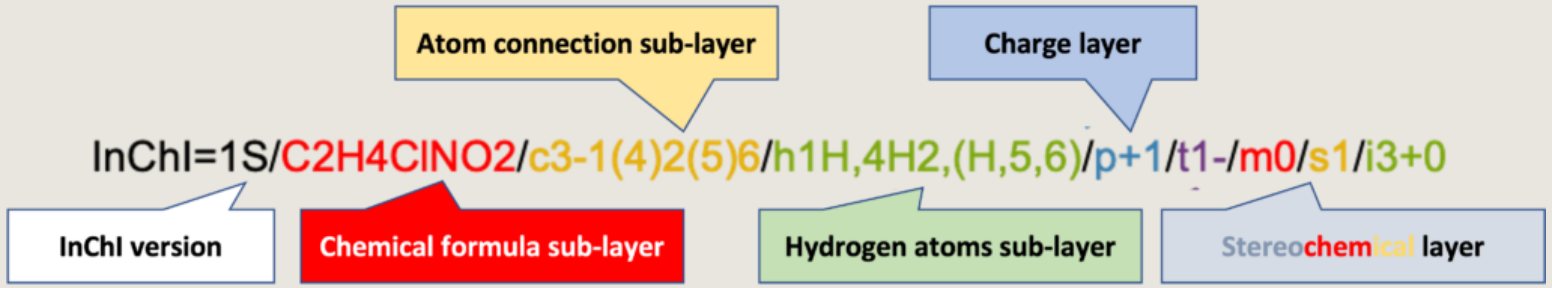
# REPRESENTACIÓN DE COMPUESTOS: INCHI

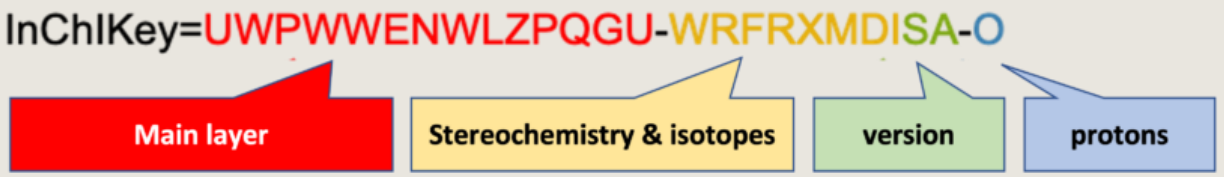
## Representa la información en capas (layers)

Permiten elegir el nivel de detalle que uno quiere incluir

Sólo la capa principal es mandatoria



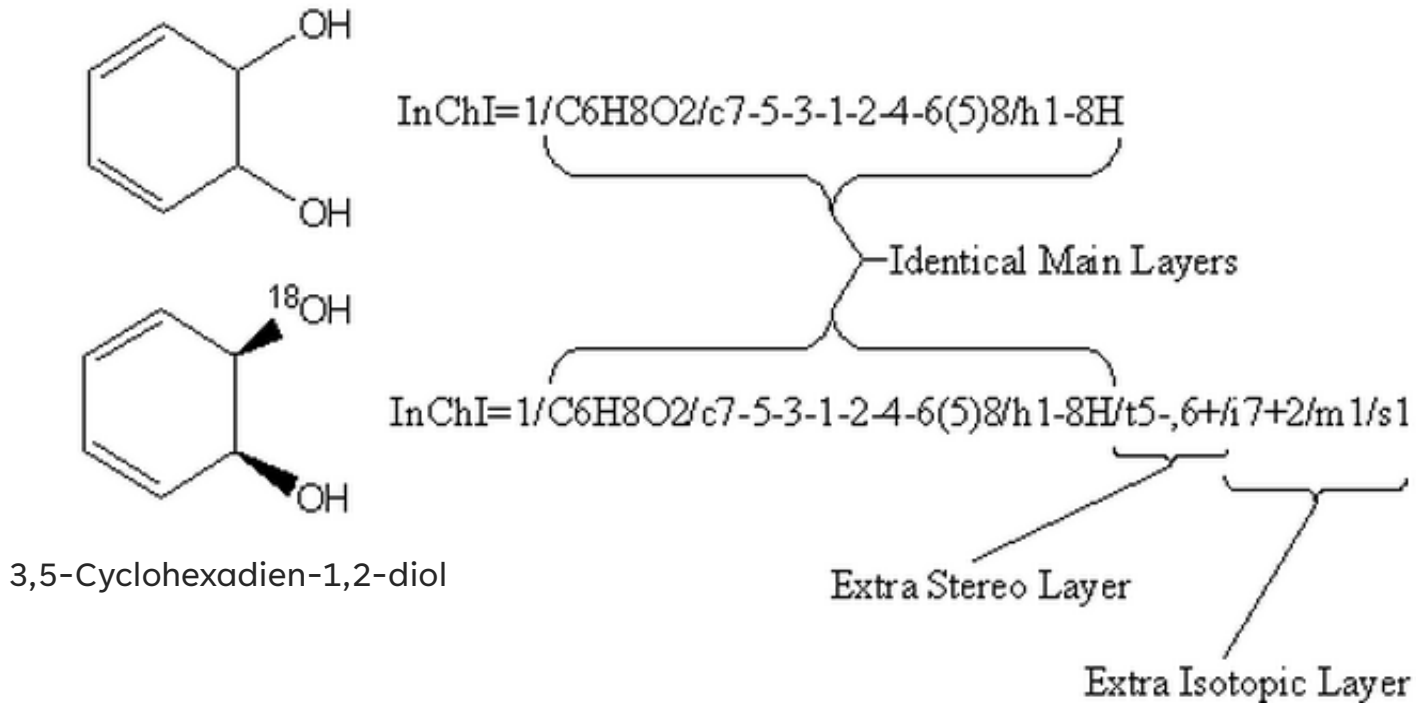




Tomado de: InChI Trust, https://www.inchi-trust.org/

# REPRESENTACIÓN DE COMPUESTOS QUÍMICOS: INCHI

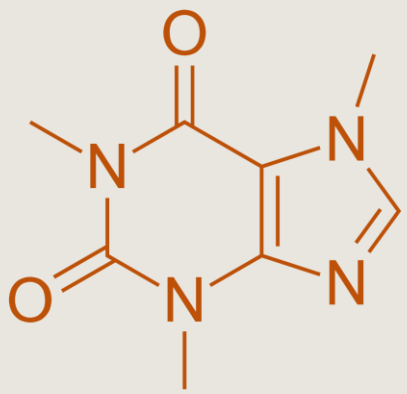
Si dos InChIs son iguales, los compuestos también lo son.

**Pero** los compuestos pueden estar representados con diferente nivel de detalle




3,5-Cyclohexadien-1,2-diol

# INCHI VS SMILES



Caffeine

Tomado de: InChI Technical FAQ
https://www.inchi-trust.org/technical-faq-2

**Valid SMILES** for Caffeine (not complete)

[c]1([n+]([CH3])[c]([c]2([c]([n+]1[CH3])[n][cH][n+]2[CH3]))[O-])[O-]
CN1C(=O)N(C)C(=O)C(N(C)C=N2)=C12
Cn1cnc2n(C)c(=O)n(C)c(=O)c12
Cn1cnc2c1c(=O)n(C)c(=O)n2C
O=C1C2=C(N=CN2C)N(C(=O)N1C)C
CN1C=NC2=C1C(=O)N(C)C(=O)N2C

**InChI:** 1S/C8H10N4O2/c1-10-4-9-6-5(10)7(13)12(3)8(14)11(6)2/h4H,1-3H3

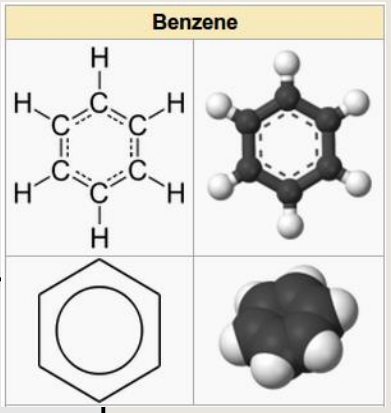
**InChI Key:** RYYVLZVUVIJVGH-UHFFFAOYSA-N

# REPRESENTACIÓN DE COMPUESTOS: MOLFILES

MDL, Molfile | Formato creado por MDL (ahora Symyx)

Contiene información sobre: Átomos, enlaces, conectividad y *coordenadas espaciales*

Permite representar moléculas tanto en 2D como en 3D




benzene.mol

**Descripción del formato**

**1. Header**

**2. Comment**

**3. General information (counts)**
6 atoms, 6 bonds, ..., V2000 standard

**4. Spacial coordinates**
X, Y, Z, element, extra information

**5. Bonding information**
1st atom, 2nd atom, bond type, extra information

**8. Final del registro.**

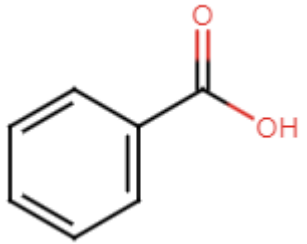
```
benzene
ACD/Labs0812062058

  6  6  0  0  0  0  0  0  0  0  1 V2000
    1.9050   -0.7932    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
    1.9050   -2.1232    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
    0.7531   -0.1282    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
    0.7531   -2.7882    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
   -0.3987   -0.7932    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
   -0.3987   -2.1232    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
  2  1  1  0  0  0  0
  3  1  2  0  0  0  0
  4  2  2  0  0  0  0
  5  3  1  0  0  0  0
  6  4  1  0  0  0  0
  6  5  2  0  0  0  0
M  END
$$$$
```

**6. Atom type**

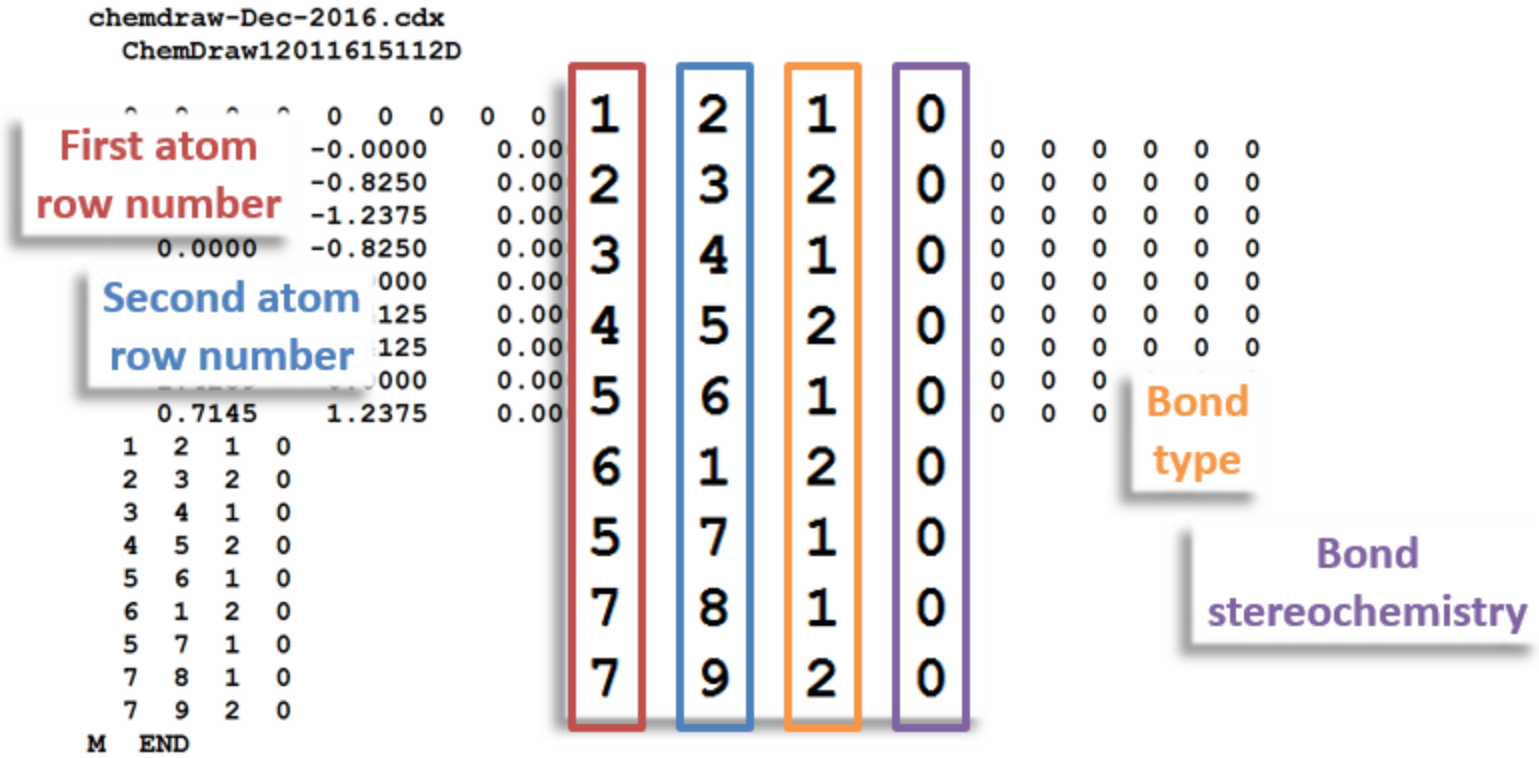
**7. Non-standard values**
Isotopes, valence, charge

# MOLFILES: BOND BLOCK

OC(=O)C1=CC=CC=C1




Anatomy of a MOL file
ChemInformatics 2017 (LibreTexts Chemistry)



# REPRESENTACIÓN DE COMPUESTOS: SDF FILES

NGC00015959-03.sdf

MOLFILE

Anotaciones

```
NCGC00015959-03
  Marvin  07111412562D

 25 30  0  0  0  0            999 V2000
    3.4098   -1.3130    0.0000 N   0  3  0  0  0  0  0  0  0  0  0  0
    4.8329   -1.3130    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
    3.4098   -2.1380    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
    4.1248   -2.5436    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
    2.6948   -2.5436    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
    4.8329   -2.1380    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
    4.1248   -0.8937    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0
    5.5547   -0.8937    0.0000 C   0  0  0  0  0  0  0  0  0  0  0  0

  1  3  1  0  0  0  0
  1  7  2  0  0  0  0
  1 25  1  0  0  0  0
  2  7  1  0  0  0  0
  2  6  2  0  0  0  0
  2  8  1  0  0  0  0
  3  4  2  0  0  0  0
  3  5  1  0  0  0  0
  4 13  1  0  0  0  0
  4  6  1  0  0  0  0
  5  9  1  0  0  0  0

M  CHG  1   1   1
M  END
>  <Formula>
C20H14NO4

>  <FW>
332.3289

>  <DSSTox_CID>
25204

>  <Active>
1
```

# CHEMICAL DATABASES

**PubChem, NCBI |** https://pubchem.ncbi.nlm.nih.gov/

repositorio abierto de information sobre moléculas y sus actividades biológicas

**ChEMBL, EBI |** https://www.ebi.ac.uk/chembl/

Repositorio abierto de bioactividades de moléculas, extraídas de la literatura

**ChemSpider, Royal Society of Chemistry |** http://www.chemspider.com/

**NIST Chemistry Web Book |** https://webbook.nist.gov/

**DrugBank |** http://www.drugbank.ca/

**Zinc Databases |** https://zinc.docking.org/

commercially-available compounds for virtual screening

# REPRESENTACIÓN DE COMPUESTOS QUÍMICOS: GRAFOS

Un grafo es una estructura *abstracta* que contiene *nodos* conectados con *aristas* (o *arcos*)

*"Los grafos son redes (networks) de puntos y líneas"*

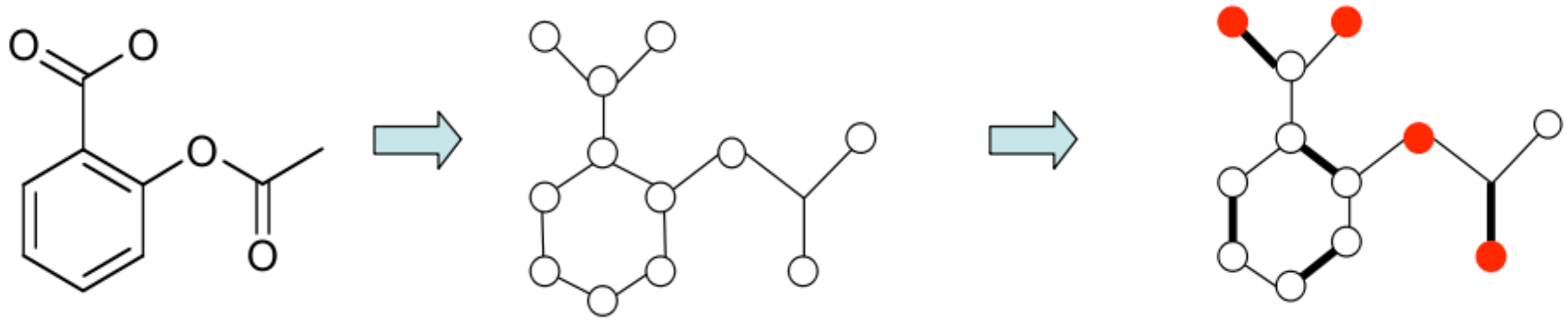
En inglés: nodes, edges

Moléculas químicas pueden representarse como *grafos:*

- Los átomos como nodos
- Los enlaces como aristas

Se pueden asociar propiedades a cada nodo (ej número atómico), y a cada arista (ej número y/o tipo de enlace)

En el grafo final pueden entonces distinguirse distintos tipos de nodos y aristas



# UN DESVÍO: HISTORIA DE LOS GRAFOS

El problema de **los 7 puentes de Königsberg**.

La ciudad de Königsberg se encuentra dividida por el río Pregel

Incluye 2 islas que se conectan con tierra mediante 7 puentes

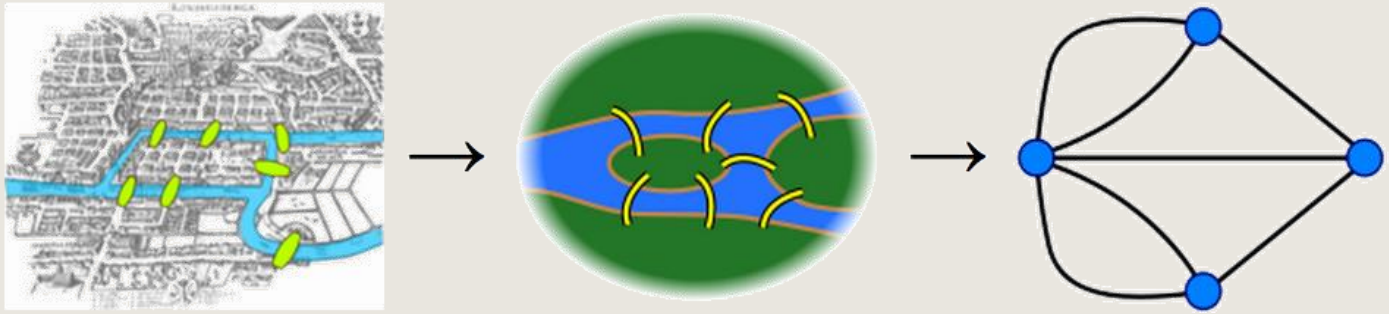
**El problema:** Encontrar un camino a través de la ciudad que cruce cada puente una sola vez. Hay que cruzar todos los puentes. Sólo se puede acceder a las islas cruzando un puente.

En 1735 Leonard Euler demostró que el problema no tiene solución.

**El razonamiento:**

La elección del camino dentro de cada porción de tierra era **irrelevante**

La única característica de la ruta elegida importante era la secuencia de puentes cruzados



**Leonard Euler (1707-1783)**

**Abstracción del problema:**
En una lista de porciones de tierra (**nodos**)

Y una lista de puentes (**aristas**)

Sólo la información de **conectividad** era relevante!

Tomado de Wikipedia
https://en.wikipedia.org/wiki/Seven_Bridges_of_K%C3%B6nigsberg

# GRAFOS: PROPIEDADES Y OPERACIONES

**Propiedades de los grafos:**

Grado de conectividad de los nodos (degree)

Direccionalidad de las aristas

Intensidad (sentido vectorial) de cada arista

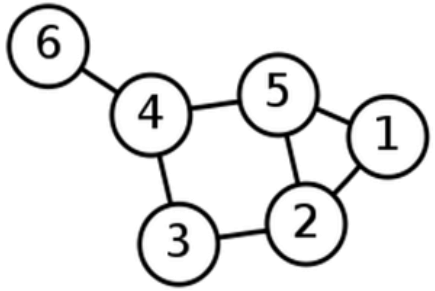
*Las aristas pueden tener asociado un valor numérico (peso, largo, costo)*

**Posibilidad de identificar los nodos**

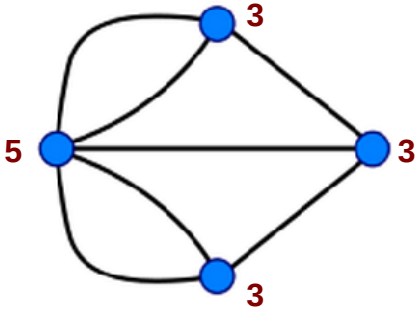
- Son elementos de un conjunto
- Grafos etiquetados (labeled)
- vs no-etiquetados (unlabeled)

**Operaciones con grafos (algunos ejemplos):**

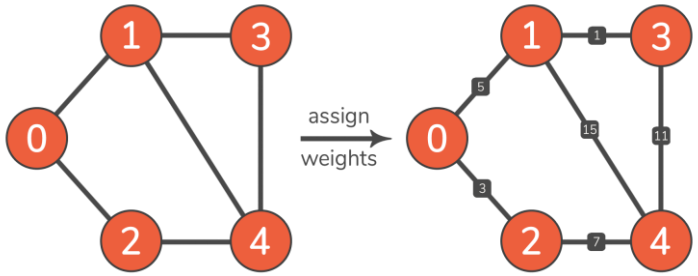
Complementación, Unión, Suma, Intersección, Diferencia, ...



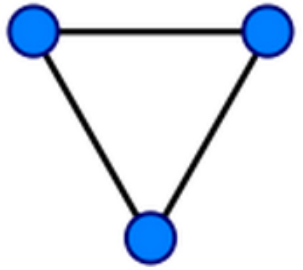
Un grafo etiquetado

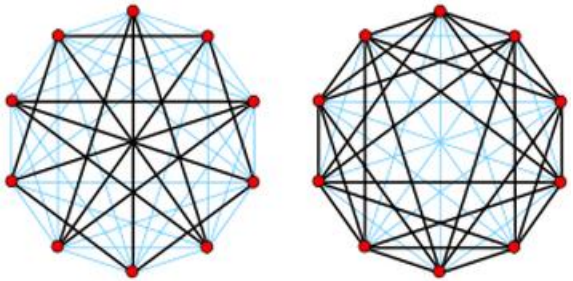


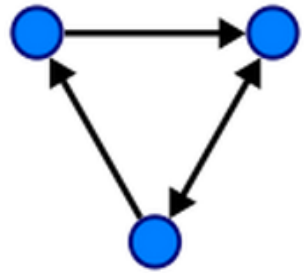
Grados de los nodos



Un grafo pesado


Grafo simple o regular


Un grafo y su complemento


Grafo dirigido (red)

# PROBLEMA: ENCONTRAR MOLÉCULAS IGUALES

**Problema frecuente en química**

Si representamos moléculas como **grafos:**

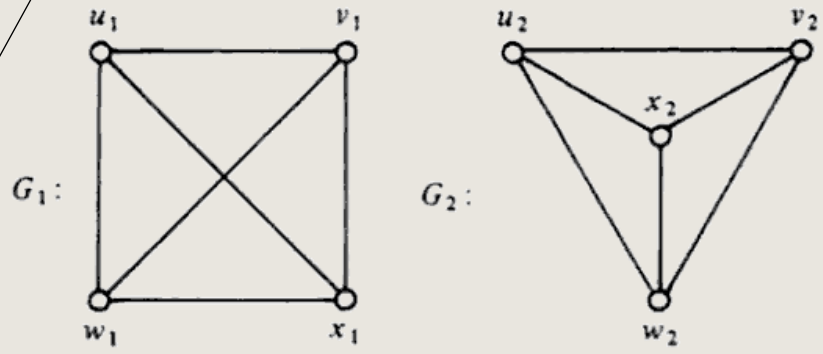
- dos moléculas son la misma si es posible redibujar una de ellas de manera que se vea idéntica a la otra: **Isomorphic graphs**

Problema visualmente interesante, pero la solución es obvia: **solo la conectividad es relevante!**

```
G1: nodos = {u1, v1, w1, x1}

aristas = { {u1,v1}, {u1,w1},{u1,x1}, {v1,x1}, {v1,w1}, {x1,w1} }


G2: nodos = {u2, v2, w2, x2}

aristas = { {u2,v2}, {u2,w2},{u2,x2}, {v2,x2}, {v2,w2}, {x2,w2} }
```

**Problema computacionalmente sencillo (usualmente)**

**Two isomorphic graphs.** Tomado de "Introductory Graph Theory". G. Chartrand (1977). Dover Publications.



# PROBLEMA MÁS DÍFICIL: ENCONTRAR MOLÉCULAS CON GRUPOS SIMILARES

## Otro problema común

**Identificar compuestos que comparten grupos químicos similares**

**Farmacóforos** – grupos químicos responsables de actividad farmacológica

**Grupos reactivos** – carbonilos, aldehidos, cetonas,

**Aplicaciones**

Agrupar compuestos químicos en familias

Desarrollo de nuevas Drogas

Inferencia

*Foye's Principles of Medicinal Chemistry* (2008).
T Lemke, DA Williams. Wolters Kluwer

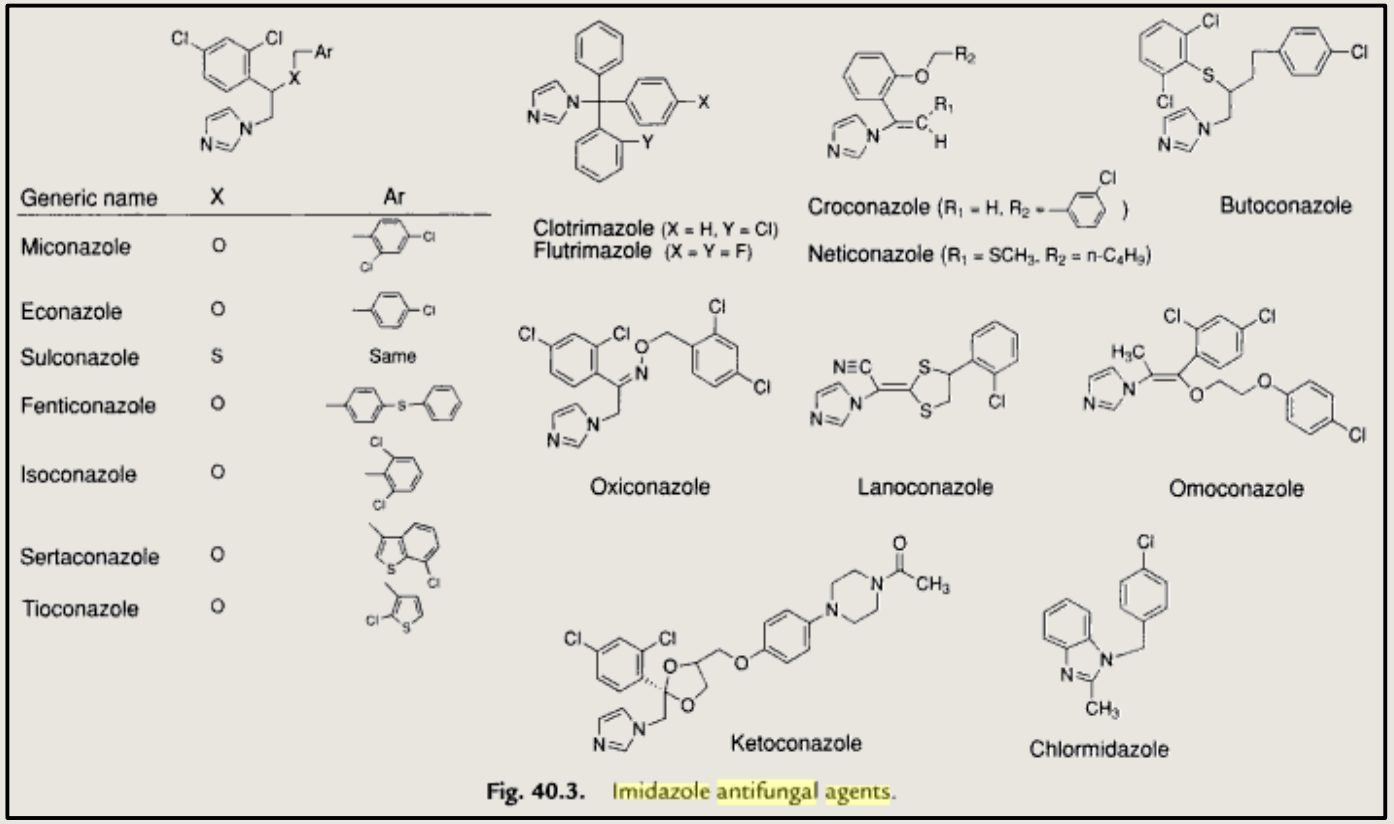


Fig. 40.3. Imidazole antifungal agents.

# PROBLEMA MÁS DÍFICIL: ENCONTRAR MOLÉCULAS CON GRUPOS SIMILARES

Computacionalmente**:** ***subgraph isomorphism problem***

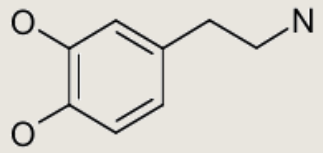
Encontrar un grafo determinado (fijo) dentro de otro grafo

Encontrar el **máximo subgrafo compartido** entre dos grafos

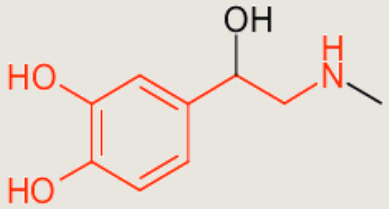
**Es un problema computacionalmente dificil!**

El tiempo se incrementa exponencialmente con el tamaño del problema (en este caso el número de nodos del grafo)

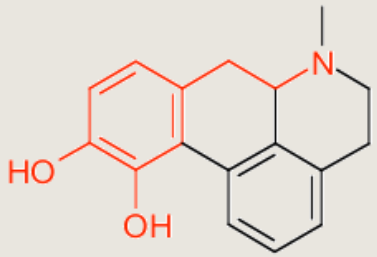
Query:



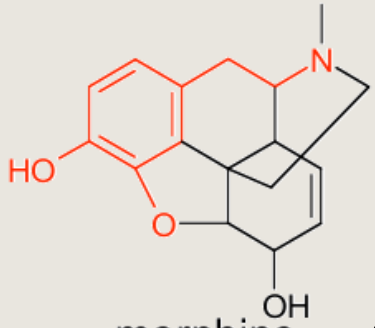

Hits:




adrenaline

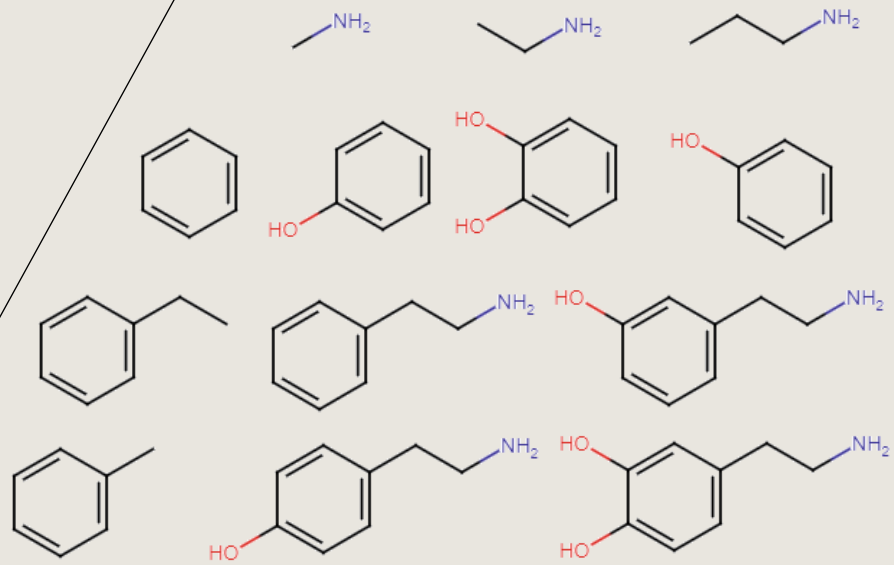



apomorphine

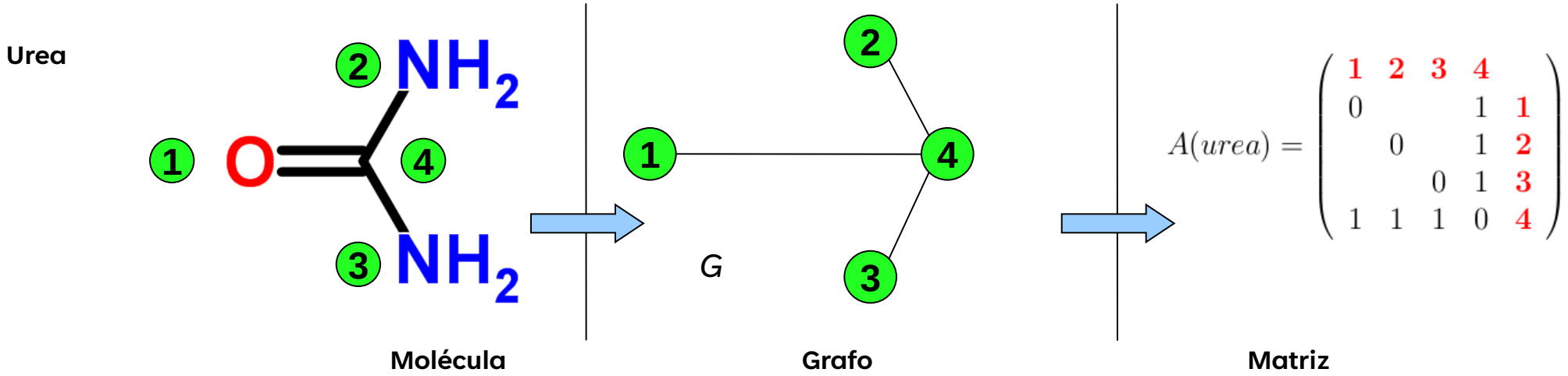



morphine

**Subgrafos compartidos**



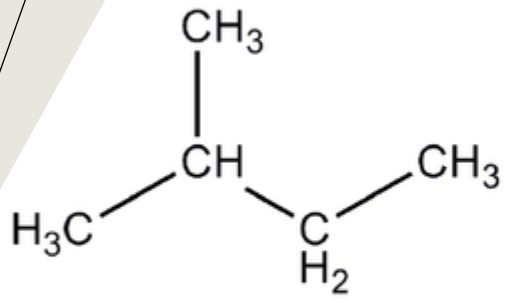
# BÚSQUEDA DE SUBESTRUCTURAS: MATRICES DE ADYACENCIA



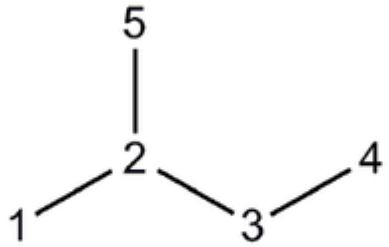

Dado un grafo, es posible construir una **matriz de adyacencia**

Es una aproximación (heurística) a la búsqueda de subestructuras: localizar coincidencias en una matriz de adyacencias

# ADJACENCY MATRICES




Molecule




Graph

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

adjacency matrix

The chemical graph and adjacency matrix of the isopentane.

# BÚSQUEDA DE SUBESTRUCTURAS: MATRICES DE ADYACENCIA

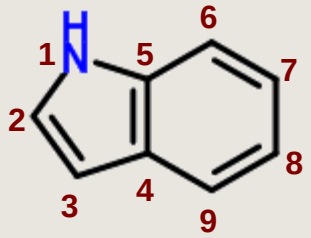
**Indol:** compuesto heterocíclico aromático, precursor de muchas drogas

Búsqueda de compuestos que contengan el grupo **indol**

1. Calcular la matriz de adyacencia para la molécula *'query'*

2. Calcular las matrices de adyacencia para todas las moléculas a testear (la base de datos)

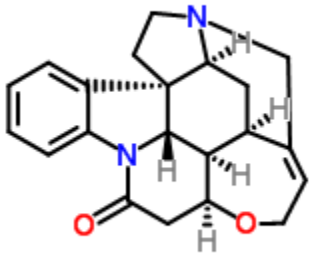
3. Buscar coincidencias en las matrices de adyacencia




**Indole**

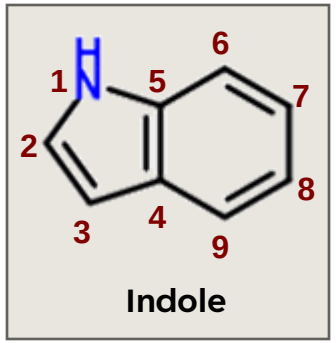
$$A(indole) = \begin{pmatrix} \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} & \mathbf{7} & \mathbf{8} & \mathbf{9} & \\ 0 & 1 & & & 1 & & & & & \mathbf{1} \\ 1 & 0 & 1 & & & & & & & \mathbf{2} \\ & 1 & 0 & 1 & & & & & & \mathbf{3} \\ & & 1 & 0 & 1 & & & & 1 & \mathbf{4} \\ 1 & & & 1 & 0 & 1 & & & & \mathbf{5} \\ & & & & 1 & 0 & 1 & & & \mathbf{6} \\ & & & & & 1 & 0 & 1 & & \mathbf{7} \\ & & & & & & 1 & 0 & 1 & \mathbf{8} \\ & & & 1 & & & & 1 & 0 & \mathbf{9} \end{pmatrix}$$

# BÚSQUEDA DE SUBESTRUCTURAS: MATRICES DE ADYACENCIA



**Strychnine**

**Database Molecule**




**Indole**

**Query Molecule**

$$A(strychnine) = \begin{pmatrix}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 25 \\
1 & 0 & 1 & & & & & & & & & & & & & 1 & & & & & & & & & & & \\
2 & 1 & 0 & 1 & 1 & & & & & & & & & & & & & & & & & & & & & \\
3 & & 1 & 0 & & & & & & & & & & & & & & & & & & & & & & \\
4 & & & 1 & 0 & 1 & & & & & & & & & & & & & 1 & & & & & & & \\
5 & & & & 1 & 0 & 1 & & & & & & & & & & & & & & & 1 & & & & 1 \\
6 & & & & & 1 & 0 & 1 & & & & & 1 & & & & & & & & & & & & & \\
7 & & & & & & 1 & 0 & 1 & & & & & & & & & & & & & & & & & \\
8 & & & & & & & 1 & 0 & 1 & & & & & & & & & & & & & & & & \\
9 & & & & & & & & 1 & 0 & 1 & & & & & & & & & & & & & & & \\
10 & & & & & & & & & 1 & 0 & 1 & & & & & & & & & & & & & & \\
11 & & & & & & & & & & 1 & 0 & 1 & & & & & & & & & & & & & \\
12 & & & & & & & & & & & 1 & 0 & 1 & 1 & & & & & & & & & & & \\
13 & & & & & & & & & & & & 1 & 0 & & & & & & & & & & & & \\
14 & & & & & & & & & & & & 1 & & 0 & & & & & & & & & & & \\
15 & 1 & & & & & & & & & & & & & & 0 & 1 & & & & & & & & & \\
16 & & & & & & & & & & & & & & 1 & 0 & 1 & & & & & & & & & \\
17 & & & & & & & & & & & & & & & 1 & 0 & 1 & & & & & & & & \\
18 & & & & & & & & & & & & & & & & 1 & 0 & 1 & & & & & & & \\
19 & & & & & & & & & & & & & & & & & 1 & 0 & 1 & 1 & & & & & \\
20 & & & & & & & & & & & & & & & & & & 1 & 0 & & & & & & \\
21 & & & & & & & & & & & & & & & & & & & 1 & 0 & 1 & & & & \\
22 & & & & & & & & & & & & & & & & & & & & 1 & 0 & 1 & & & \\
23 & & & & & & & & & & & & & & & & & & & & & 1 & 0 & 1 & & \\
24 & & & & & & & & & & & & & & & & & & & & & & 1 & 0 & 1 & \\
25 & & & & & & & & & & & & & & & & & & & & & & & 1 & 0 &
\end{pmatrix}$$

$$A(indole) = \begin{pmatrix}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & \\
1 & 0 & 1 & & & 1 & & & & & 1 \\
2 & 1 & 0 & 1 & & & & & & & 2 \\
3 & & 1 & 0 & 1 & & & & & & 3 \\
4 & & & 1 & 0 & 1 & & & & 1 & 4 \\
5 & 1 & & & 1 & 0 & 1 & & & & 5 \\
6 & & & & & 1 & 0 & 1 & & & 6 \\
7 & & & & & & 1 & 0 & 1 & & 7 \\
8 & & & & & & & 1 & 0 & 1 & 8 \\
9 & & & & 1 & & & & 1 & 0 & 9
\end{pmatrix}$$

# BÚSQUEDA DE SUBESTRUCTURAS: MATRICES DE ADYACENCIA

**Problema de esta estrategia (hasta acá):**

Puede dar falsos positivos

Grafos que tienen el mismo número de nodos, con la misma adyacencia, pero cuyos nodos están compuestos por distintos átomos (en el caso de moléculas)

**Posible solución:**

**Screening** – realizar la búsqueda sólo sobre un subconjunto de moléculas (grafos) compatibles

Ej: (query = indol) filtrar la base de datos: seleccionar solamente moléculas que tengan al menos 1 átomo de nitrógeno

# MAXIMUM COMMON SUBSTRUCTURE SEARCH

| | Query 1 | Query 2 | Query 3 |
|---|---|---|---|
| Target 1 | Q1-T1 0.50 | Q2-T1 0.30 | Q3-T1 0.53 |
| Target 2 | Q1-T2 0.53 | Q2-T2 0.37 | Q3-T2 0.56 |
| Target 3 | Q1-T3 0.41 | Q2-T3 0.33 | Q3-T3 0.44 |
| Target 4 | Q1-T4 0.29 | Q2-T4 0.59 | Q3-T4 0.53 |

https://docs.chemaxon.com/display/docs/jklustor_maximum-common-substructure-mcs-search.md

# INTERVALO

15 minutos

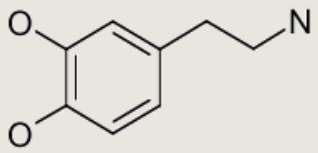
# BÚSQUEDA DE SUBESTRUCTURAS

Screenings

Simple:

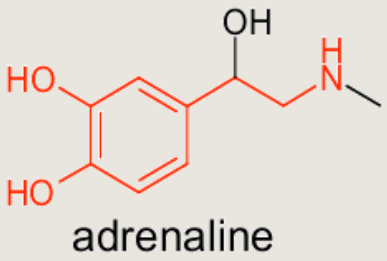
- Usa la fórmula molecular (ej $C_8O_2N$)
  - La fórmula de todos los compuestos está almacenada en la base de datos
  - La fórmula de la molécula *query* se calcula al inicio de la búsqueda
  - Se descartan moléculas a las que les faltan átomos requeridos

Query:

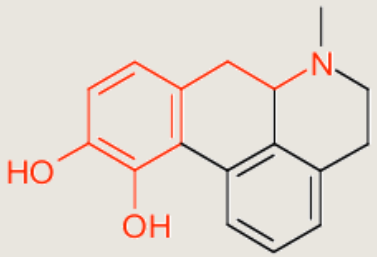



**MF: C8 O2 N (H implícito)**

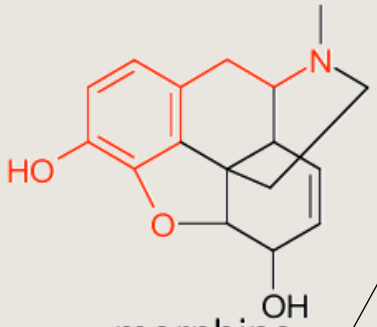
Hits:




adrenaline




apomorphine

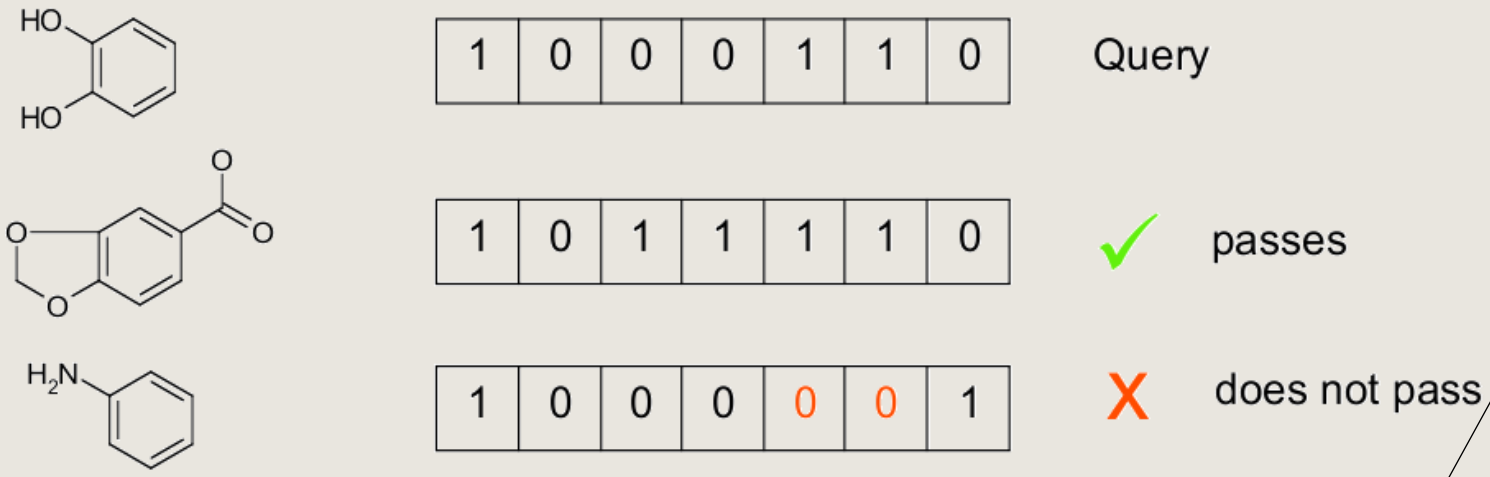



morphine

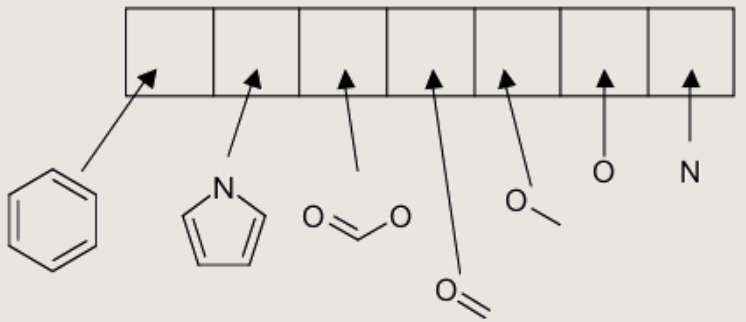
# BÚSQUEDA DE SUBESTRUCTURAS: FINGERPRINTS

**Fingerprint:** representación abstracta de características o propiedades de una molécula (features)

- Presencia/ausencia de cada elemento
- Configuraciones electrónicas inusuales (carbono sp3, nitrógeno unido con un triple enlace)
- Anillos y sistemas de anillos (naftaleno, piridina, cyclohexano)
- Grupos funcionales (alcoholes, aminas, carboxilos, etc.)
- Se suelen utilizar tanto para búsquedas de subestructuras como para detectar similitud




**Un fingerprint**



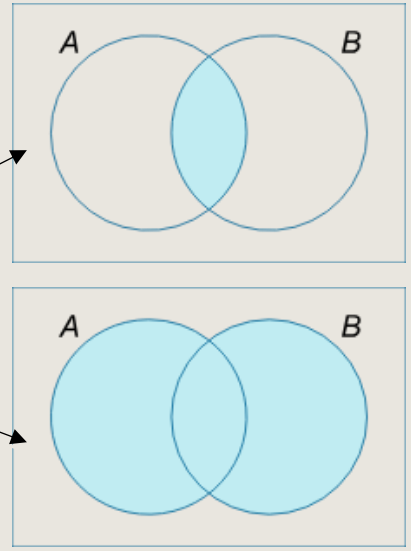
# BÚSQUEDA DE SUBESTRUCTURAS Y SIMILITUD: FINGERPRINTS

**Ventajas:** screening extremadamente rápido

Se evalúa equivalencia entre conjuntos de bits usando el operador AND binario

Se pueden calcular distancias de similitud a partir de los bits significativos

```
X          10001101
Y          01010111
-------------------
X AND Y    00000101
-------------------
X OR  Y    11011111
```



# DISTANCE METRICS: SIMILARITY, DISIMILARITY

**Jaccard index (J) = Jaccard similarity coefficient = Tanimoto Index = Tanimoto similarity coefficient**

*(tambien llamado "Intersection Over Union")*

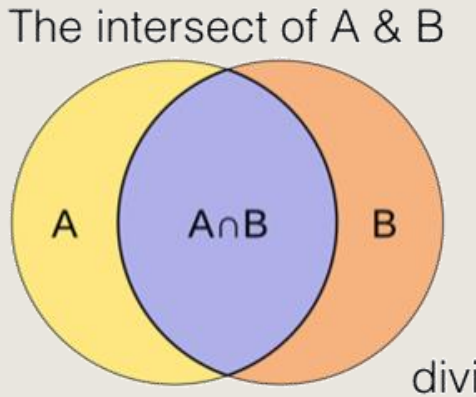
Compara similitudes entre conjuntos de datos finitos

**Jaccard distance ($d_J$)**

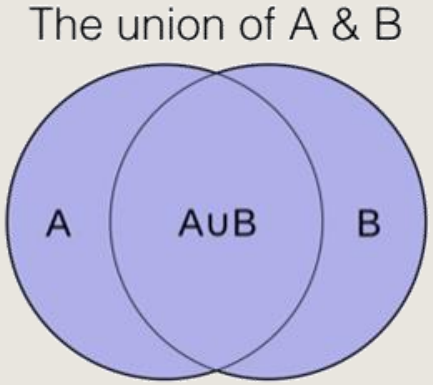
$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

$$d_J(A,B) = 1 - J(A,B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}$$

Cociente entre el tamaño de la intersección y el tamaño de la unión de los conjuntos de datos




division




https://en.wikipedia.org/wiki/Jaccard_index

# DISTANCE METRICS

***There are many ways to measure distances...***

Safizadeh H, *et al*. Improving Measures of Chemical Structural Similarity Using Machine Learning on Chemical-Genetic Interactions. J Chem Inf Model **61,** 4156-4172 (2021). https://doi.org/ 10.1021/acs.jcim.0c00993.

Raymond, J.W., Willett, P. Effectiveness of graph-based and fingerprint-based similarity measures for virtual screening of 2D chemical structure databases. *J Comput Aided Mol Des* **16**, 59–71 (2002). https://doi.org/10.1023/A:1016387816342.

sqrt = square root $\sqrt{\phantom{x}}$

X = number of bits **set** in ***both fingerprints***
Y = number of bits **set** in the ***first fingerprint***
Z = number of bits **set** in the ***second fingerprint***
W = total number of bits in the bitstring

| Name | Measurement | Range |
|---|---|---|
| Braun-Blanquet | x / max(x,y) | 0 – 1 |
| Cosine | x / sqrt(yz) | 0 – 1 |
| Dice | 2x / (y + z) | 0 – 1 |
| Dot product | x | 0 – ∞ |
| Euclidean | 1 / 1 + sqrt( y + z – 2x) | 0 – 1 |
| Kulczunksi | | 0 – 1 |
| McConnaughey | x (y + z) – yz / yz | -1 – 1 |
| Russel / Rao | x / w | 0 – 1 |
| Simpson | x / min(y,z) | 0 – 1 |
| Sokal / Sneath | x / (2y + 2z – 3x) | 0 – 1 |
| Tanimoto / Jaccard | x ( y + z – x) | 0 – 1 |
| Tullos | xyz | 0 – 1 |
| Tversky | x / α (y-x) + (1 – α) (z – x) + x | 0 – 1 |

# MOLECULAR FINGERPRINTS

***There are many ways to fingerprint molecules ...***

**Table 1. Molecular Fingerprints**[a]

| ID | name | description | features | reference(s) |
|---|---|---|---|---|
| FP1 | AP2D | topological atom pairs | 1211 | (44) |
| FP2 | ASP | all-shortest paths | 26,194 | (45) |
| FP3 | AT2D | topological atom triplets | 56,963 | (44) |
| FP4 | DFS | all-paths (depth-first search) | 48,448 | (46) |
| FP5 | ECFP | extended connectivity fingerprints | 42,672 | (47) |
| FP6 | LSTAR | local path environments | 85,232 | (48) |
| FP7 | MACCS | MDL public keys (166 keys) | 155 | (49) |
| FP8 | PHAP2POINT2D | topological pharmacophore pairs | 17 | (50) |
| FP9 | PHAP3POINT2D | topological pharmacophore triplets | 302 | (50) |
| FP10 | RAD2D | topological molprint-like fingerprints | 92,191 | (48) |
| FP11 | RDKit | topological daylight-like fingerprints | 65,183 | (43,51) |

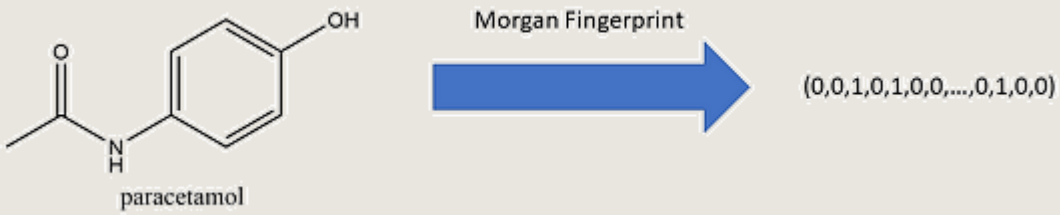
Safizadeh H, *et al*. Improving Measures of Chemical Structural Similarity Using Machine Learning on Chemical-Genetic Interactions. J Chem Inf Model **61,** 4156-4172 (2021). https://doi.org/ 10.1021/acs.jcim.0c00993.

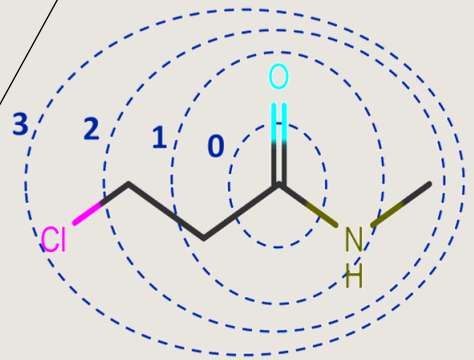
# EXTENDED CONNECTIVITY FINGERPRINTS

## *Circular fingerprints*

Concepto similar al de "***extended connectivity***" de Morgan

1. Assign each atom with an identifier
2. Update each atom's identifiers based on its neighbors
3. Remove duplicates
4. Fold list of identifiers into a 2048-bit vector (a Morgan fingerprint)




Extended Connectivity Circular Fingerprints

**ECFP6 (radius = 3)**

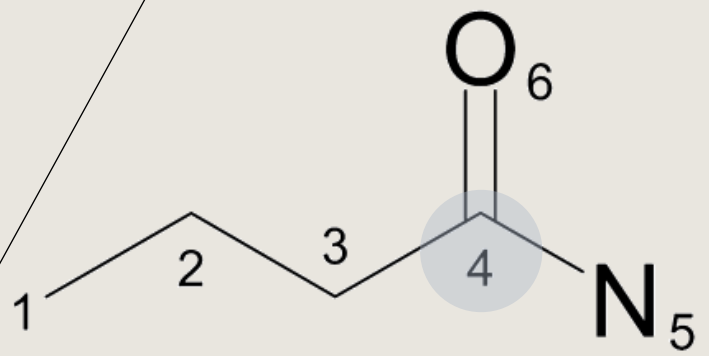
1024 or 2048 bits

# EXTENDED CONNECTIVITY FINGERPRINTS

**1. Assign each atom with an identifier**

We choose an atom in the molecule (e.g. #4) and take note of:

- number of nearest-neighbour non-hydrogen atoms: **3**
- number of bonds attached to the atom (not including bonds to hydrogens): **4**
- atomic number: **6**
- atomic mass: **12**
- number of hydrogens connected to the atom: **0**
- is the atom in a ring (1) or not (0)?: **0**
- **Resulting list of numbers is (3,4,6,12,0,0)**
- **Hash this list of numbers into an integer (identifier)**
  - In Python: `hash((3, 4, 6, 12, 0, 0, 0))` ➔ `-5700861834356229464`




A beginner's guide for understanding Extended-Connectivity Fingerprints(ECFPs). Manish Kumar (2021). https://chemicbook.com/2021/03/25/a-beginners-guide-for-understanding-extended-connectivity-fingerprints.html

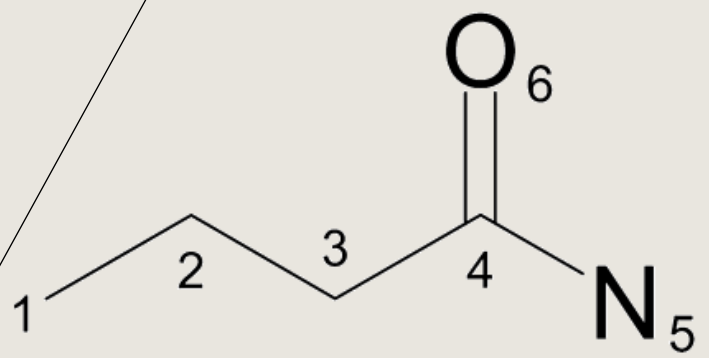
# EXTENDED CONNECTIVITY FINGERPRINTS

```
# identificadores para cada atomo
atomo1 = hash((1, 1, 6, 12, 0, 3, 0)) # -CH3
atomo2 = hash((2, 2, 6, 12, 0, 2, 0)) # -CH2
atomo3 = hash((2, 2, 6, 12, 0, 2, 0)) # -CH2
atomo4 = hash((3, 4, 6, 12, 0, 0, 0)) # -C
atomo5 = hash((1, 2, 7, 14, 0, 0, 0)) # -NH2
atomo6 = hash((1, 2, 8, 16, 0, 0, 0)) # =O
```

```
atomo 1 4940186308562569707
atomo 2 -78159851478978265 76
atomo 3 -78159851478978265 76
atomo 4 -57008618343562294 64
atomo 5 -62963877442778008 66
atomo 6 86184117556823738 92
```

**List of features (6)**

https://andrewbrookins.com/technology/pythons-default-hash-algorithm/




A beginner's guide for understanding Extended-Connectivity Fingerprints(ECFPs). Manish Kumar (2021). https://chemicbook.com/2021/03/25/a-beginners-guide-for-understanding-extended-connectivity-fingerprints.html

# EXTENDED CONNECTIVITY FINGERPRINTS

**Update each atom's identifiers based on its neighbors**

Each atom collects its identifier and the identifiers of its immediately neighboring atoms, into an array (list)

And we hash this list again into a new identifier.

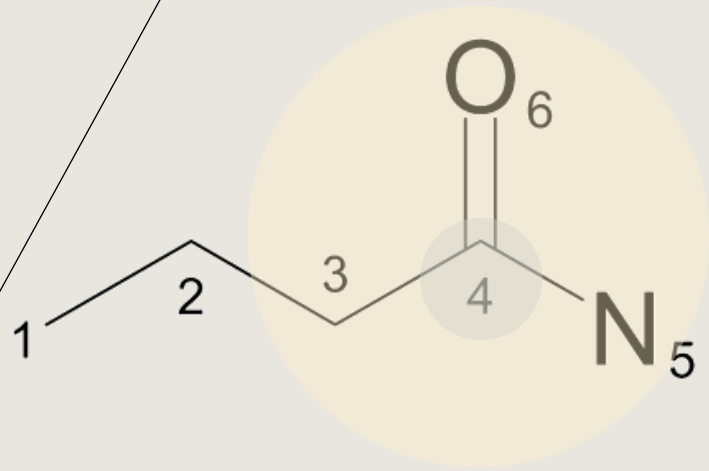
**Paso anterior**

```
atomo 1 4940186308562569707
atomo 2 -7815985147897826576
atomo 3 -7815985147897826576
atomo 4 -5700861834356229464
atomo 5 -6296387744277800866
atomo 6 8618411755682373892
```

```
atomo4_updated = hash((
  1, -5700861834356229464,
  1, -7815985147897826576,
  1, -6296387744277800866,
  2, 8618411755682373892
))

-6784272694619664722
```

repetimos para los 6 átomos

**List of features (12)**

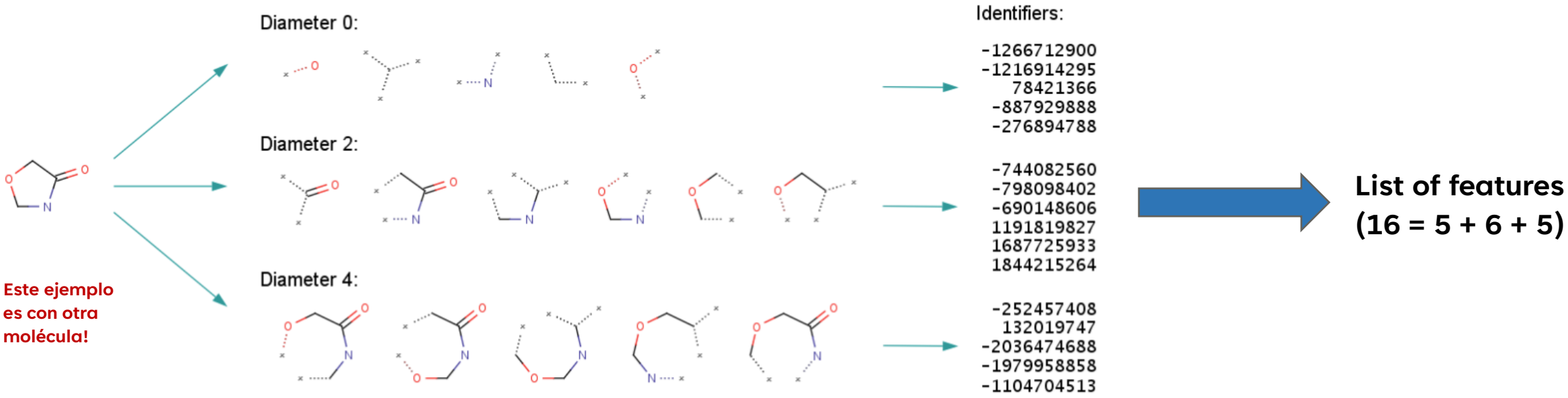



A beginner's guide for understanding Extended-Connectivity Fingerprints(ECFPs). Manish Kumar (2021). https://chemicbook.com/2021/03/25/a-beginners-guide-for-understanding-extended-connectivity-fingerprints.html

# EXTENDED CONNECTIVITY FINGERPRINTS

- After that, several iterations are performed to combine the initial atom identifiers with identifiers of neighboring atoms ***until a specified diameter is reached***. Each iteration captures larger and larger circular neighborhoods around each atom

- ECFP4 = Extended Circular Fingerprint with **diameter = 4** ***(radius = 2)***
- ECFP6 = Extended Circular Fingerprint with **diameter = 6** ***(radius = 3)***



# FINGERPRINTS: FOLDING AND BIT COLLISIONS

Para acomodar estos ***features*** en un fingerprint de 1024 bits

- Inicializar el fingerprint con **todos los bits en 0 (OFF)**
- Dividir cada identificador por 1024, y anotar el **resto de la división**
  - **En Python: operador módulo (%)**
- **Ese es el número de bit ➔ que se pone en 1 (ON)**


**Ejemplos:**

```
132019747  % 1024 = 547
1687725933 % 1024 = 877
-798098402 % 1024 = 30
```

**Bit Collision:**

```
-14439656419269748   % 1024 = 908
-4080868480043360372 % 1024 = 908
```

**Solution: increase fingerprint size**

```
-14439656419269748   % 2048 = 908
-4080868480043360372 % 2048 = 1932
```

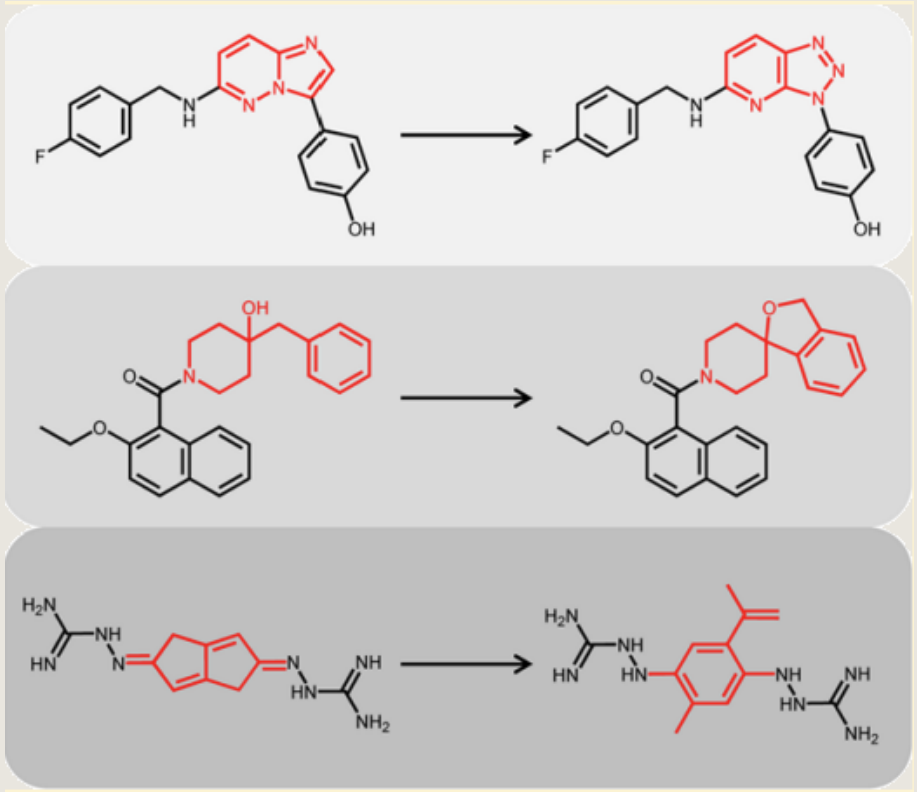
**Folding**

**Fixed-length binary representation**

0001000000001000001000000110010000100010000000000000000100000[...]0000100000000000010

# SCAFFOLD HOPING

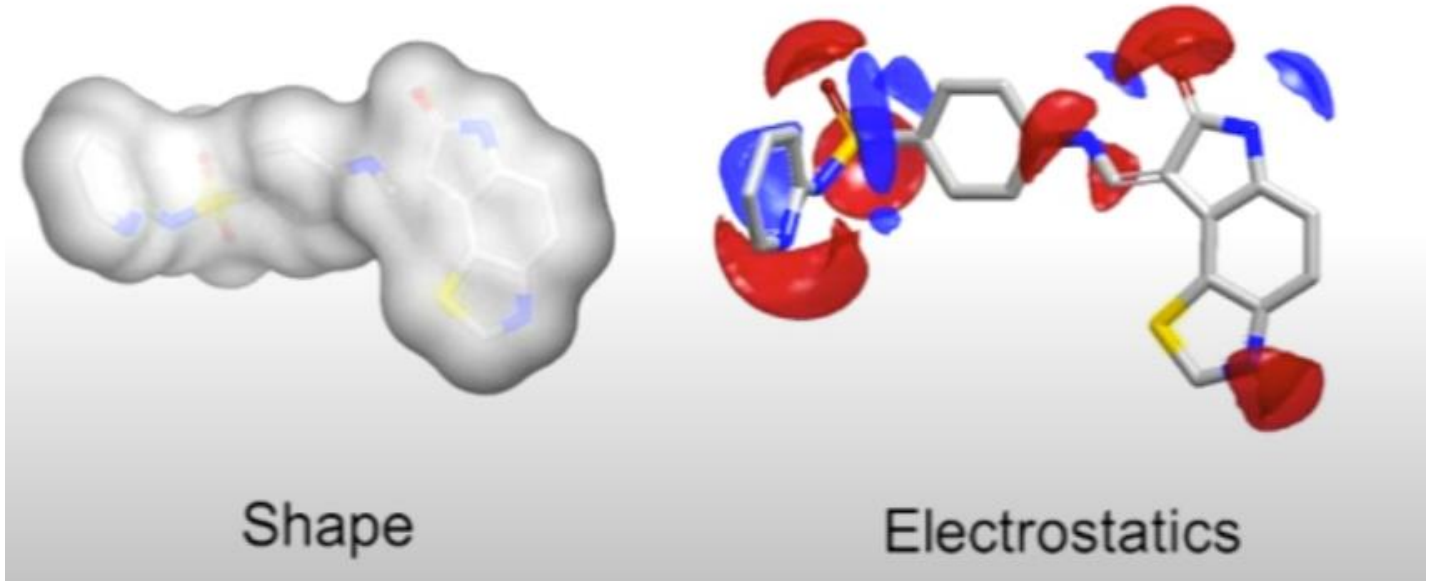
*"the identification of **isofunctional** molecular structures with chemically completely different core structures"*



*computer-aided search for active compounds containing different core structures*

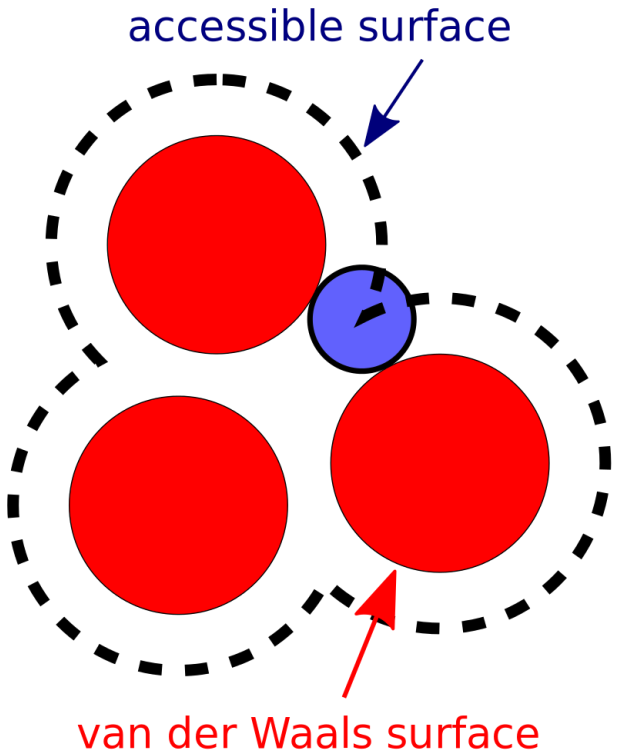
Hu Y, Stumpfe D, Bajorath J. Recent Advances in Scaffold Hopping. J Med Chem. 2017 Feb 23;60(4):1238-1246. doi: 10.1021/acs.jmedchem.6b01437. Epub 2016 Dec 21. PMID: 28001064.

# OTRAS REPRESENTACIONES DE MOLÉCULAS



# SOLVENT ACCESSIBLE SURFACE AREA CALCULATION

- VSA = van der Waals Surface Area
- AS = Accessible Surface Area



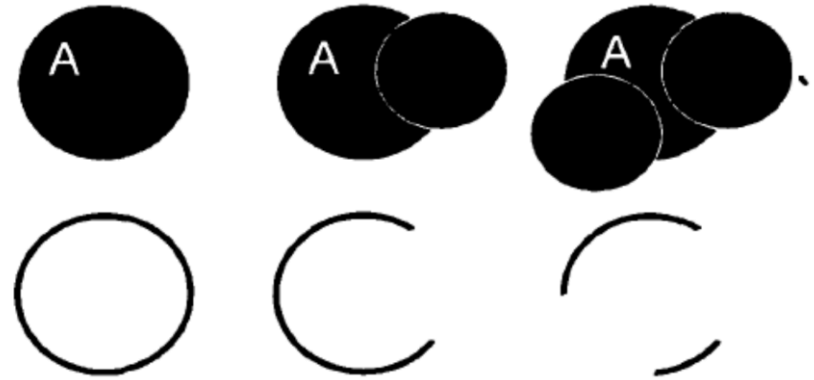



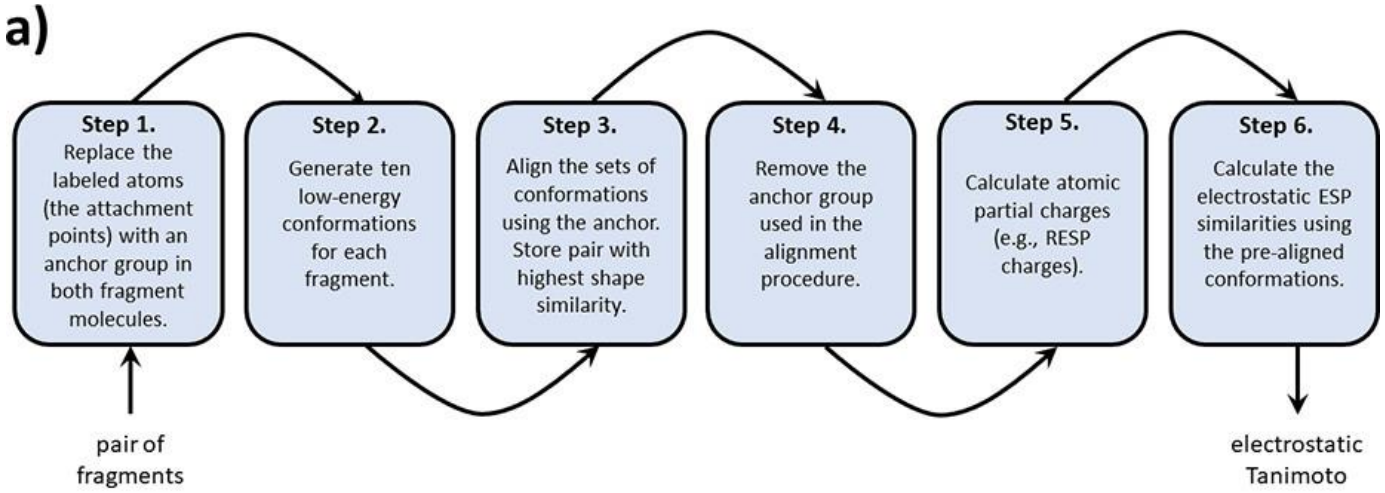

*Figure 1. Assuming spherical atoms, the surface area of atom A is the amount of surface area not contained in other atoms.*

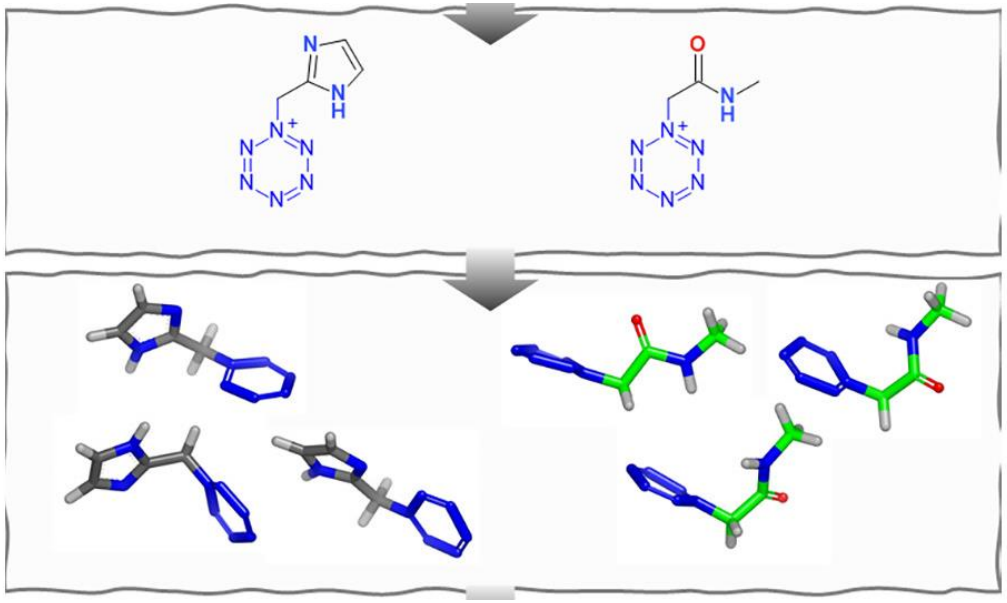
Mitternacht S. FreeSASA: An open source C library for solvent accessible surface area calculations. F1000Res. 2016 Feb 18;5:189. doi: 10.12688/f1000research.7931.1. PMID: 26973785; PMCID: PMC4776673.

# ESP-SIM: COMPARISON OF ELECTROSTATIC POTENTIAL AND SHAPE

https://github.com/hesther/espsim

Bolcato G, Heid E, Boström J. On the Value of Using 3D Shape and Electrostatic Similarities in Deep Generative Methods. J Chem Inf Model. 2022 Mar 28;62(6):1388-1398. doi: 10.1021/acs.jcim.1c01535. Epub 2022 Mar 10. PMID: 35271260; PMCID: PMC8965872.

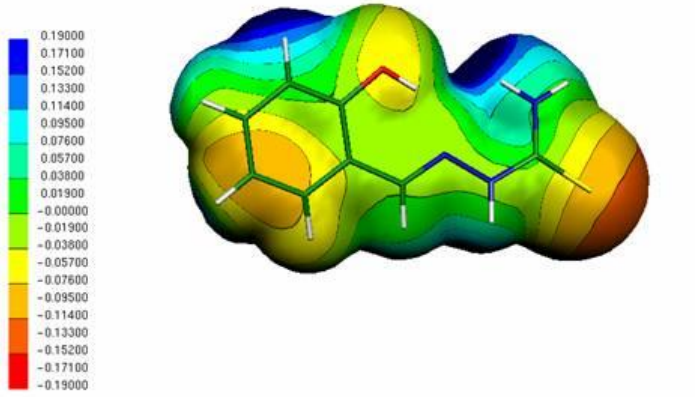


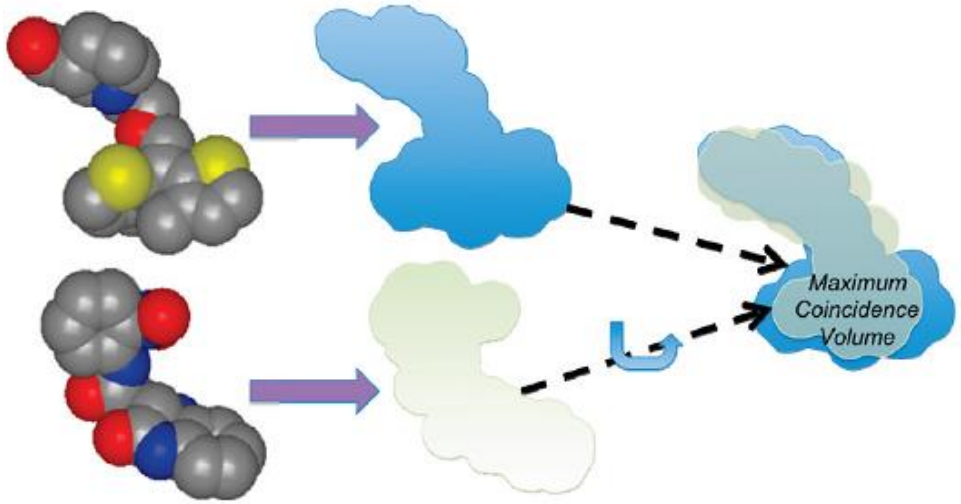



https://github.com/hesther/espsim/blob/master/scripts/short_demonstration.ipynb

# REPRESENTACIÓN DE MOLECULAS: 3D

- Una representación tridimensional de la molécula requiere no sólo especificar coordenadas espaciales de átomos
  - También hay que especificar
    - ***Volumen***
      - **Fused spheres**
      - **Atom-centered Gaussians**
    - ***Superficie***
    - ***Forma***
      - **Coincidencia de volumenes**



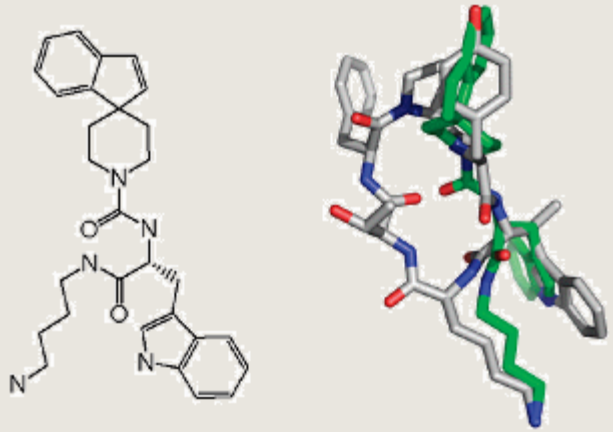




Molecular shape and medicinal chemistry: a perspective. 2010. A Nicholls *et al*. J Med Chem 53: 3862

# REPRESENTACIÓN DE FORMA (SHAPE)

Varias aplicaciones posibles:

- Búsqueda de moléculas similares
  - En este caso la similitud es a nivel de forma
    - Se pueden agregar adicionalmente limitaciones
  - Varias implementaciones en la industria farmacéutica
    - Virtual screening
      - Varios casos de éxito conocidos
      - Merck, primer aplicación publicada del método
        - Identificación de análogos no-peptídicos de:
          - antagonista endógeno del receptor de fibrinógeno (Arg-Gly-Pro)
          - Somatotrophin release inhibitor factor



# REPRESENTACIÓN DE FORMA (SHAPE)

KIN: Bristol-Myers Squibb

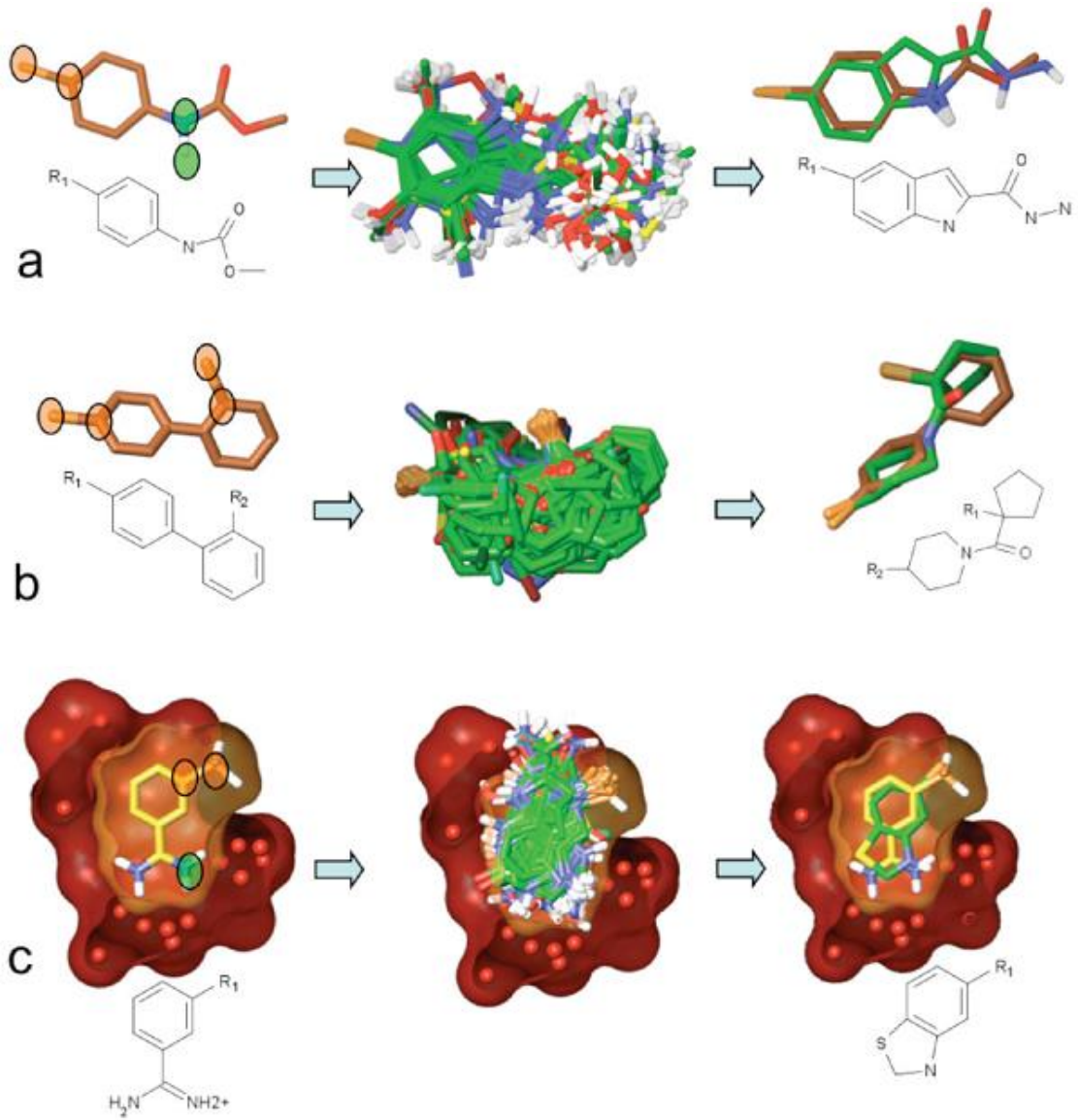
Varias aplicaciones posibles:

**Lead optimization**

Uno cuenta con una molécula activa que quiere optimizar

**Scaffold Hoping**

Facilmente explorable utilizando metodos computacionales




Molecular shape and medicinal chemistry: a perspective. 2010. A Nicholls *et al.* J Med Chem 53: 3862

# CALCULO DE PROPIEDADES

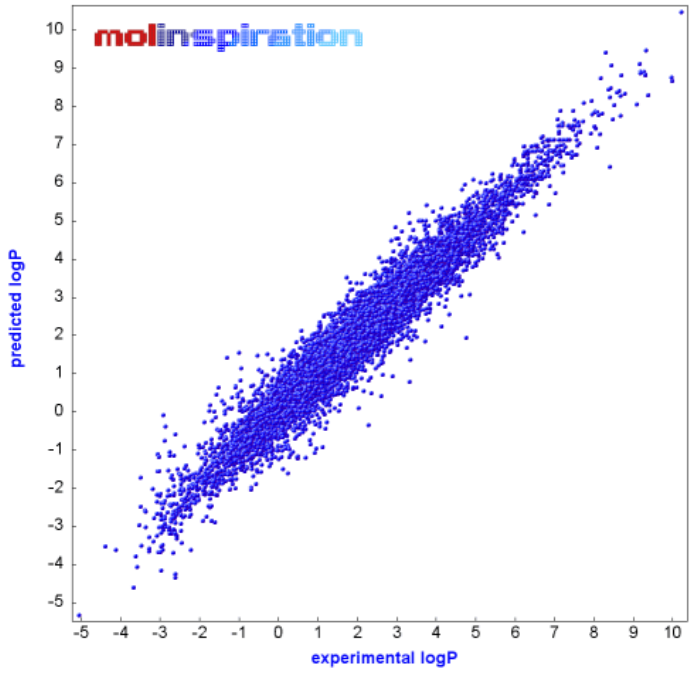
Enlaces rotables

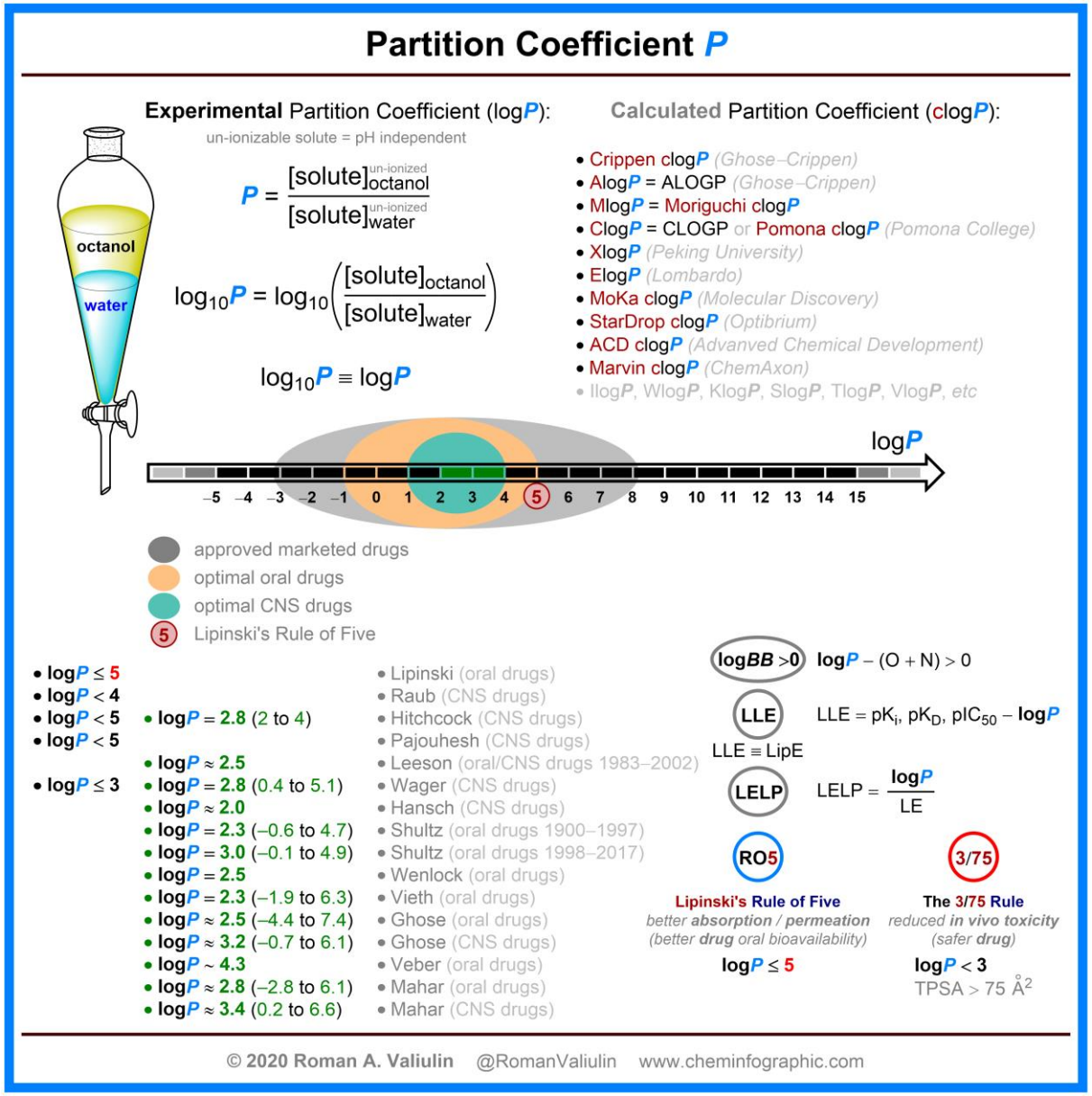
Dadores / Aceptores de puentes de hidrógeno

cLogP (coeficiente de partición octanol / agua)

PSA (polar surface area) / TPSA (topological surface area)

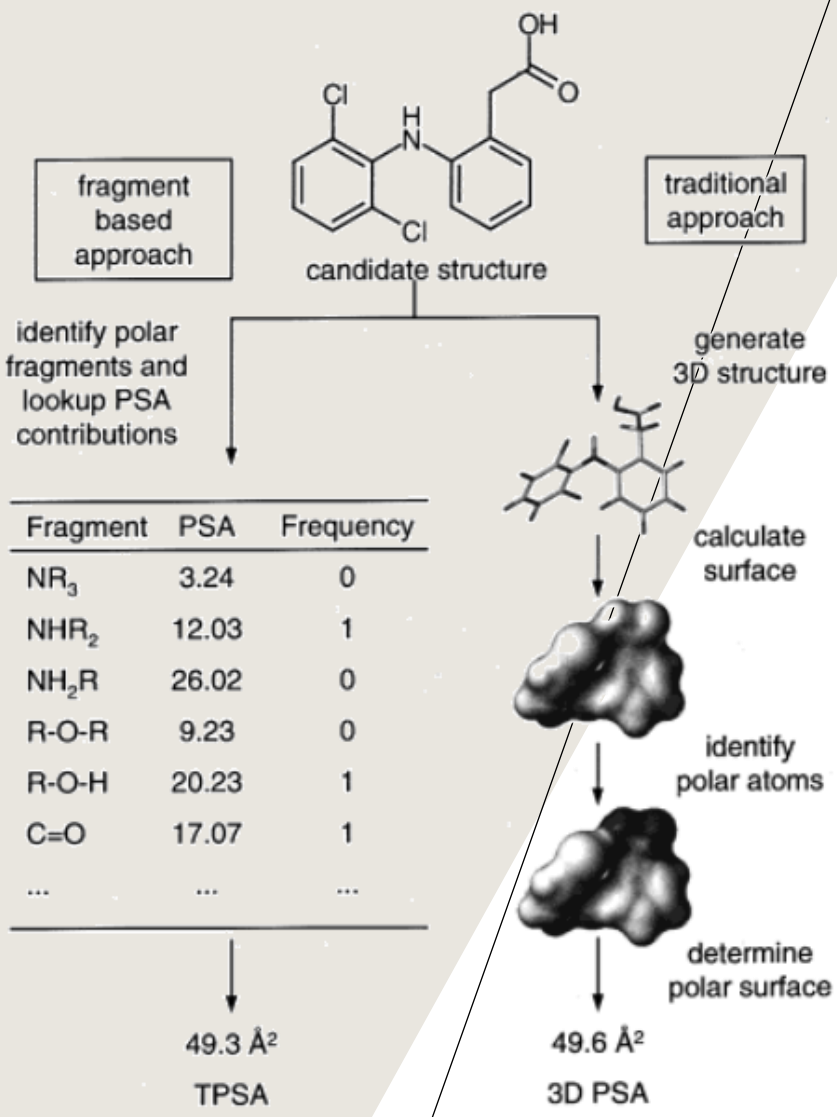
# LOGP PARTITION COEFFICIENT



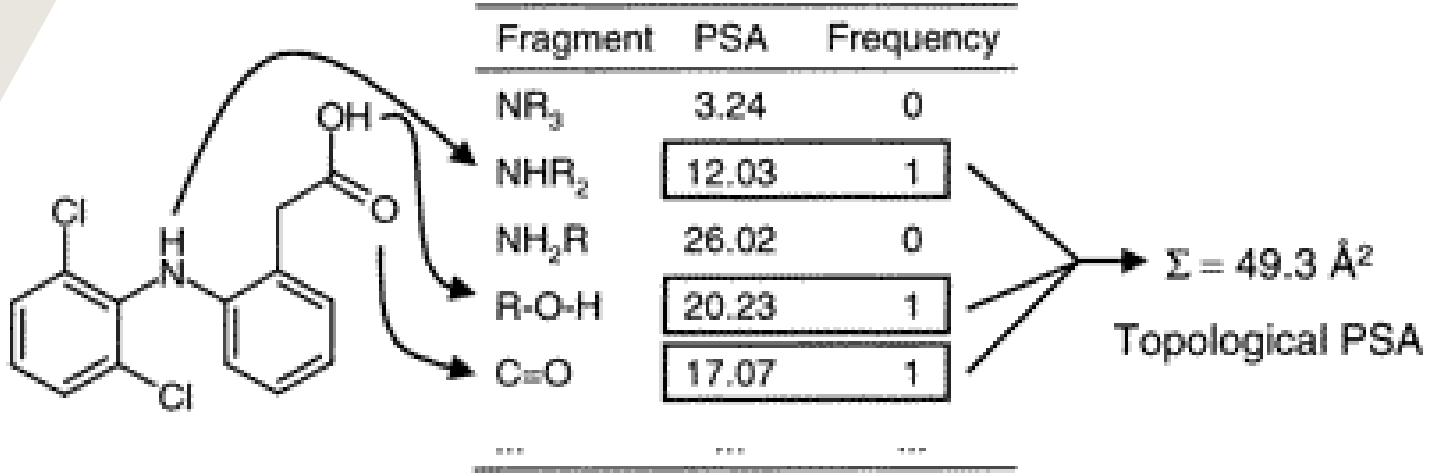




https://cheminfographic.wordpress.com/2020/05/01/partition-coefficient-p-logp/

# PSA / TPSA




| Fragment | PSA | Frequency |
|---|---|---|
| $NR_3$ | 3.24 | 0 |
| $NHR_2$ | 12.03 | 1 |
| $NH_2R$ | 26.02 | 0 |
| R-O-R | 9.23 | 0 |
| R-O-H | 20.23 | 1 |
| C=O | 17.07 | 1 |
| ... | ... | ... |




**Polar Surface Area (PSA, costoso)**

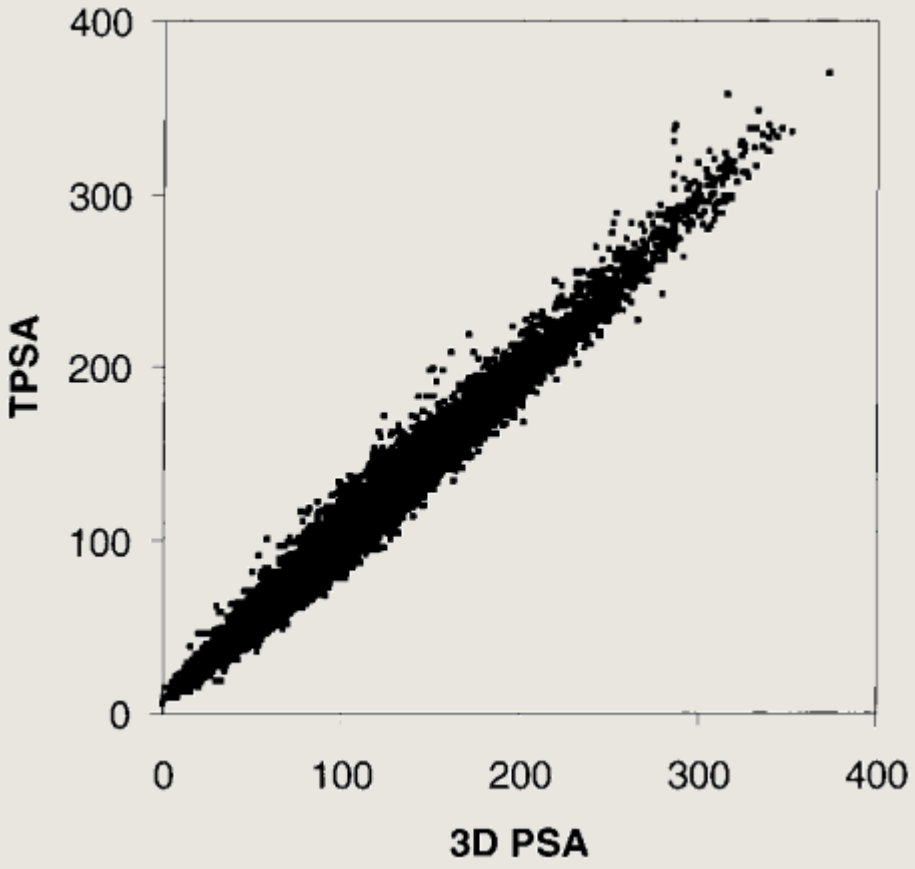
- Requiere generar conformeros 3D para calcular SA (Surface Area)

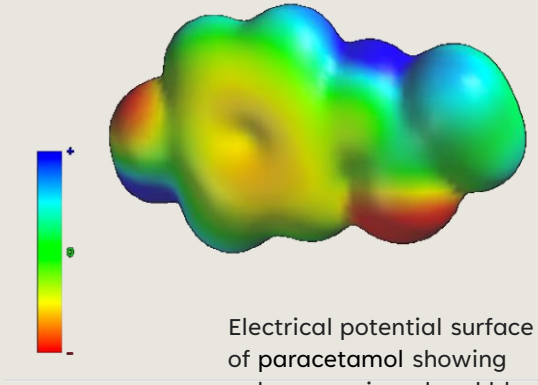
**Topological Polar Surface Area (TPSA)**

- Sumatoria de contribuciones tabuladas de fragmentos polares


Ertl P, Rohde B, Selzer P. Fast calculation of molecular polar surface area as a sum of fragment-based contributions and its application to the prediction of drug transport properties. J Med Chem. 2000 Oct 5;43(20):3714-7. doi: 10.1021/jm000942e. PMID: 11020286.

# TPSA VS PSA (3D)

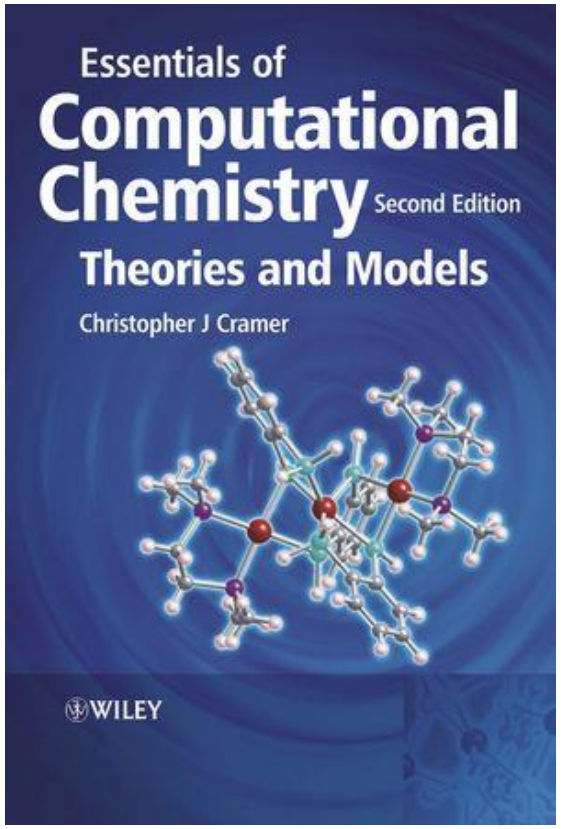


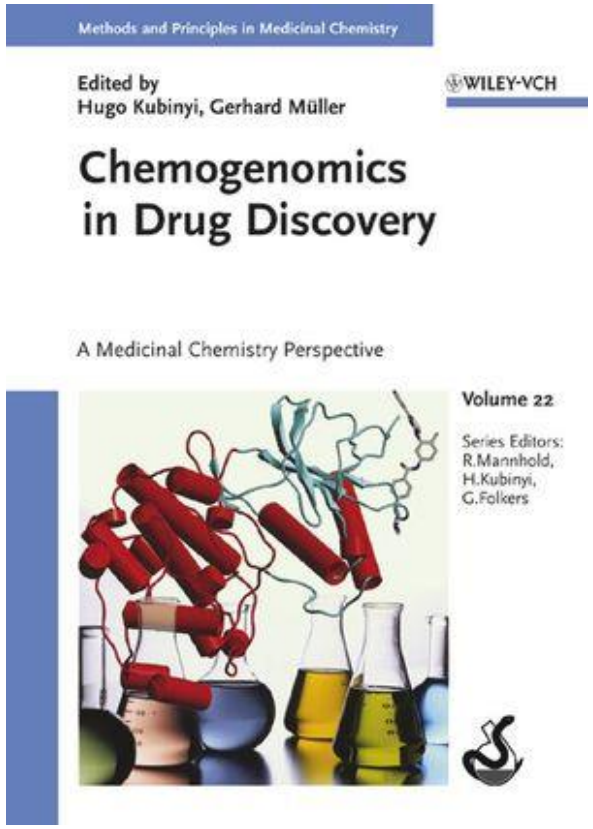
Electrical potential surface of **paracetamol** showing polar areas in red and blue

Ertl P, Rohde B, Selzer P. Fast calculation of molecular polar surface area as a sum of fragment-based contributions and its application to the prediction of drug transport properties. J Med Chem. 2000 Oct 5;43(20):3714-7. doi: 10.1021/jm000942e. PMID: 11020286.

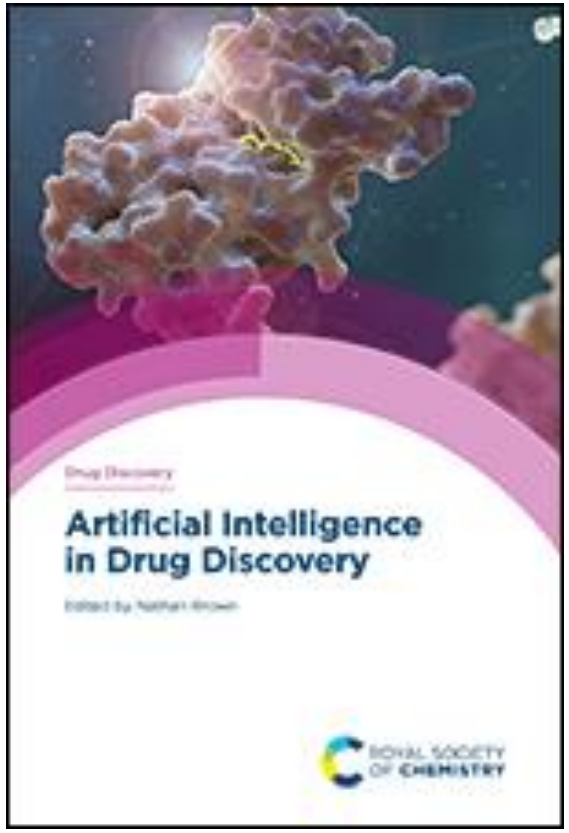
# BIBLIOGRAFÍA | MATERIAL DE LECTURA



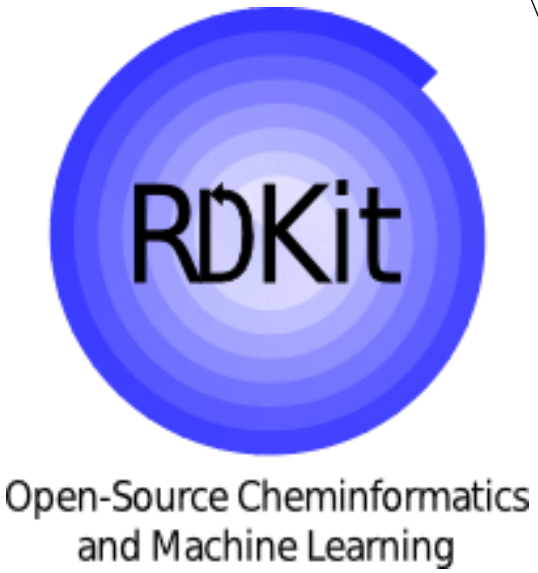

Essentials of Computational Chemistry (2004), 2nd Ed, CJ Cramer. Wiley. https://www.wiley.com/en-sg/Essentials+of+Computational+Chemistry:+Theories+and+Models,+2nd+Edition-p-9780470091821




Chemogenomics in Drug Discovery: A Medicinal Chemistry Perspective (2006). Edited by Hugo Kubinyi & Gerhard Müller, Wiley-VCH. https://www.wiley.com/en-us/Chemogenomics+in+Drug+Discovery%3A+A+Medicinal+Chemistry+Perspective-p-9783527604029



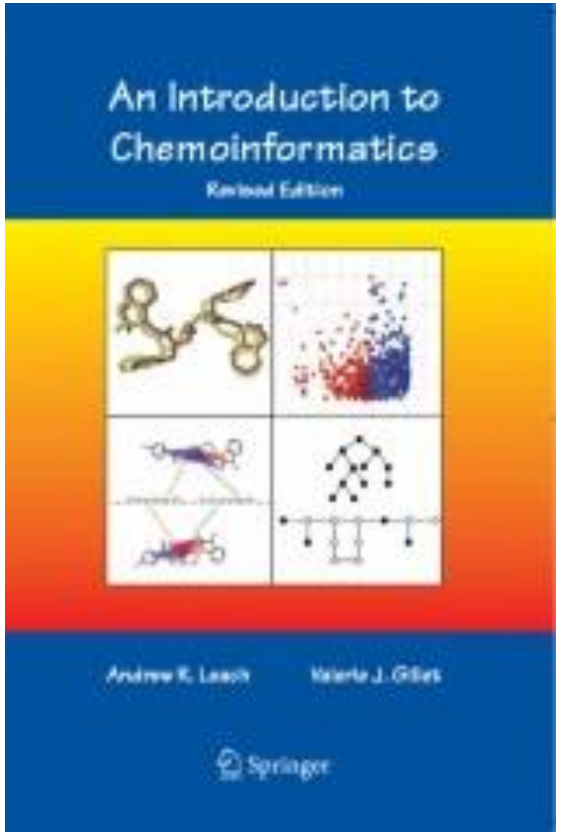

Artificial Intelligence in Drug Discovery (2020). Edited by Nathan Brown. Royal Society of Chemistry. https://doi.org/10.1039/9781788016841




The RDKit Book (2023). https://www.rdkit.org/docs/RDKit_Book.html

# BIBLIOGRAFÍA | MATERIAL DE LECTURA


https://www.ebi.ac.uk/chebi/aboutChebiForward.do




An Introduction to Chemoinformatics (2007). Andrew Leach and Valerie Gillet. Springer. https://link.springer.com/book/10.1007/978-1-4020-6291-9


The Chemistry Development Kit (CDK), v2.0: atom typing, depiction, molecular formulas and substructure searching. https://link.springer.com/article/10.1186/s13321-017-0220-4


Dalke J Cheminform (2019) 11:76
https://doi.org/10.1186/s13321-019-0398-8

Journal of Cheminformatics

METHODOLOGY Open Access

The chemfp project

Andrew Dalke*

**Abstract**

The chemfp project has had four main goals: (1) promote the FPS format as a text-based exchange format for dense binary cheminformatics fingerprints, (2) develop a high-performance implementation of the BitBound algorithm that could be used as an effective baseline to benchmark new similarity search implementations, (3) experiment with funding a pure open source software project through commercial sales, and (4) publish the results and lessons learned as a guide for future implementors. The FPS format has had only minor success, though it did influence development of the FPB binary format, which is faster to load but more complex. Both are summarized. The chemfp benchmark and the no-cost/open source version of chemfp are proposed as a reference baseline to evaluate the effectiveness of other similarity search tools. They are used to evaluate the faster commercial version of chemfp, which can test 130 million 1024-bit fingerprint Tanimotos per second on a single core of a standard x86-64 server machine. When combined with the BitBound algorithm, a k = 1000 nearest-neighbor search of the 1.8 million 2048-bit Morgan fingerprints of ChEMBL 24 averages 27 ms/query. The same search of 970 million PubChem fingerprints averages 220 ms/query, making chemfp one of the fastest CPU-based similarity search implementations. Modern CPUs are fast enough that memory bandwidth and latency are now important factors. Single-threaded search uses most of the available memory bandwidth. Sorting the fingerprints by popcount improves memory coherency, which when combined with 4 OpenMP threads makes it possible to construct an N × N similarity matrix for 1 million fingerprints in about 30 min. These observations may affect the interpretation of previous publications which assumed that search was strongly CPU bound. The chemfp project funding came from selling a purely open-source software product. Several product business models were tried, but none proved sustainable. Some of the experiences are discussed, in order to contribute to the ongoing conversation on the role of open source software in cheminformatics.

**Keywords:** Molecular fingerprints, Similarity searching, Tanimoto, High-performance, Format, Open source, FOSS, Performance benchmark

**Introduction**

Molecular similarity search is a fundamental concept in cheminformatics. The most common form is almost certainly a Tanimoto similarity search of bitstring fingerprints. Complete search systems are available from many vendors, or a good programmer can implement a basic system with reasonable search performance in only a few hours. High-performance search systems, which combine fast popcount evaluation and pruning algorithms, require significantly more development effort. This paper starts with a review of those approaches, many of which are either described in the cheminformatics literature in an incrementalist fashion which make them difficult to discover, or only published in the specialist literature of other fields.

The chemfp project started in order to develop a de facto file format for chemical fingerprints. This requires some consideration of why such a format did not already exist, in order to understand which factors to focus on during format development. Two formats were developed; the text-based FPS exchange format, which is simple to read and write, easily compressed, and appropriate for streaming workflows, and the binary FPB application format which is more complex and requires random-access reads, but has significantly shorter load times.

The chemfp package for Python includes optimized threshold and $k$-nearest implementations FPS file scan search implementations, highly-optimized implementations of the BitBound pruning method to search data

*Correspondence: dalke@dalkescientific.com
Andrew Dalke Scientific AB, Trollhättan, Sweden

BMC



The ChemFP Project (2019). https://link.springer.com/article/10.1186/S13321-019-0398-8

# PREGUNTAS?

Fernán Agüero

fernan@iib.unsam.edu.ar