dalke scientific
More science. Less time.

Python Libraries for Computational Chemistry and Biology

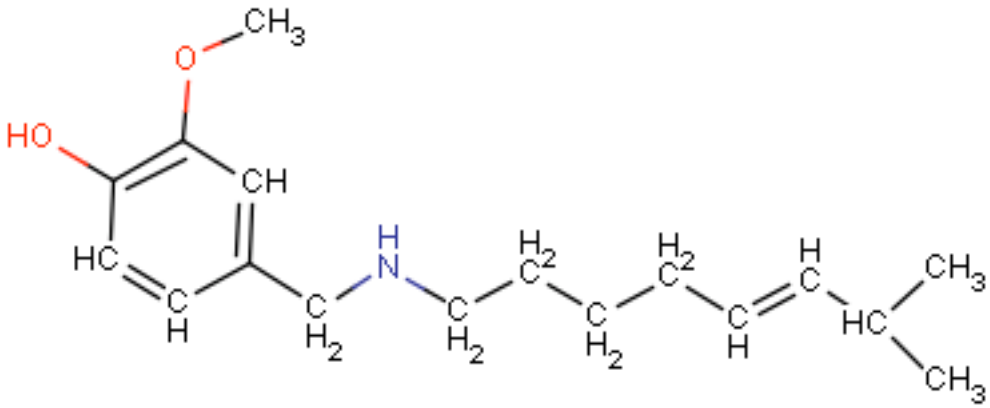
Andrew Dalke
Dalke Scientific Software, LLC
www.dalkescientific.com

My Background

- Started doing molecular mechanics in 1992
- Also worked on structure visualization, bioinformatics, and chemical informatics
- Now a consultant.
- Develop tools for researchers to get more science done in less time
- First saw Python in 1995, full-time in 1998
- Why Python? Both (computational) scientists and software developers like it.

New Mexican Chilies

Capsaicin



(Drawn in ChemAxon's MarvinSketch - Java)

- Benzene ring in the head
- Large, so you don't smell it
- Long hydrocarbon tail - dissolves well in oil, not water

How do we find more data about capsaicin?

Could search for the name “capsaicin”.

Assuming everyone uses the same name...

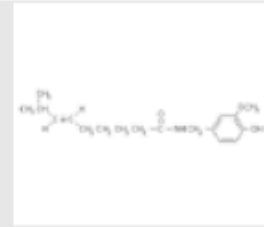
Or that someone compiles a list of all aliases.

Sigma-Aldrich Co.

Product No.	Description
M2028	**Capsaicin** min. 95 % from Capsicum

Identifiers

Synonyms	8-Methyl-N-vanillyl-trans-6-nonenamide
Molecular Formula	$C_{18}H_{27}NO_3$
Molecular Weight	305.41
CAS Number	404-86-4
Beilstein Registry Number	2816484
EG/EC Number	2069698
MDL number	MFCD00017259



Enlarge

Description

Biochem/physiol Actions	Prototype vanilloid receptor agonist. Neurotoxin; activates sensory neurons that give rise to unmyelinated C-fibers, many of which contain substance P. Topical application desensitizes the sensory nerve endings giving a paradoxical antinociceptive effect; systemic administration can be neurotoxic to capsaicin-sensitive cells, especially in newborn animals.
ID Clarifier	Active component of cayenne pepper

Properties

Solubility	
ethanol	soluble
water	insoluble
Storage temp.	2-8°C

References

Literature	Holzer, P., Local effector functions of capsaicin-sensitive sensory nerve endings: involvement of tachykinins, calcitonin gene-related peptide and other neuropeptides *Neuroscience* **24**, 739-768 (1988)
	Bevan, S., and Szolcsanyi, J., Sensory neuron-specific actions of capsaicin: mechanisms and applications *Trends Pharmacol. Sci.* **11**, 330-333 (1990)
Merck	*Merck***13**,1774

But...

But how does that compiler know that the different names refer to the same thing?

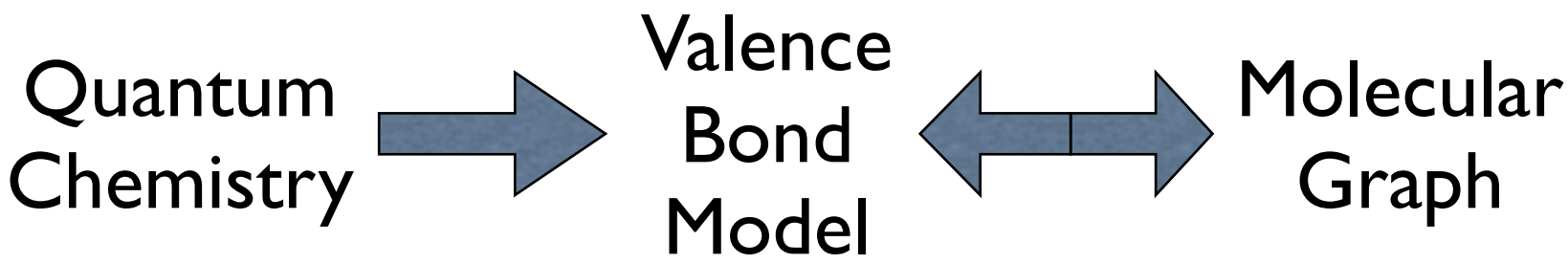
What if you isolate a compound and want know if others have reported it before?

What about finding information about compounds similar to capsaicin?

Chemical Informatics

Use *chemically-based* approaches (the valence bond model) to store and search chemical data.

"valence bond model" means a molecule has atoms and bonds between two atoms. It's only a (very useful) approximation.



Some graph searches

- Convert the structure to its "canonical" (unique) name then do a string lookup
- Search for a specific subgraph ("compounds with a benzene ring") or the largest common subgraph ("maximum common substructure")
- Index based on chemical features ("has more than 3 oxygens, has halogen, has fused rings...". The bit pattern is called a *fingerprint*).
- Find the nearest match in fingerprint space

PyDaylight

Daylight sells C/Fortran libraries for chemical informatics.

PyDaylight is a "thick" wrapper to make it Pythonic.

- real objects, with attributes
- iterators (old-style)
- hooks into Python's garbage collection
- errors raise exceptions

My company supports PyDaylight, under the LGPL.

PyDaylight example

```
>>> from daylight import Smiles, Smarts
>>> mol = Smiles.smilin("COC(C=1)=C(O)C=CC1CNCCCCC=CC(C)C")
>>> mol.cansmiles()
'c1(c(ccc(c1)CNCCCCC=CC(C)C)O)OC'
>>> print len(mol.atoms), "atoms", len(mol.bonds), "bonds", len(mol.cycles), "cycles"
20 atoms 20 bonds 1 cycles
>>> mol.atoms[2].symbol, mol.atoms[2].aromatic
('C', 1)
>>> len(mol.atoms[2].bonds)
3
>>> [(bond.bondtype, bond.bondorder) for bond in mol.atoms[2].bonds]
[(1, 1), (4, 1), (4, 2)]
>>> pat = Smarts.compile("[!#6][#6]")
>>> for m in pat.match(mol):
...   print m.atoms[0].symbol, m.bonds[0].symbol, m.atoms[1].symbol
...
O - C
O - C
O - C
N - C
N - C
>>>
```

Frowns

Actually, that wasn't PyDaylight. It was "Frowns",
a free (BSD license) reimplementation of part of PyDaylight.

Written by Brian Kelley - frowns.sourceforge.net

Supports SMILES, structure perception, canonicalization,
SMARTS searches, fingerprints.

Designed for flexibility and correctness, not for speed.
Not yet robust. Fails with some aromatic nitrogens.
Chirality not quite correct.

Thor lookup searches

Find all known aliases for “capsaicin”

```
import sys
from daylight import Thor

db = Thor.open_fullname("medchem04@green")

NAM = db.get_datatype("$NAM")

entry_list = db.xrefget_tdts(NAM, "capsaicin")
if entry_list:
    for entry in entry_list:
        for datatree in [entry] + entry.datatrees:
            for dataitem in datatree.dataitems:
                if dataitem.datatype == NAM:
                    print dataitem.datafields[0].stringvalue,
    print
```

Merlin similarity searches

```
import sys, string
from daylight import Merlin, Grid, Task
from daylight import DX_TAG_SIMILARITY, DX_FUNC_SHORTEST

pool = Merlin.open_fullname("medchem99@green")
hitlist = pool.hitlist()

similar_col = pool.typename_column(DX_TAG_SIMILARITY)
pcn_col = pool.typename_column("PCN", func = DX_FUNC_SHORTEST)

grid = Grid.Grid(hitlist, [similar_col, pcn_col])
hitlist.zapna(pcn_col)

smiles = "c1(c(ccc(c1)CNCCCCC=CC(C)C)O)OC"
task = hitlist.task(similar_col.similar.tanimoto(smiles, 0.0),
                            Merlin.NEW_LIST)
result = task.notify(Task.TextStatusBar())
if not result:
    print "   No similar structures found."
else:
    hitlist.sort(similar_col.sort.default_sort, 0)
    print "\nNo. Similarity  Shortest available LOCAL NAME"
    print "---", "-"*10, "-"*30
    for i in range(10):
        print "%2d. %-10s %s" % (i+1, grid[i, 0], grid[i, 2])
    print "---", "-"*10, "-"*30, "-"*30, "\n"
```

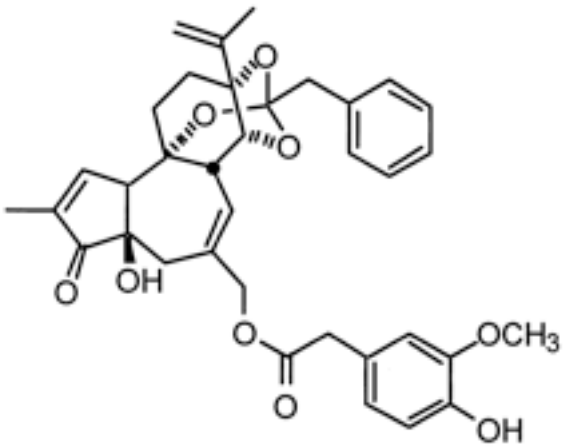
Uggh! Very Daylight specific.

Nowadays, everyone with $50,000 or more to spare is switching to an Oracle cartridge and using SQL.

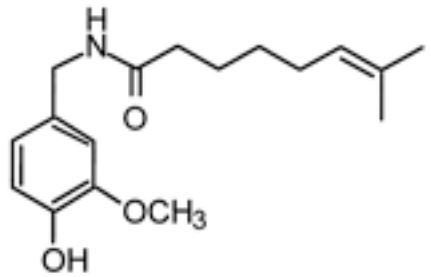
Still vendor specific, but much easier to use.

Don’t need a special-purpose API - DB-API is just fine.

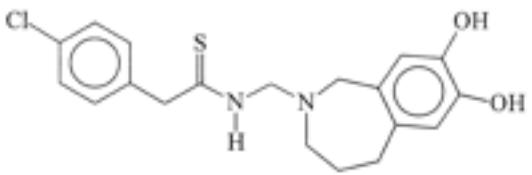
Capsaicin is a vanilloid



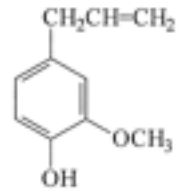
Resiniferatoxin



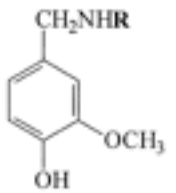
Capsaicin



Capsazepine



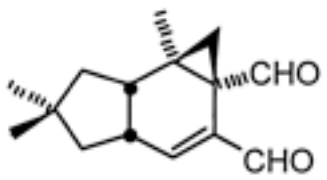
Eugenol



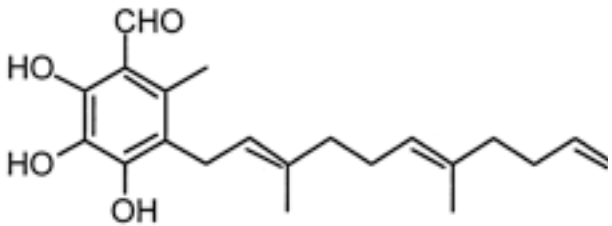
Capsaicin, **R**: $CO(CH_2)_7CH_3$

Olvanil, **R**: $CO(CH_2)_7CH{=}CH(CH_2)_7CH_3$

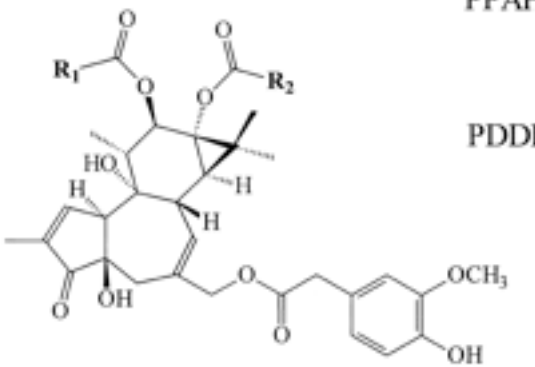
Compound 57, **R**: $CSNH(CH_2)_7CH_3$

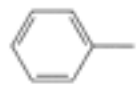


Isovelleral



Scutigeral



PPAHV, R_1:  R_2: CH_3

PDDHV, $R_1 = R_2$: $(CH_2)_8CH_3$

From http://pharmrev.aspetjournals.org/cgi/content/full/51/2/159/F8

How does capsaicin work?

First need to know what it's affecting!

Experiments suggested it affects calcium uptake by nerve cells. Using rat cDNA and cell cultures, use calcium imaging to find which cDNA encodes the capsaicin receptor.

Only one sequence was found, cloned, and sequenced. It's now called "VR1" ("vanilloid receptor 1")

(Sounds so simple, doesn't it?)

What does VR1 do?

Proteins often come in families, derived from a common ancestor and separated by mutation and evolution.

Knowing how molecules similar to VR1 work gives ideas of how VR1 works. Let's search for those!

But the techniques from (small molecule) chemical informatics don't work well here. We need something else for the large proteins and nucleic acids used by biology.

Bioinformatics

Use *biologically-based* approaches to store and search biological sequence data.

The primary model is a linear sequence of subunits drawn from a small pool of possible types (4 possible bases in DNA, 20 possible residues in protein), plus random mutation and evolution.

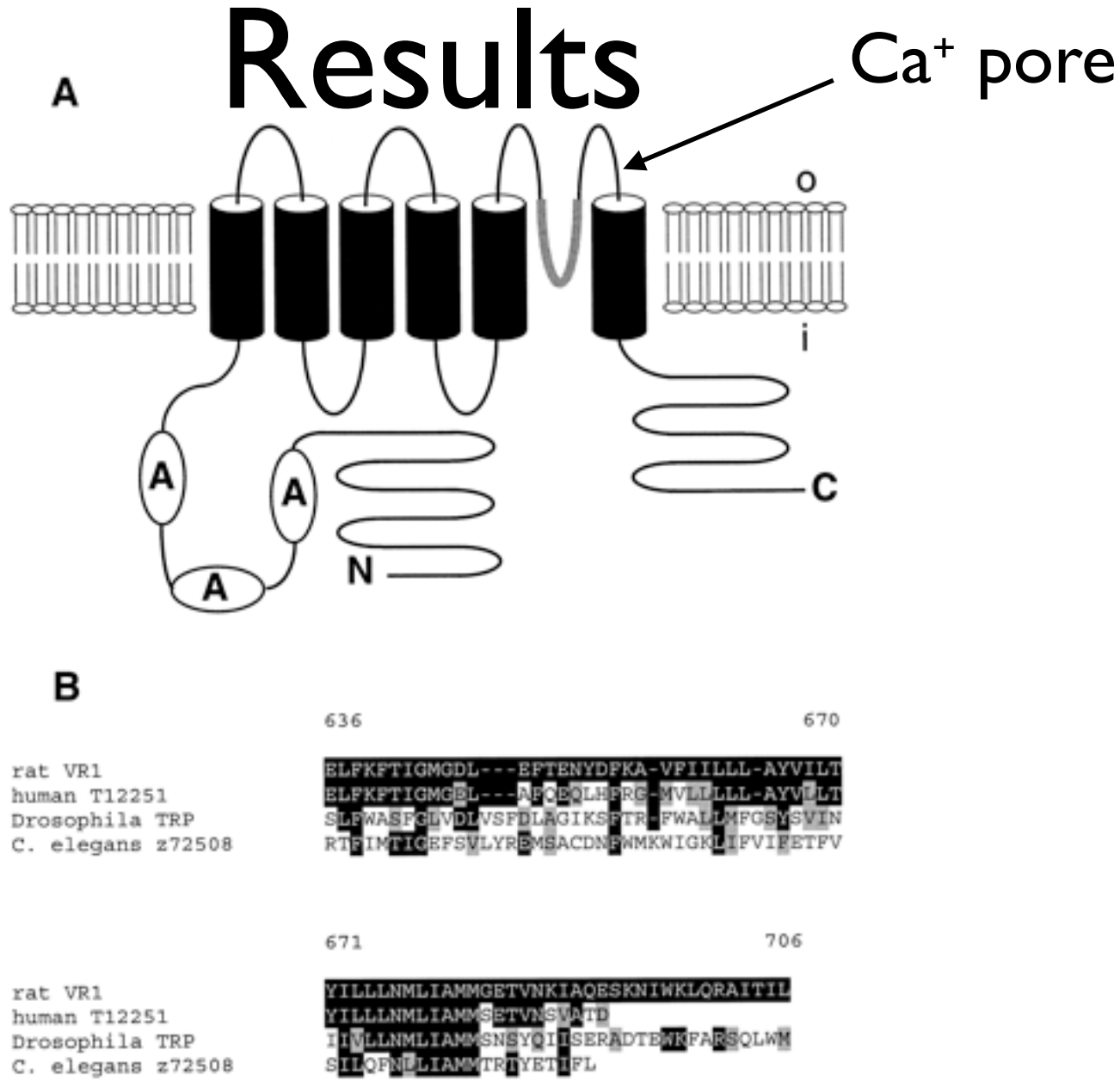
Biopython - biopython.org

- parsers (many bioinformatics formats)
- interfaces to local binaries (NCBIStandalone, ClustalW, Emboss)
- interfaces to web services and databases (NCBIWWW, EUtils, sequence retrieval)
- sequence, alignment, pathways, structure APIs
- implements algorithms for clustering, hidden Markov models, support vector machine
- Supports the BioSQL schema

Similarity-based predictions

- Get the sequence for VR1 (either through NCBI's web interface or Bio.EUtils)
- Use Bio.Blast.NCBIWWW to find similar sequence alignments
- Fetch the corresponding sequence records to find what's known about those regions

Results



From http://pharmrev.aspetjournals.org/cgi/content/full/51/2/159/F8

And in Bio speak....

"The rat VR1 cDNA contains an open reading frame of 2514 nucleotides. This cDNA encodes a protein of 838 amino acids with a molecular mass of 95 kDa. At the N terminus, VR1 has three ankyrin repeat domains (Fig. 9A). The carboxy terminus has no recognizable motifs. Predicted membrane topology of VR1 features six transmembrane domains and a possible pore-loop between the fifth and sixth membrane-spanning regions (Fig. 9A). There are three possible protein kinase A phophorylation sites on the VR1 that might play a role in receptor desensitization.

VR1 is a distant relative of the transient release potential (TRP) family of store-operated calcium channels (Montell and Rubin, 1989; Hardie and Minke, 1993; Wes et al., 1995; Clapham, 1996; Colbert et al., 1997; Roayaie et al., 1998). There is considerable homology between VR1 and the drosophila TRP protein in retina (Fig. 9B). This sequence similarity seems to be restricted to the pore-loop and the adjacent sixth transmembrane segment in VR1. Interestingly, VR1 also shows similarity to a Soares human retina cDNA (L. Hillier, N. Clark, T. Dubuque, K. Elliston, M. Hawkins, M. Holman, M. Hultman, T. Kucaba, M. Le, G. Lennon, M. Marra, J. Parsons, L. Rifkin, T. Rohlfing, F. Tan, E. Trevaskis, R. Waterston, A. Williamson, P. Wohldman and R. Wilson, unpublished observations, Washington University-Merck expressed sequence tags (EST) Project; Accession: AA047763). Because capsaicin causes a marked calcium accumulation in rat retina (Ritter and Dinh, 1993), it might be speculated that the retina has a site, related to VR1, that recognizes vanilloids. OSM-9, a novel protein with similarity to rat VR1, plays a role in olfaction, mechanosensation, and olfactory adaptation in *Caenorhabditis elegans* (Colbert et al., 1997). OSM -9, however, does not recognize capsaicin (Cornelia Bargmann, personal communication). These findings imply that 1) in contrast to previous beliefs, VR isoforms did occur early during evolution, but 2) the capsaicin recognition site is a recent addition to VR1."

From http://pharmrev.aspetjournals.org/cgi/content/full/51/2/159/F8

But how does it work?

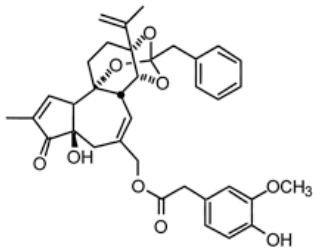
VR1 is heat sensitive. Temperatures over ~48℃ open the pore. Calcium ions go through it, which your nervous system interprets as **pain**.

VR1 is a shape-specific receptor.
It's a "lock" and capsaicin is the "key".

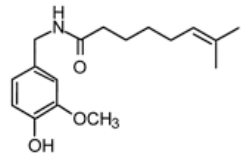
Capsaicin causes VR1 to lower the activation temperature to below body temperature.

Shape of the lock

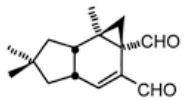
Use molecular modeling and QSAR



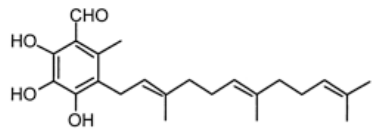
Resiniferatoxin



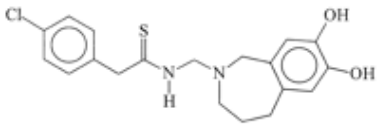
Capsaicin



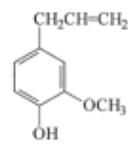
Isovelleral



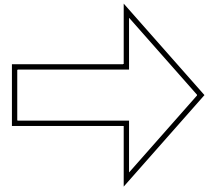
Scutigeral

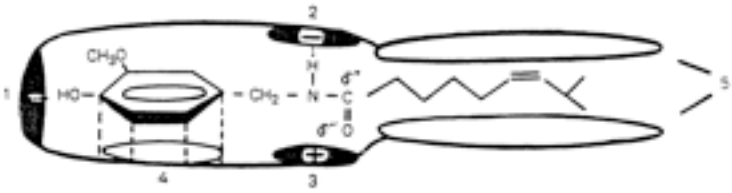


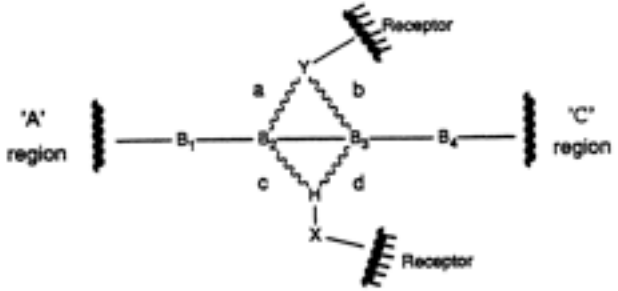
Capsazepine

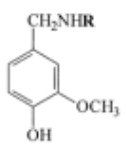


Eugenol





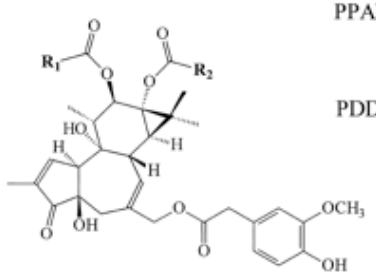




Capsaicin, R: $CO(CH_2)_7CH_3$

Olvanil, R: $CO(CH_2)_7CH{=}CH(CH_2)_7CH_3$

Compound 57, R: $CSNH(CH_2)_7CH_3$



PPAHV, R_1:  R_2: CH_3

PDDHV, $R_1 = R_2$: $(CH_2)_8CH_3$

From http://pharmrev.aspetjournals.org/cgi/content/full/51/2/159/F8

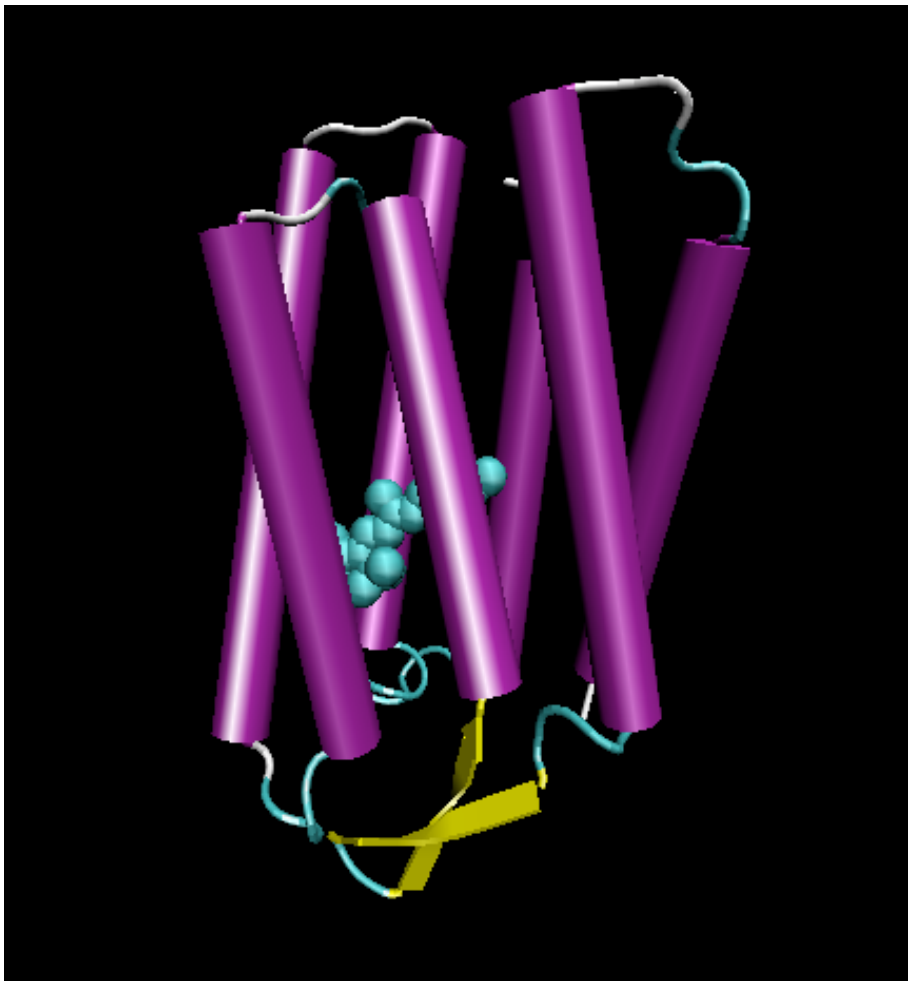
OpenEye's libraries

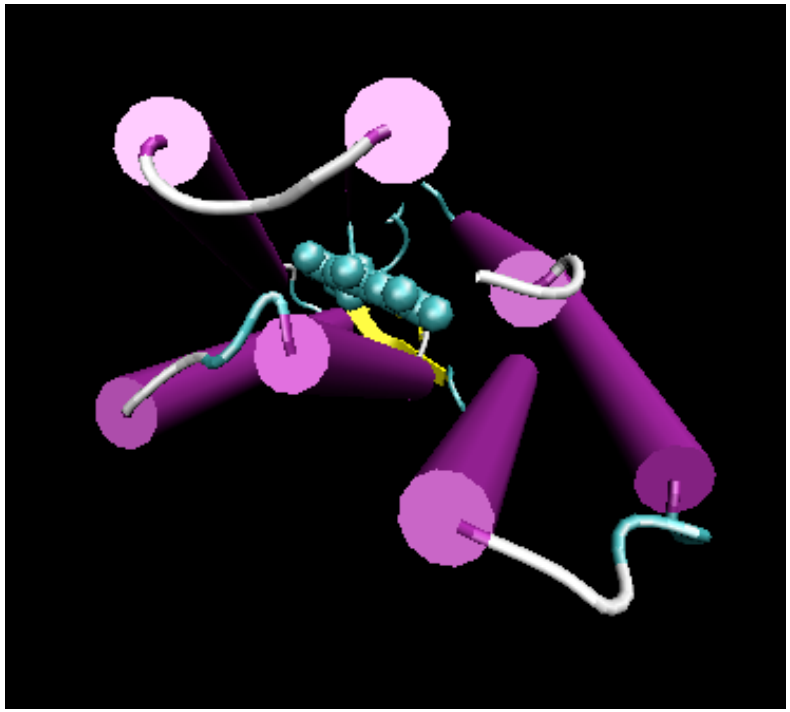
www.eyesopen.com

- C++ with medium-weight Python bindings
- Good chemical informatics capabilities (except databases and fingerprints)
- Strong support for 3D structure, conformation generation, electrostatics, and shape fitting
- ... and they keep writing more code!
- Focus on high-performance

View the structure

(VR1 structure isn't known - this bacteriorhodopsin)





Some Python structure visualization programs

- PyMol - www.delanoscientific.com
- VMD - www.ks.uiuc.edu/Research/vmd/
- PMV and ViPEr - www.scripps.edu/~sanner
- Chimera - www.cgl.ucsf.edu

Molecular mechanics

Capsaicin binding causes some sort of change to the VR1 structure.

Can simulate it numerically with molecular mechanics.

MMTK - The Molecular Modelling Toolkit

starship.python.net/crew/hinsen/MMTK/

Quantum mechanics

Sometimes molecular mechanics isn't enough.
(Probably is okay for capsaicin/VR1 modeling.)

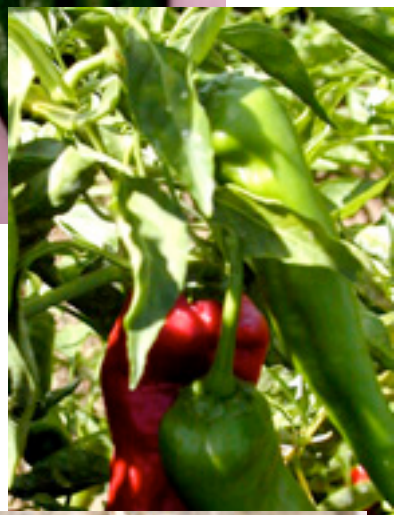
It doesn't make/break bonds, change energy states, react with light (as in photosynthesis)

Need quantum mechanics instead.

PyQuante - pyquante.sourceforge.net

Summary

- Python popular in structural biology and small-molecule chemistry
- Less common in bioinformatics (Perl) and quantum mechanics (Fortran)
- Others I didn't mention (crystallography, NMR, metabolism, gene expression)



Roasted Hatch Chile