Friedrich-Alexander-Universität
Naturwissenschaftliche Fakultät

43384 – Digital Alchemy

Unit 02 – Molecular Structure Representations (Basics)

Prof. Dr. Carolin Müller

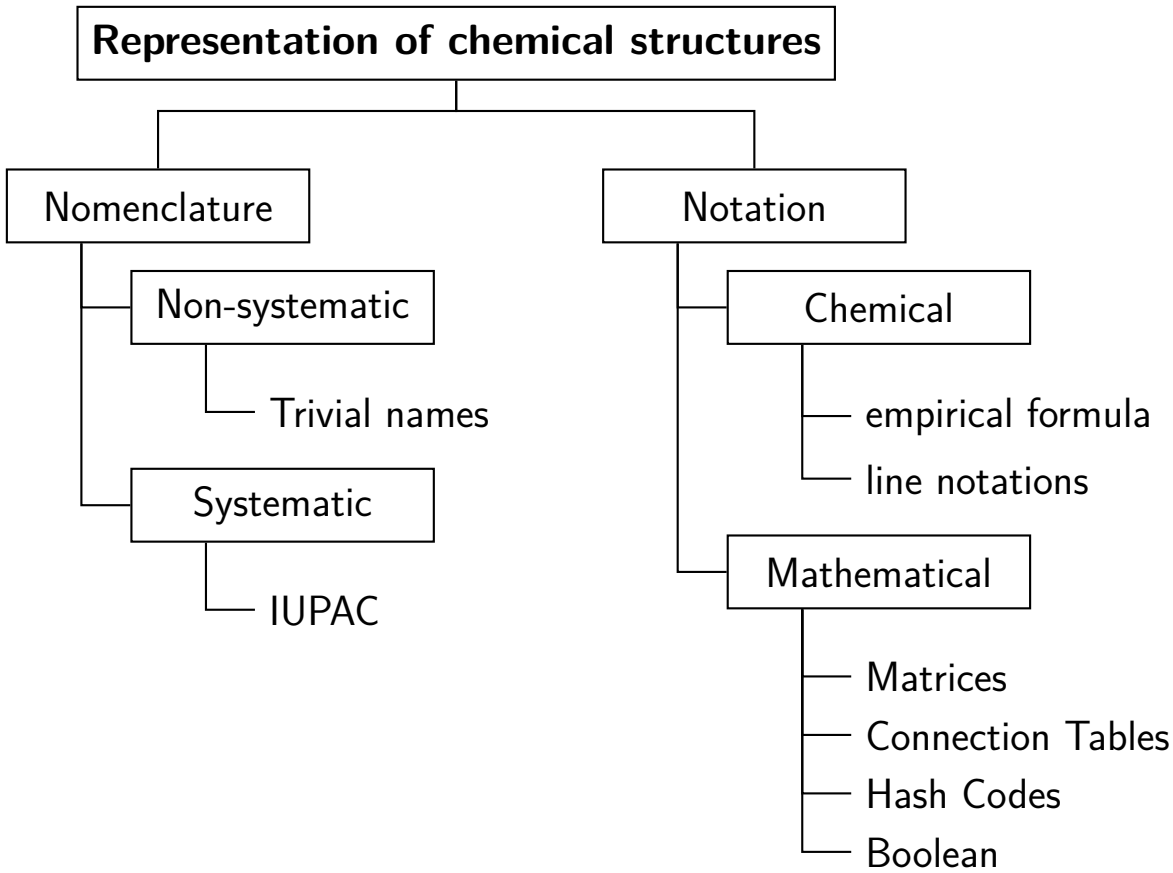
October 20, 2025

Digital Alchemy

Repetition Unit 01

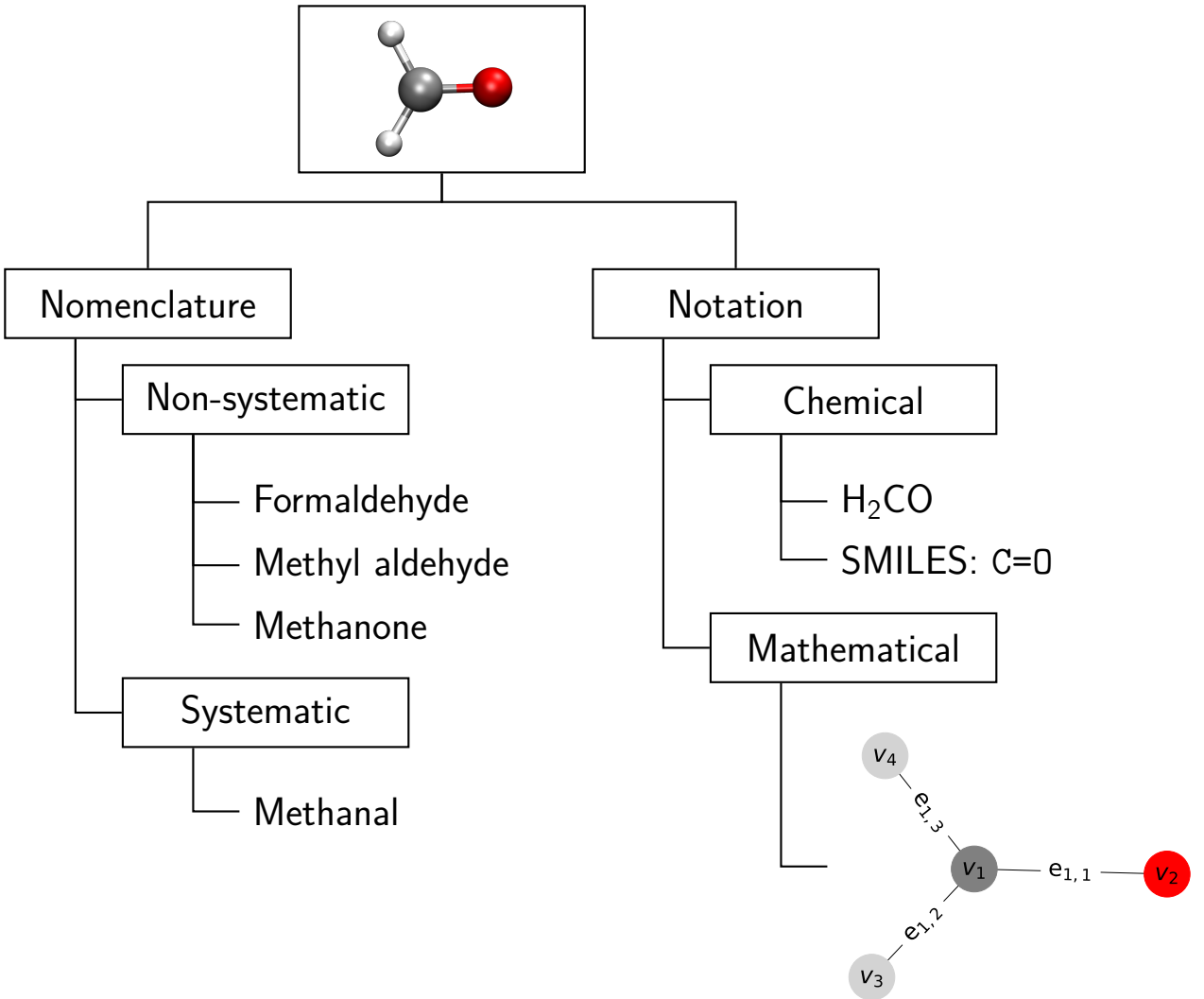
Molecular Representations

Introduction and Principles



Molecular Representations

Introduction and Principles



Chemical Nomenclature

Non-systematic Nomenclature

- *Alchymia*: first systematic textbook (1597)
- names do not reflect chemical composition or structure
- 3-letter, 1-letter system for amino acids

Systematic Nomenclature

Drawbacks of Chemical Nomenclature for Data Processing:

Chemical Nomenclature

Non-systematic Nomenclature

- *Alchymia*: first systematic textbook (1597)
- names do not reflect chemical composition or structure
- 3-letter, 1-letter system for amino acids

Systematic Nomenclature

- *IUPAC* Nomenclature
- unique names derived from chemical structure
- name derived from parts of compound structure (*e.g.* fragments like methyl, phenyl, amino, ...)

Drawbacks of Chemical Nomenclature for Data Processing:

Chemical Nomenclature

Non-systematic Nomenclature

- *Alchymia*: first systematic textbook (1597)
- names do not reflect chemical composition or structure
- 3-letter, 1-letter system for amino acids

Systematic Nomenclature

- *IUPAC* Nomenclature
- unique names derived from chemical structure
- name derived from parts of compound structure (*e.g.* fragments like methyl, phenyl, amino, ...)

Drawbacks of Chemical Nomenclature for Data Processing:

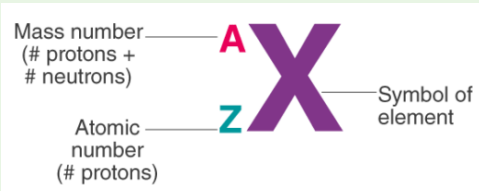
- lack of biuniqueness (name $\leftrightarrow$ structure)
- chemical structure not well defines (*e.g.* tautomers)
- length of names

Chemical Notations

a) Empirical Formula

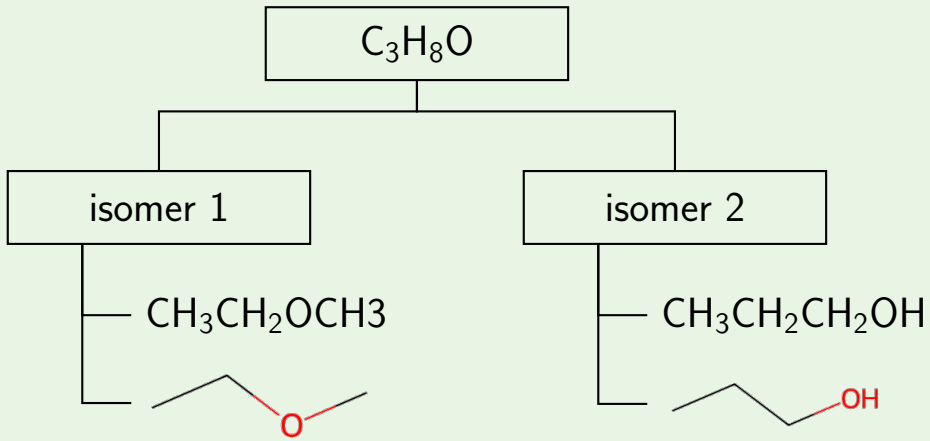
chemical elements

- element symbols (abbreviations)
- element characteristics (e.g., number of protons and electrons, oxidation numbers, number of atoms, charge, isotopes, ...)



(in)organic compounds

- Hill system: listing of elements and indication of stoichiometry by index
- condensed formulas: fragments in Hill notation



Drawbacks of Chemical Notations for Data Processing:

- no unique compound indexing (e.g. isomers have same empirical formula)
- chemical structure not well defined (*e.g.* constitution)

Chemical Notations

b) Line Notations

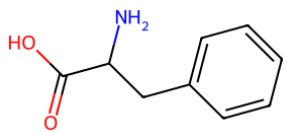
Line Notations

Line notations represent the structure of a chemical compound as linear sequence of letters, numbers, and symbols.

Examples:

- InChI, InChIKey
- SMILES, SMARTS, SELFIES

Trivial name	Phenylalanine
IUPAC Name	2-Amino-3phenylpropanoic acid



Empirical formula	$C_9H_{11}NO_2$
SMILES	NC(Cc1ccccc1)C(O)=O
InChI	InChI=1S/C9H11NO2/c10-8(9(11)12)6
	-7-4-2-1-3-5-7/h1-5,8H,6,10H2,(H,11,12)

Chemical Notations

SMILES

SMILES

SMILES (**S**implified **M**olecular **I**nput **L**ine **E**ntry **S**ystem) is a chemical notation that allows a user to represent a chemical structure in a way that can be used by the computer. **SMILES** is an easily learned and flexible notation based on rules.

Chemical Notations

SMILES

SMILES

SMILES (**S**implified **M**olecular **I**nput **L**ine **E**ntry **S**ystem) is a chemical notation that allows a user to represent a chemical structure in a way that can be used by the computer. **SMILES** is an easily learned and flexible notation based on rules.

Bonds:

- \- Single bond
- = Double bond
- # Triple bond
- * Aromatic bond
- . Disconnected Structures

Chemical Notations

SMILES

SMILES

SMILES (**S**implified **M**olecular **I**nput **L**ine **E**ntry **S**ystem) is a chemical notation that allows a user to represent a chemical structure in a way that can be used by the computer. **SMILES** is an easily learned and flexible notation based on rules.

Bonds:

-	Single bond
=	Double bond
#	Triple bond
*	Aromatic bond
.	Disconnected Structures

Simple Chains:

CC	CH_3CH_3
C=C	CH_2CH_2
CBr	CH_3Br
C#N	$C \equiv N$
Na.Cl	NaCl

Chemical Notations

SMILES

SMILES

SMILES (**S**implified **M**olecular **I**nput **L**ine **E**ntry **S**ystem) is a chemical notation that allows a user to represent a chemical structure in a way that can be used by the computer. **SMILES** is an easily learned and flexible notation based on rules.

Bonds:

-	Single bond
=	Double bond
#	Triple bond
*	Aromatic bond
.	Disconnected Structures

Simple Chains:

CC	CH_3CH_3
C=C	CH_2CH_2
CBr	CH_3Br
C#N	$C \equiv N$
Na.Cl	NaCl

Rings:

C=1CCCCC1	Cyclohexene
c1ccccc1	benzene
C1OC1CC	Ethyloyirane

Chemical Notations

SMILES

SMILES

SMILES (**S**implified **M**olecular **I**nput **L**ine **E**ntry **S**ystem) is a chemical notation that allows a user to represent a chemical structure in a way that can be used by the computer. **SMILES** is an easily learned and flexible notation based on rules.

Bonds:

-	Single bond
=	Double bond
#	Triple bond
*	Aromatic bond
.	Disconnected Structures

Simple Chains:

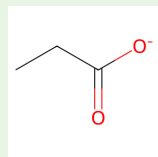
CC	CH_3CH_3
C=C	CH_2CH_2
CBr	CH_3Br
C#N	$C \equiv N$
Na.Cl	NaCl

Rings:

C=1CCCCC1	Cyclohexene
c1ccccc1	benzene
C1OC1CC	Ethyloyirane

Charged Atoms:

CCC(=O)[O-]



Chemical Notations

SMILES

SMILES

SMILES (**S**implified **M**olecular **I**nput **L**ine **E**ntry **S**ystem) is a chemical notation that allows a user to represent a chemical structure in a way that can be used by the computer. **SMILES** is an easily learned and flexible notation based on rules.

Bonds:

-	Single bond
=	Double bond
#	Triple bond
*	Aromatic bond
.	Disconnected Structures

Rings:

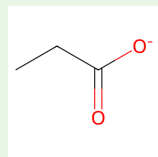
C=1CCCCC1	Cyclohexene
c1ccccc1	benzene
C1OC1CC	Ethyloyirane

Simple Chains:

CC	CH_3CH_3
C=C	CH_2CH_2
CBr	CH_3Br
C#N	$C \equiv N$
Na.Cl	NaCl

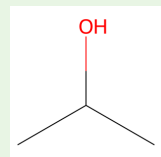
Charged Atoms:

CCC(=O)[O-]

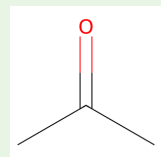


Branches:

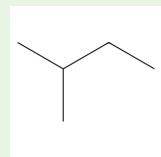
CC(O)C



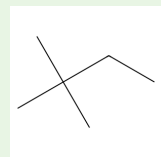
CC(=O)C



CC(CC)C

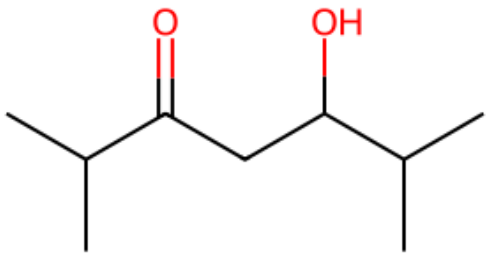


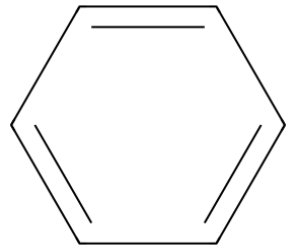
CC(C)(C)C



SMILES

SMILES (**S**implified **M**olecular **I**nput **L**ine **E**ntry **S**ystem) is a chemical notation that allows a user to represent a chemical structure in a way that can be used by the computer. **SMILES** is an easily learned and flexible notation based on rules.





InChI

The International Chemical Identifier

InChI

The IUPAC **In**ternational **Ch**emical **I**dentifier is a nonproprietary biunique identifier, namely a alphanumeric string, for chemical compounds released in 2000.

InChI

The International Chemical Identifier

InChI

The IUPAC **In**ternational **Ch**emical **I**dentifier is a nonproprietary biunique identifier, namely a alphanumeric string, for chemical compounds released in 2000.

Composition of InChI (`InChI=1S/`) in six hierarchical sublayers:

1. Main layer
   - chemical formula
   - connection sublayer `/c`
   - H-atom sublayer `/h`

InChI

The International Chemical Identifier

InChI

The IUPAC **In**ternational **Ch**emical **I**dentifier is a nonproprietary biunique identifier, namely a alphanumeric string, for chemical compounds released in 2000.

Composition of InChI (`InChI=1S/`) in six hierarchical sublayers:

1. Main layer
   - chemical formula
   - connection sublayer `/c`
   - H-atom sublayer `/h`
2. Charge layer
   - charge sublayer `/q`
   - proton sublayer `/p`

InChI

The International Chemical Identifier

InChI

The IUPAC **In**ternational **Ch**emical **I**dentifier is a nonproprietary biunique identifier, namely a alphanumeric string, for chemical compounds released in 2000.

Composition of InChI (`InChI=1S/`) in six hierarchical sublayers:

1. Main layer
   - chemical formula
   - connection sublayer `/c`
   - H-atom sublayer `/h`
2. Charge layer
   - charge sublayer `/q`
   - proton sublayer `/p`
3. Stereochemical layer
   - double bond sublayer `/b`
   - tetrahedral stereochemistry sublayers `/t, /m, /s`

InChI

The International Chemical Identifier

InChI

The IUPAC **In**ternational **Ch**emical **I**dentifier is a nonproprietary biunique identifier, namely a alphanumeric string, for chemical compounds released in 2000.

Composition of InChI (`InChI=1S/`) in six hierarchical sublayers:

1. Main layer
   - chemical formula
   - connection sublayer `/c`
   - H-atom sublayer `/h`
2. Charge layer
   - charge sublayer `/q`
   - proton sublayer `/p`
3. Stereochemical layer
   - double bond sublayer `/b`
   - tetrahedral stereochemistry sublayers `/t, /m, /s`
4. Isotopic layer
   - `/i, /h, /b, /t, /m, /s` sublayer

InChI

The International Chemical Identifier

InChI

The IUPAC **In**ternational **Ch**emical **I**dentifier is a nonproprietary biunique identifier, namely a alphanumeric string, for chemical compounds released in 2000.

Composition of InChI (`InChI=1S/`) in six hierarchical sublayers:

1. Main layer
   - chemical formula
   - connection sublayer `/c`
   - H-atom sublayer `/h`
2. Charge layer
   - charge sublayer `/q`
   - proton sublayer `/p`
3. Stereochemical layer
   - double bond sublayer `/b`
   - tetrahedral stereochemistry sublayers `/t, /m, /s`
4. Isotopic layer
   - `/i, /h, /b, /t, /m, /s` sublayer
5. Fixed-H layer (only nonstandard InChI)
6. Reconnected layers for metals (only nonstandard InChI)

InChI

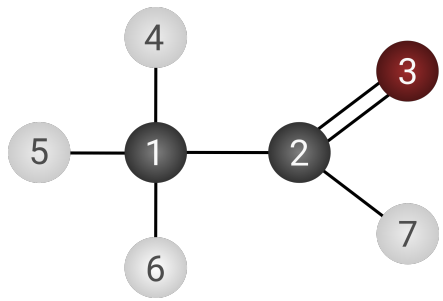
The International Chemical Identifier

InChI

The IUPAC **In**ternational **Ch**emical **I**dentifier is a nonproprietary biunique identifier, namely a alphanumeric string, for chemical compounds released in 2000.

Sublayers in InChI (`InChI=1S/`):

1. Main layer
   - chemical formula
   - connection sublayer `/c`
   - H-atom sublayer `/h`



`InChI = 1S`

Main layer

InChI

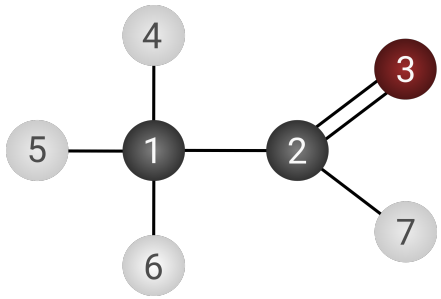
The International Chemical Identifier

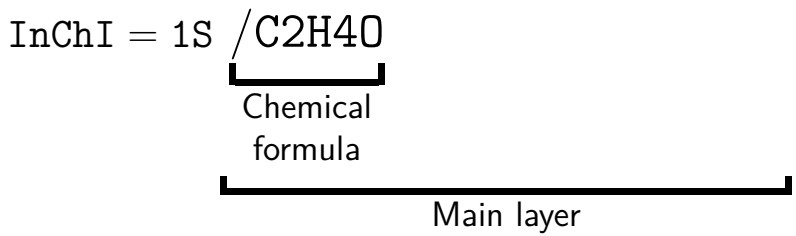
InChI

The IUPAC **In**ternational **Ch**emical **I**dentifier is a nonproprietary biunique identifier, namely a alphanumeric string, for chemical compounds released in 2000.

Sublayers in InChI (`InChI=1S/`):

1. Main layer
   - chemical formula
   - connection sublayer `/c`
   - H-atom sublayer `/h`





InChI

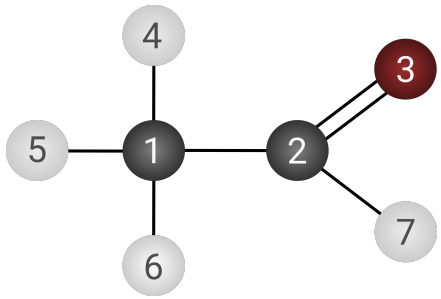
The International Chemical Identifier

InChI

The IUPAC **In**ternational **Ch**emical **I**dentifier is a nonproprietary biunique identifier, namely a alphanumeric string, for chemical compounds released in 2000.

Sublayers in InChI (`InChI=1S/`):

1. Main layer
   - chemical formula
   - connection sublayer `/c`
   - H-atom sublayer `/h`



$$\texttt{InChI} = \texttt{1S}\ \underbrace{\underbrace{/\texttt{C2H4O}}_{\text{Chemical formula}}\ \underbrace{/\texttt{c1-2-3}}_{\text{Connection sublayer}}}_{\text{Main layer}}$$

InChI

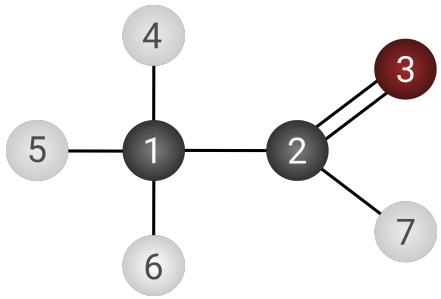
The International Chemical Identifier

InChI

The IUPAC **In**ternational **Ch**emical **I**dentifier is a nonproprietary biunique identifier, namely a alphanumeric string, for chemical compounds released in 2000.

Sublayers in InChI (`InChI=1S/`):

1. Main layer
   - chemical formula
   - connection sublayer `/c`
   - H-atom sublayer `/h`

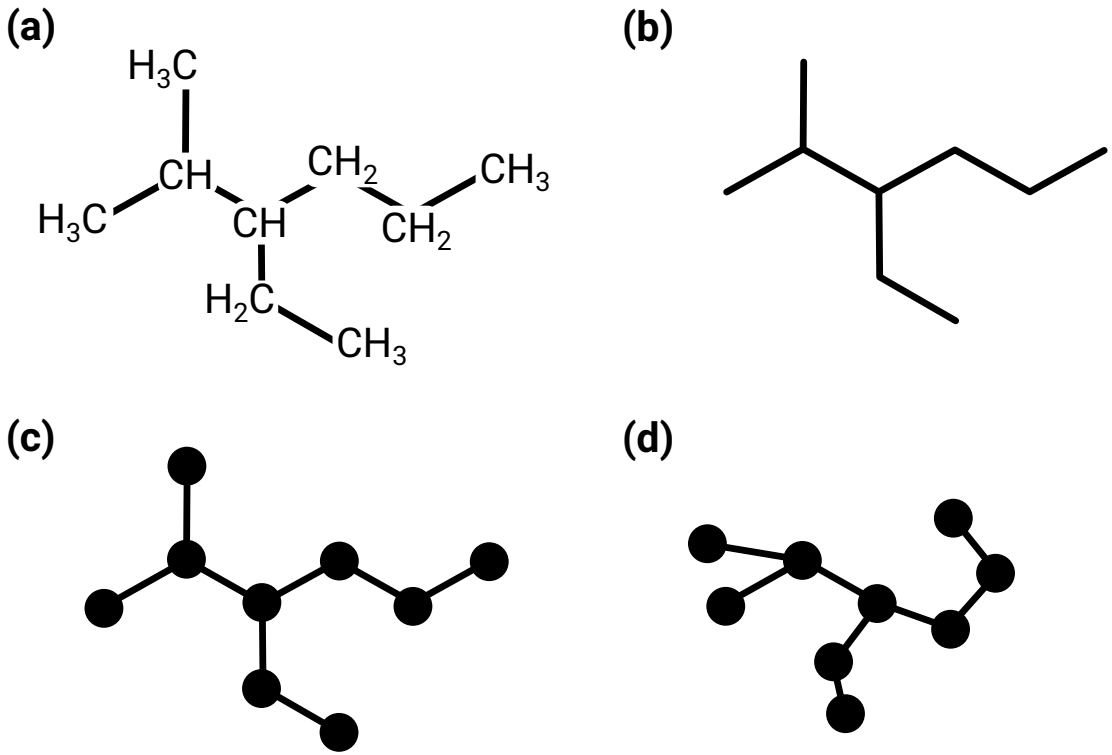


$$\texttt{InChI} = \texttt{1S}\ \underbrace{\underbrace{\texttt{/C2H4O}}_{\text{Chemical formula}}\ \underbrace{\texttt{/c1-2-3}}_{\text{Connection sublayer}}\ \underbrace{\texttt{/h2H,1H3}}_{\text{H atom sublayer}}}_{\text{Main layer}}$$

Chemical Notations

c) Mathematical Notations

Graph theory has particular significance for the representation of chemical structures. A structure diagram can be considered in mathematical terms as a graph where the nodes and the edges correspond to the atoms and the bonds, respectively.



Graphs (c) and (d) represent the same graphs, since graphs bear no 2D and 3D information.

Definitions in Graph Theory

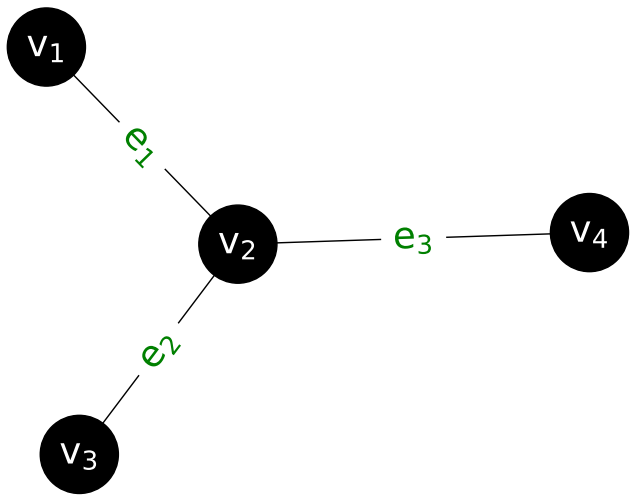
c) Mathematical Notations

Simple Graph

A **simple** graph is a mathematical object and consists of a an ordered pair (tuple) of sets of vertices (nodes) $V(G) = \{v_1, \ldots, v_n\}$ and sets of edges (lines) $E(G) = \{e_1, \ldots, e_m\}$, defined as $G(V, E)$.

The number of elements n in the vertex set $V(G) = \{v_1, v_2, \ldots, v_n\}$ gives the number of vertices in a graph G. Analogously, the number of elements m in the set $E(G)$ gives the number of edges in a graph.

Example:



Simple graph with four vertices $V = \{v_1, v_2, v_3, v_4\}$ and three edges $E = \{e_1, e_2, e_3\}$.

Definitions in Graph Theory

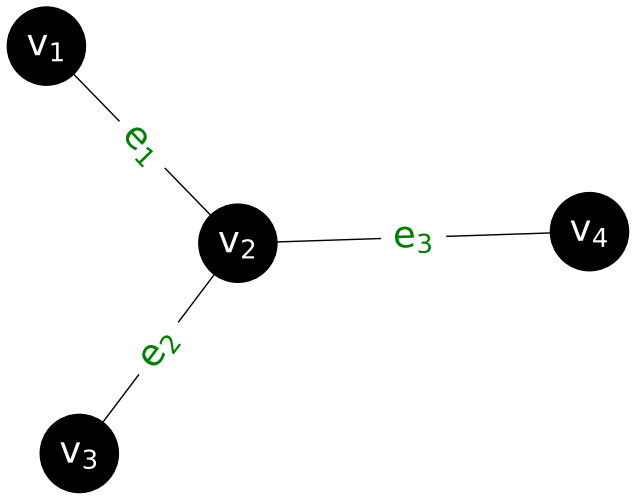
c) Mathematical Notations

Incidence of Vertices and Edges

Each edge E connects two vertices V, called the ends of this edge. An edge e with the ends in the two vertices v_i and v_j is denoted as $e_{i,j}$. We say that $e_{i,j}$ is **incident** into v_i and v_j.

*If a vertex is an endpoint of edge, we say they are **incident**.*

Example:



Definitions in Graph Theory

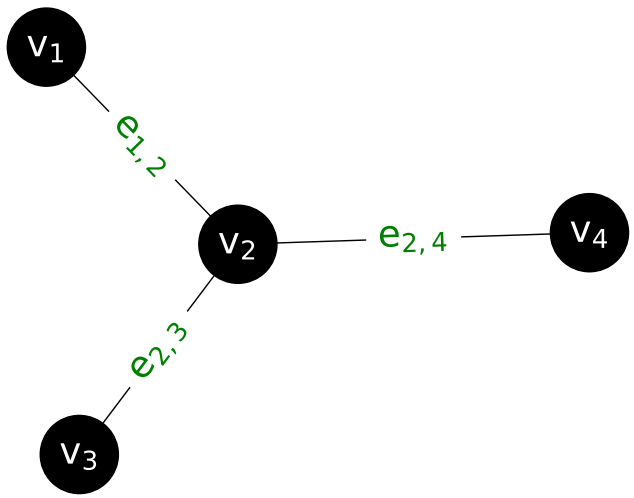
c) Mathematical Notations

Adjacent Vertices and Edges

Two vertices v_i and v_j are **adjacent** if they are the endpoints of an edge $e_{i,j}$, *i.e.*, if they are incident in a common edge $e_{i,j}$. Two distinct edges e_i and e_j are *adjacent* if they have at least one vertex $v_{i,j}$ in common.

*If two vertices in a graph are connected by an edge, we say the vertices are **adjacent**.*

Example:



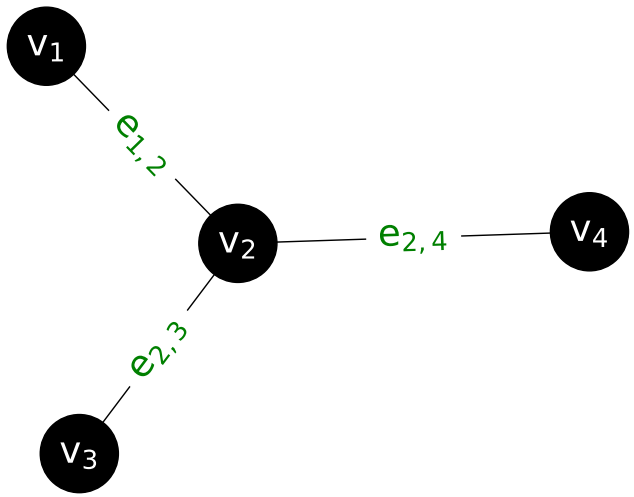
Definitions in Graph Theory

c) Mathematical Notations

Degree of a Vertex

The **degree** $d(v_i)$ of a vertex v_i is the number of edges incident with v_i.

Example:



Definitions in Graph Theory

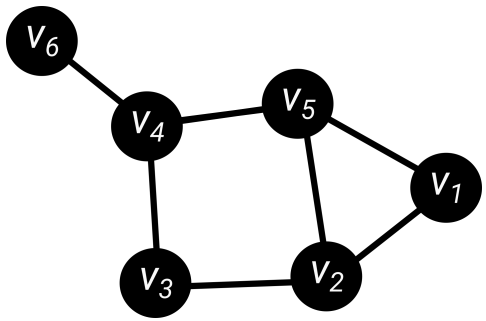
c) Mathematical Notations

Neighbourhood

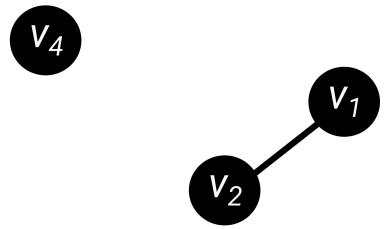
The set of all vertices *adjacent* to a specific vertex v_i is called the **(open) neighbourhood** of v_i and denoted by $N(v_i)$. The set $N(v_i)$, which includes v_i itself ($N(V_i) \cup \{v_i\}$) is called **closed neighbourhood**.

The **neighbourhood** of a vertex v_i is a graph G induced by all vertices adjacent to v_i, *i.e.*, the graph composed of the vertices adjacent to v_i and all edges connecting vertices adjacent to v_i.

Example: **Original Graph:**



Neighbourhood:



Definitions in Graph Theory

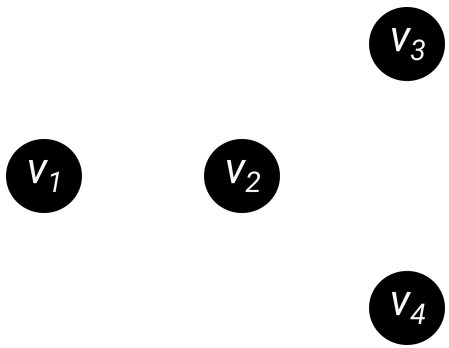
c) Mathematical Notations

Multigraph

A **multigraph** is characterized by two vertices that are linked by more than one edge (multiple bond).

*When a graph has more than one edge with the same endpoints it is called a **multigraph**.*

Example:



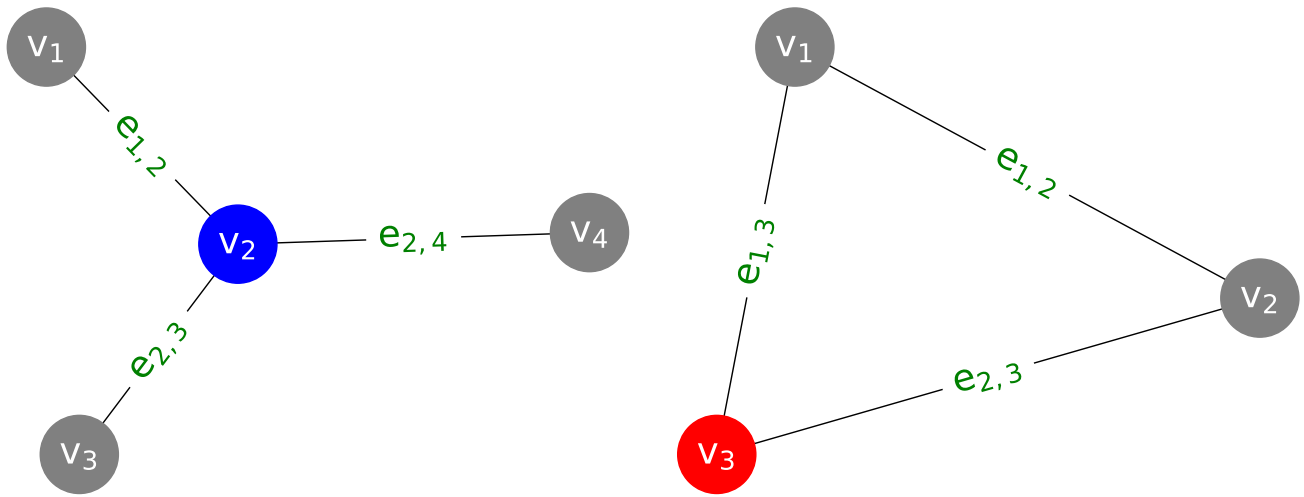
Definitions in Graph Theory

c) Mathematical Notations

Chromatic Number

We can assign labels to edges and vertices. In the case of coloring, the tuple $G = (V, E)$ is augmented with the property Color C to a triple $G = (V, E, C)$. The minimum number of colors needed to label the vertices/edges (so that *adjacent* objects receive different colors) is called **chromatic number**.

Example:



Definitions in Graph Theory

c) Mathematical Notations

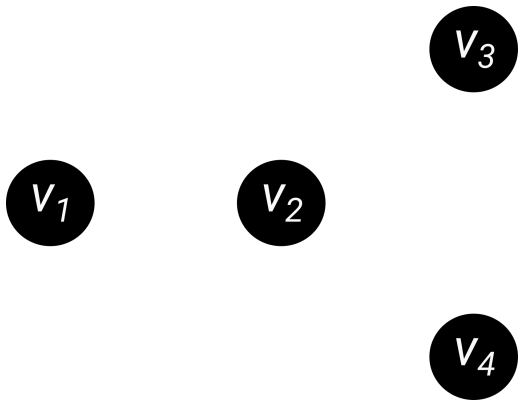
Undirected and Directed Graphs

In an **undirected** graph the sets of edges have symmetric lines between pairs of vertices.

A directed graph G_2 has no symmetric edges between pairs of vertices in the set of edges (*e.g.*, to represent polar bonds).

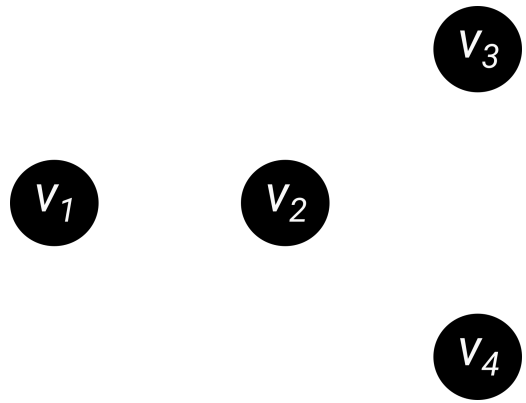
Example:

Undirected Graph ($G_1(V_1, E_1)$):



$V_1 = \{v_1, v_2, v_3, v_4\}$,
$E_1 = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_2, v_4\}\}$

Directed Graph ($G_2(V_2, E_2)$):



$V_2 = \{v_1, v_2, v_3, v_4\}$,
$E_2 = \{(v_1, v_2), (v_2, v_4), (v_3, v_2)\}$

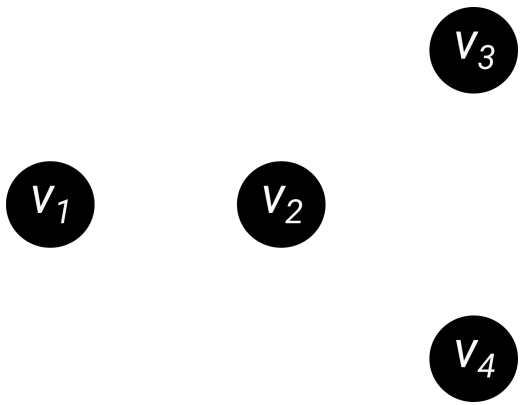
Definitions in Graph Theory

c) Mathematical Notations

Walk

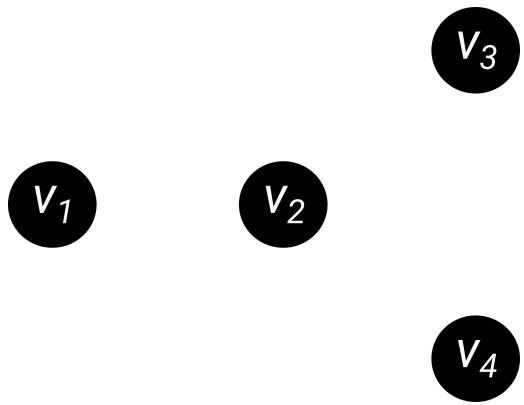
An alternating sequence of vertices and eges is called a walk, $w = (v_1, \ldots, v_n)$ with v_i and v_{i+1} being *adjacent* for $1 \leq i \leq n-1$. Each edge in a walk is *incident* with the vertices immediately preceding and succeeding it in the sequence.

Example: $w_1 = (v_1, v_2, v_3, v_4)$:



$V_1 = \{v_1, v_2, v_3, v_4\}$,
$E_1 = \{(v_1, v_2), (v_2, v_3), (v_3, v_4)\}$

$w_2 = (v_1, v_3, v_2, v_4)$:



$V_1 = \{v_1, v_2, v_3, v_4\}$,
$E_1 = \{(v_1, v_3), (v_3, v_2), (v_2, v_4)\}$

Definitions in Graph Theory

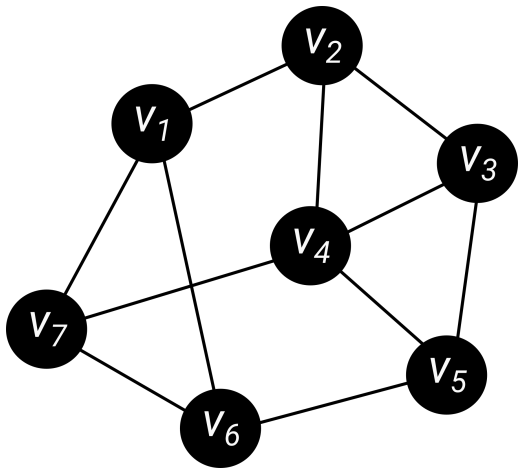
c) Mathematical Notations

Paths and Cycles

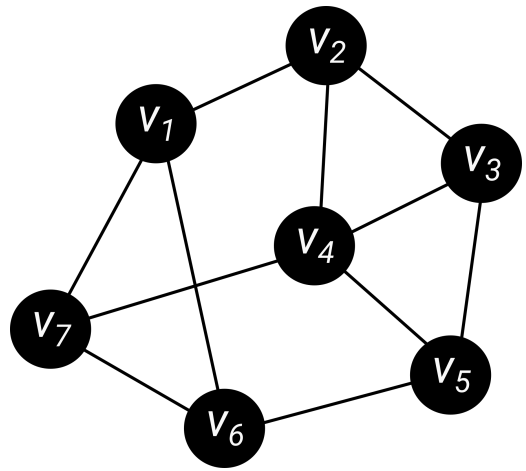
• A **path** is a graph P_n on vertices $\{v_1, v_2, \ldots, v_n\}$ with edges (v_i, v_{i+1}) for $1 \leq i \leq n-1$, and no other edges.

• A **cycle** is a graph C_n on vertices $\{v_1, v_2, \ldots, v_n\}$ with edges (v_i, v_{i+1}) for $1 \leq i \leq n$ and no other edges; this is a path in which the first and last vertices have been joined by an edge.

Example: **Paths** starting from v_1:



Cycles starting from v_1:



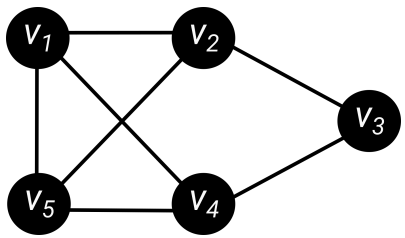
Definitions in Graph Theory

c) Mathematical Notations

Subgraphs and Supergraphs

A graph $G_2 = (V_2, E_2)$ is a **subgraph** of $G_1 = (V_1, E_1)$, if V_2 is a subset of V_1 $(V_2 \subseteq V_1)$ and E_2 is a subset of E_1 $(E_2 \subseteq E_1)$. If G_2 is a **subgraph** of G_1 it is also called G_2 is isomorphic to a subgraph of G_1. *Vice versa*, G_1 is a **supergraph** of G_2, if G_2 is a subgraph of G_1.

Example: **Supergraph** G:



Subgraphs (H_1, H_2, and H_3):

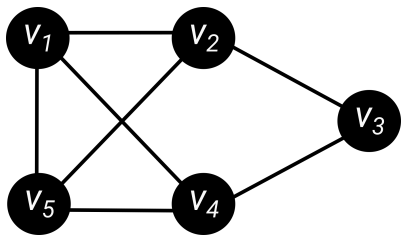
Definitions in Graph Theory

c) Mathematical Notations

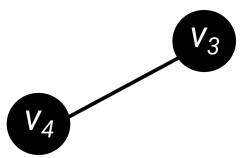
Connectivity

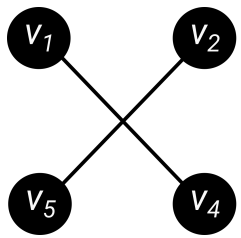
In a unidirected graph G, two vertices v_i and v_j are called connected if G contains a path from v_i to v_j. Otherwise, they are called isconnected. If the two vertices are connected by a path of length 1 (they are endpoints of a single edge), the vertices are called adjacent. A graph with just one vertex is connected. An edgeless graph with two or more vertices is disconnected.

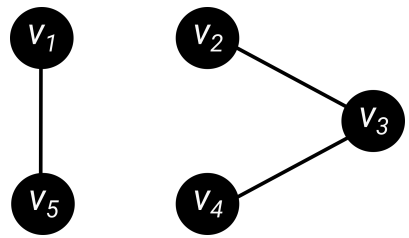
Example: **Supergraph** G:



Subgraphs (H_1, H_2, and H_3):







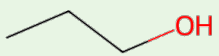
Molecules as Graphs

c) Mathematical Notations

- **Molecules as Graphs:**
  - Vertices: Atoms
  - Edges: Bonds
  - additional properties (atom-type, charge, ...)
- `RDKit` **Library:**
  - `RDKit`: powerful, open-source cheminformatics Python toolkit
  - functionalities for working with chemical structures and data

`RDKit` demo for representing molecules as graphs:

```
1 from rdkit import Chem
2 from rdkit.Chem import AllChem
3
4 mol = Chem.MolFromSmiles('CCCO')
5 mol = Chem.AddHs(mol)
6 AllChem.EmbedMolecule(mol, AllChem.ETKDG())
7 >
```



Representation of Molecules

Molecular Descriptors

Molecular Descriptors

A molecular representation encodes chemical identity of a molecular entity. Only after the chemical identity is converted into a **descriptor** (an array of numbers), computer can efficiently process a large number of structures.

Requirements for Descriptors:

- physicochemical property coverage

Representation of Molecules

Molecular Descriptors

Molecular Descriptors

A molecular representation encodes chemical identity of a molecular entity. Only after the chemical identity is converted into a **descriptor** (an array of numbers), computer can efficiently process a large number of structures.

Requirements for Descriptors:

- physicochemical property coverage
- information richness

Representation of Molecules

Molecular Descriptors

Molecular Descriptors

A molecular representation encodes chemical identity of a molecular entity. Only after the chemical identity is converted into a **descriptor** (an array of numbers), computer can efficiently process a large number of structures.

Requirements for Descriptors:

- physicochemical property coverage
- information richness
- respect symmetries

Representation of Molecules

Molecular Descriptors

Molecular Descriptors

A molecular representation encodes chemical identity of a molecular entity. Only after the chemical identity is converted into a **descriptor** (an array of numbers), computer can efficiently process a large number of structures.

Requirements for Descriptors:

- physicochemical property coverage
- information richness
- respect symmetries
- transferability

Representation of Molecules

Molecular Descriptors

Molecular Descriptors

A molecular representation encodes chemical identity of a molecular entity. Only after the chemical identity is converted into a **descriptor** (an array of numbers), computer can efficiently process a large number of structures.

Requirements for Descriptors:

- physicochemical property coverage
- information richness
- respect symmetries
- transferability
- scalability

Representation of Molecules

Molecular Descriptors

Molecular Descriptors

A molecular representation encodes chemical identity of a molecular entity. Only after the chemical identity is converted into a **descriptor** (an array of numbers), computer can efficiently process a large number of structures.

Requirements for Descriptors:

- physicochemical property coverage
- information richness
- respect symmetries
- transferability
- scalability
- dimensionality reduction

Representation of Molecules

Molecular Descriptors

Molecular Descriptors

A molecular representation encodes chemical identity of a molecular entity. Only after the chemical identity is converted into a **descriptor** (an array of numbers), computer can efficiently process a large number of structures.

Requirements for Descriptors:

- physicochemical property coverage
- information richness
- respect symmetries
- transferability
- scalability
- dimensionality reduction

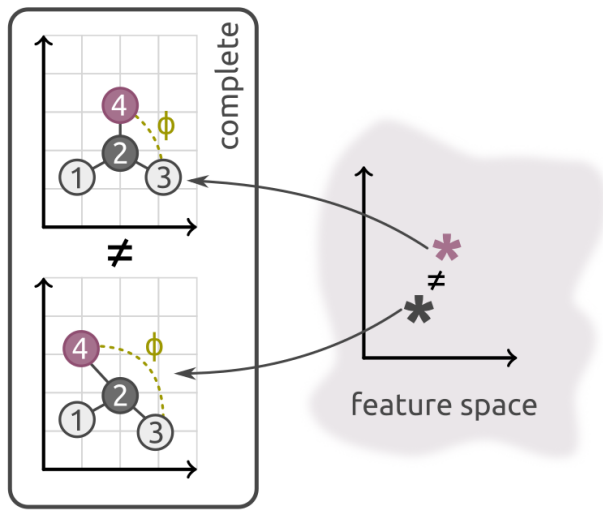
Graph/Matrix Descriptors

- tabular expression of any expression
- well-know matrix operations apply
- capture **invariances** of physical systems
- encode structural information without introducing additional descriptors

Requirements for Effective Structural Representations

Structure representations have to be ...

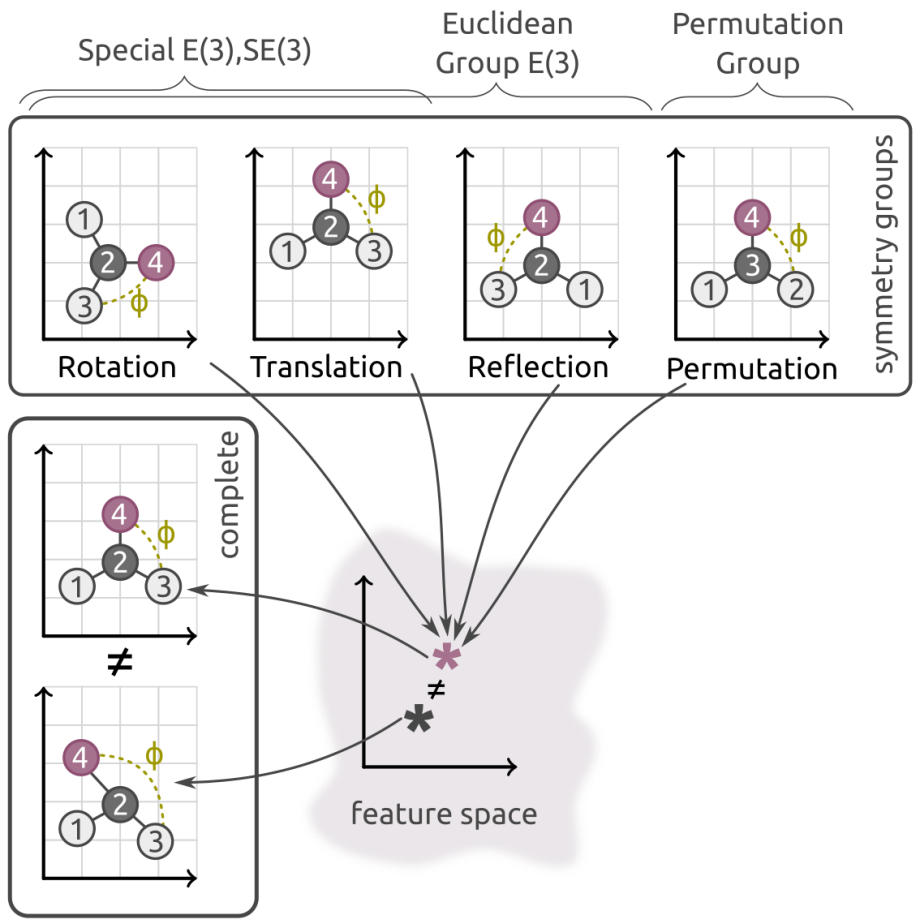
- **complete:** Inequivalent structures should be mapped to distinct features.



Requirements for Effective Structural Representations

Structure representations have to be ...

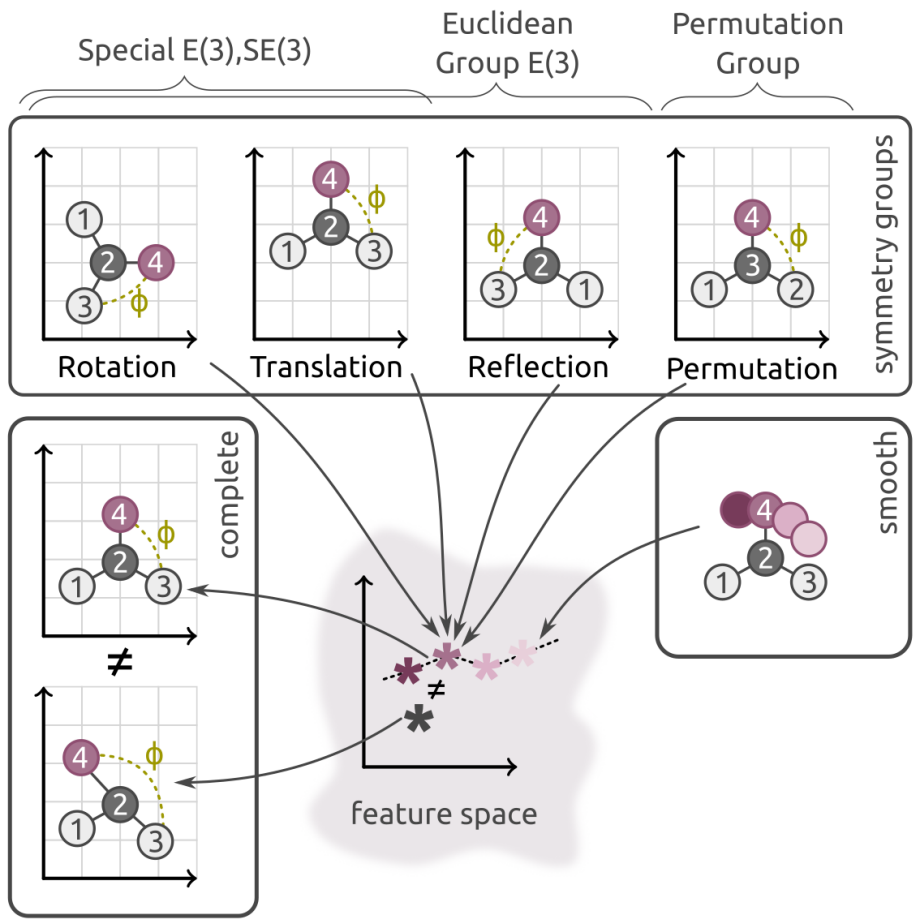
- **complete:** Inequivalent structures should be mapped to distinct features.
- **symmetry-respecting:** Equivalent structures should be mapped to the same features, obeying fundamental physical symmetries.



Requirements for Effective Structural Representations

Structure representations have to be ...

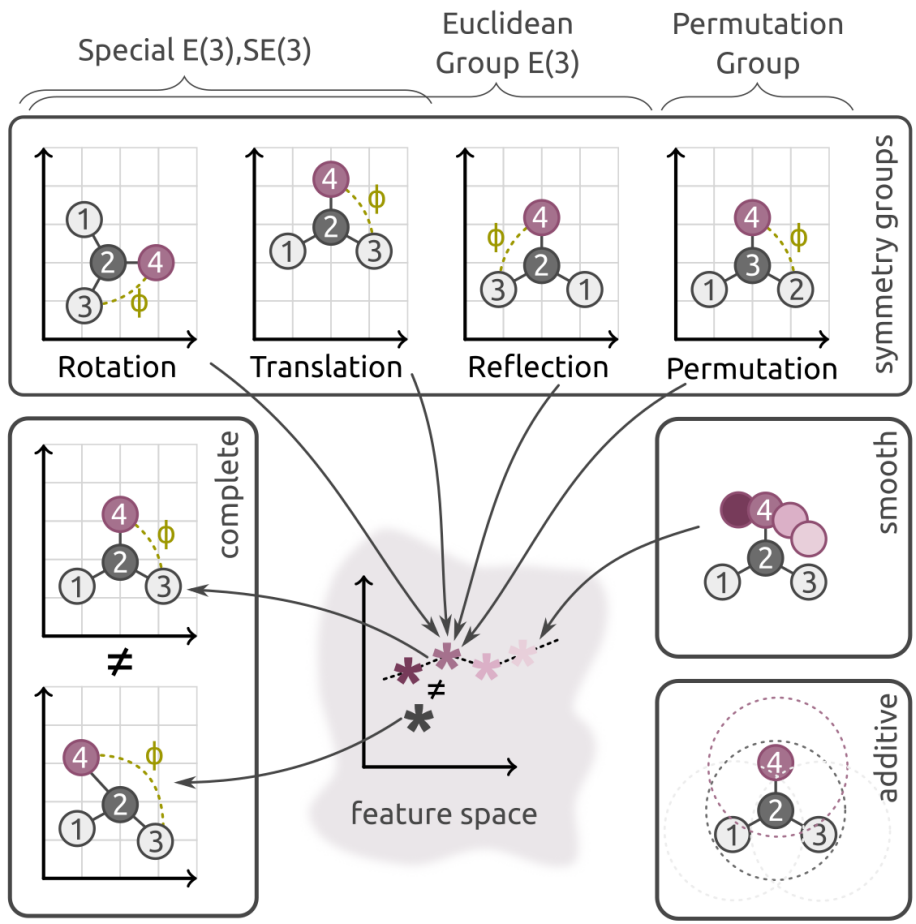
- **complete:** Inequivalent structures should be mapped to distinct features.
- **symmetry-respecting:** Equivalent structures should be mapped to the same features, obeying fundamental physical symmetries.
- **smooth:** Continuous deformations of a structure should result in smooth deformations of the associated features.



Requirements for Effective Structural Representations

Structure representations have to be ...

- **complete:** Inequivalent structures should be mapped to distinct features.
- **symmetry-respecting:** Equivalent structures should be mapped to the same features, obeying fundamental physical symmetries.
- **smooth:** Continuous deformations of a structure should result in smooth deformations of the associated features.
- **additiv:** For heterogeneous datasets (different molecular sizes), the representation should be decomposable into a sum of local environments (usually atom-centered), ensuring transferability and extensivity of predictions.



Matrix Representations

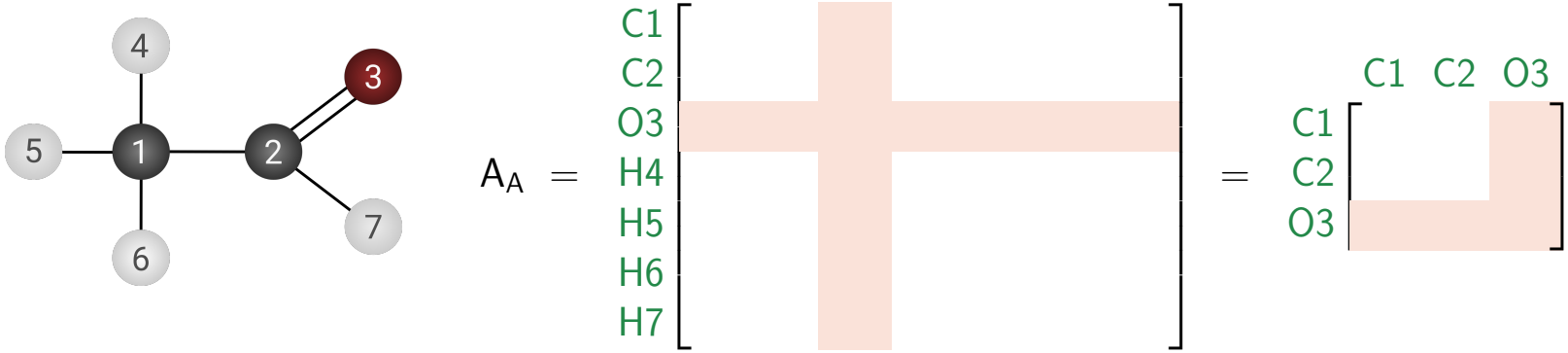
Adjacency Matrix

Adjacency Matrix (A_A)

The **adjacency** matrix of an undirected graph $G(V, E)$ with n labelled vertices (atoms) is a symmetric, $(n \times n)$-matrix with the entries $a_{i,j}$ representing the connectivity of all atoms.

$$a_{i,j} = \begin{cases} 1 & \forall\, i \neq j \wedge e_{i,j} \in E(G) \\ 0 & \forall\, i \neq j \vee e_{i,j} \notin E(G) \end{cases}$$

Thus, the matrix entry $a_{i,j}$ obtains a value if the nodes v_i and v_j are adjacent, and zero otherwise.



- **Note:** A_A has no information on bond orders, free electrons, or stereochemistry

Matrix Representations

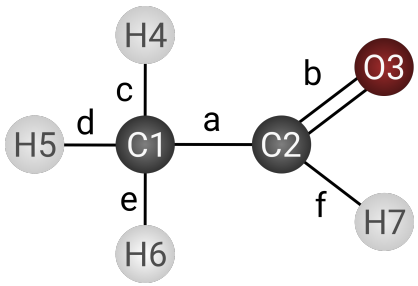
Incidence Matrix

Incidence Matrix (A_I)

The **incidence** matrix of an undirected Graph $G(V, E)$ is an $n \times m$ matrix showing the relationship between nodes (n, columns, atoms) and edges (m, rows, bonds).

$$a_{i,j} = \begin{cases} 1 & \forall\ v_i \in e_j (\text{undirected}) \vee e_j = (v_i, x) (\text{directed}) \\ 0 & \forall\ v_i \notin e_j \end{cases}$$

A matrix entry $a_{i,j}$ is set to a value of 1, if any edge e_j leaves or enters vertex v_i (incidence), respectively or zero if v_i and e_j are unrelated to each other.



$A_I =$

	C1	C2	O3	H4	H5	H6	H7
a							
b							
c							
d							
e							
f							

Matrix Representations

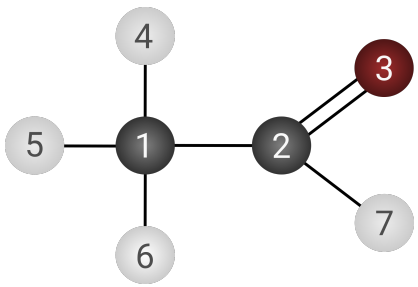
Distance Matrix

Distance Matrix (A_D)

The **distance** matrix A_D of a graph G with n vertices is an $n \times n$ matrix, where the matrix elements a_{ij} are the distances between the vertices v_i and v_j in the graph, *i.e.*, the number of edges on the shortest path (N_e) between the two vertices involved.

$$a_{i,j} = \begin{cases} N_e & \forall\, i \neq j \\ 0 & \forall\, i = j \end{cases}$$

Distances can be either expressed as geometric distances (*e.g.* in pm) or as topological distances (in number of bonds).



A_D =

	C1	C2	O3	H4	H5	H6	H7
C1							
C2							
O3							
H4							
H5							
H6							
H7							

(**topological**)

Representation of Molecules

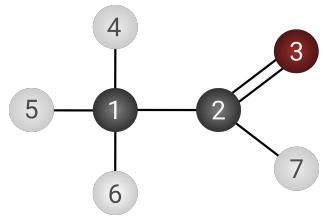
Distance Matrix

Distance Matrix

The **distance** matrix A_D of a graph G with n vertices is an $n \times n$ matrix, where the matrix elements a_{ij} are the distances between the vertices v_i and v_j in the graph, *i.e.*, the number of edges on the shortest path between the two vertices involved.

Requirements for Descriptors:

	A_D
• physicochemical property coverage	
• **translation**al invariant	
• **rotation**al invariant	
• **permutation** invariant	
• transferability	
• discriminate structures	
• scalability	
• dimensionality reduction	



	C1	C2	O3	H4	H5	H6	H7
C1	0	1	2	1	1	1	2
C2	1	0	1	2	2	2	1
O3	2	1	0	3	3	3	2
H4	1	2	3	0	2	2	3
H5	1	2	3	2	0	2	3
H6	1	2	3	2	2	0	3
H7	2	1	2	3	3	3	0

Matrix Representations

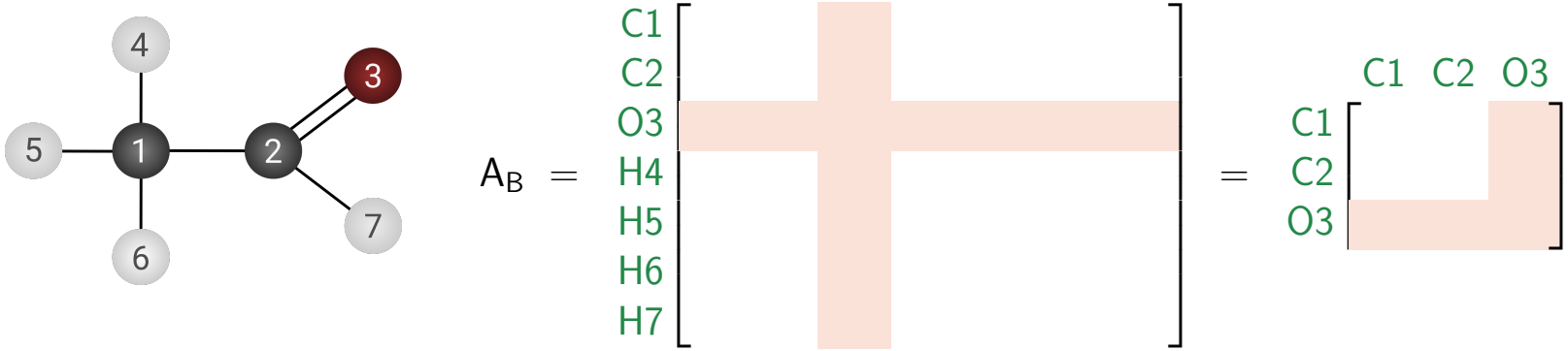
Bond Matrix

Bond Matrix (A_B)

The bond matrix is related to the adjacency matrix but additionally provides information on bond order of connected atoms. Elements of the matrix are set to the value of the bond order, *i.e.*,

```
1  BO = {'no bond': 0, 'single': 1, 'double': 2, 'triple': 3}
```

For symmetry reasons this matrix type also contains redundant information.



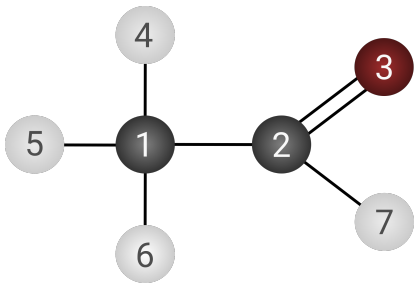
Matrix Representations

Bond Electron Matrix

Bond Electron Matrix (A_{BE})

In addition to the bond matrix, the bond-electron matrix lists the number of free valence electrons of an atom as its corresponding diagonal element, and thus reflects the valence electrons of all atoms in a molecule forming a chemical bond (off-diagonal) or associated with free electrons of atoms (diagonal).

- valence electrons of atom i: $s_i = \sum_j b_{ji} = \sum_j b_{ji}$
- total number of valence electrons: $S = \sum_i \sum_j b_{ij}$
- cross sum: octet rule



$A_{BE} =$

	C1	C2	O3	H4	H5	H6	H7	Row sum
C1								
C2								
O3								
H4								
H5								
H6								
H7								
Column sum								(# valence e^-)

Representation of Molecules

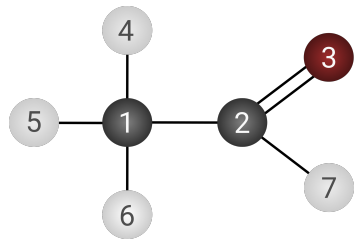
Bond-Electron Matrix

Bond-Electron Matrix

In addition to the bond matrix, the bond-electron matrix lists the number of free valence electrons of an atom as its corresponding diagonal element, and thus reflects the valence electrons of all atoms in a molecule forming a chemical bond (off-diagonal) or associated with free electrons of atoms (diagonal).

Requirements for Descriptors:

	A_{BE}
• physicochemical property coverage	
• **translation**al invariant	
• **rotation**al invariant	
• **permutation** invariant	
• transferability	
• discriminate structures	
• scalability	
• dimensionality reduction	



	C1	C2	O3	H4	H5	H6	H7	
C1	0	1	0	1	1	1	0	4
C2	1	0	2	0	0	0	1	4
O3	0	2	4	0	0	0	0	6
H4	1	0	0	0	0	0	0	1
H5	1	0	0	0	0	0	0	1
H6	1	0	0	0	0	0	0	1
H7	0	1	0	0	0	0	0	1
	4	4	6	1	1	1	1	36

Representation of Molecules

Molecular Descriptors

Molecular Descriptors

A molecular representation encodes chemical identity of a molecular entity. Only after the chemical identity is converted into a **descriptor** (an array of numbers), computer can efficiently process a large number of structures.

Requirements for Descriptors:

	A_A	A_D	A_I	A_B	A_{BE}
• physicochemical property coverage		(✓)			✓
• **translation**al invariant		✓			✓
• **rotation**al invariant		✓			✓
• **permutation** invariant		✗			✗
• transferability		✓			(✓)
• discriminate structures		(✓)			(✓)
• scalability		✓			✓
• dimensionality reduction		✓			✓

Representation of Molecules

Molecular Descriptors

Molecular Descriptors

A molecular representation encodes chemical identity of a molecular entity. Only after the chemical identity is converted into a **descriptor** (an array of numbers), computer can efficiently process a large number of structures.

Requirements for Descriptors:

- physicochemical property coverage
- information richness
- invariance
- transferability
- scalability
- dimensionality reduction

Discussed Representations

- **Chemical**:
  - empirical formula
  - line notations: InChi, SMILES
- **Graphs** (Mathematical):
  - Adjacency Matrix (A_A)
  - Distance Matrix (A_D)
  - Incidence Matrix (A_I)
  - Bond, Bond-Electron Matrix (A_B, A_{BE})

Representation of Molecules

Outlook

Physics-inspired molecular structure descriptors:

- **Coulomb Matrix**
- **Bag-of-Bonds (BoB)**
- **Bonds-in-molecules (BIM)**
- **Bonds-Angles-Dihedrals (BAD)**
- **Atom-centered symmetry functions (ACSFs)**
- **Fingerprints**
- Smooth overlap of atomic positions (SOAP)
- Gaussian Potential (GAP)
- ...