Computational modeling of NMR chemical shifts for structure elucidation: From empirical modeling to quantum chemistry and machine learning

Hands-on exercises

Raghunathan Ramakrishnan
ramakrishnan@tifrh.res.in
Tata Institute of Fundamental Research Hyderabad, India

27 January 2026

NMR Meets Biology 6
25-31 January 2026, Gokarna, Karnataka, India

General information

- It may not be possible to complete all the exercises discussed during the workshop. Participants are encouraged to skim through the exercises and select the problems they would like to work on during the hands-on sessions. Some exercises can also be attempted in the evenings after the lectures and discussed informally during breaks.
- During the workshop, you may perform a full DFT calculation for one selected exercise, while for the remaining tutorials, you can work through the provided input and output files to understand the methodology and results.
- Please work on the exercises in small groups to encourage discussion and collaborative learning.
- All workshop materials will be maintained on GitLab: https://gitlab.com/raghurama123/nmrworkshop2026

Software requirements (optional)

- Electronic structure calculation
    - Orca (input/output files are provided)
    - Quantum Espresso (input/output files are provided)

- Structure visualisation
    - Jmol/Avogadro/Vesta/VMD/Pymol, use anything that works

Content

- Exercises 1-6: *Core / Hands-on*
- Exercises 7-8: *Advanced / Discussion-oriented*

Basic tasks and keywords in DFT

It is the most popular electronic structure method, founded on the Hohenberg-Kohn existence theorem. The ground state energy and all other ground state properties are uniquely determined by the electron density $\rho(\vec{r})$. So, the ground state energy can be written as an energy functional

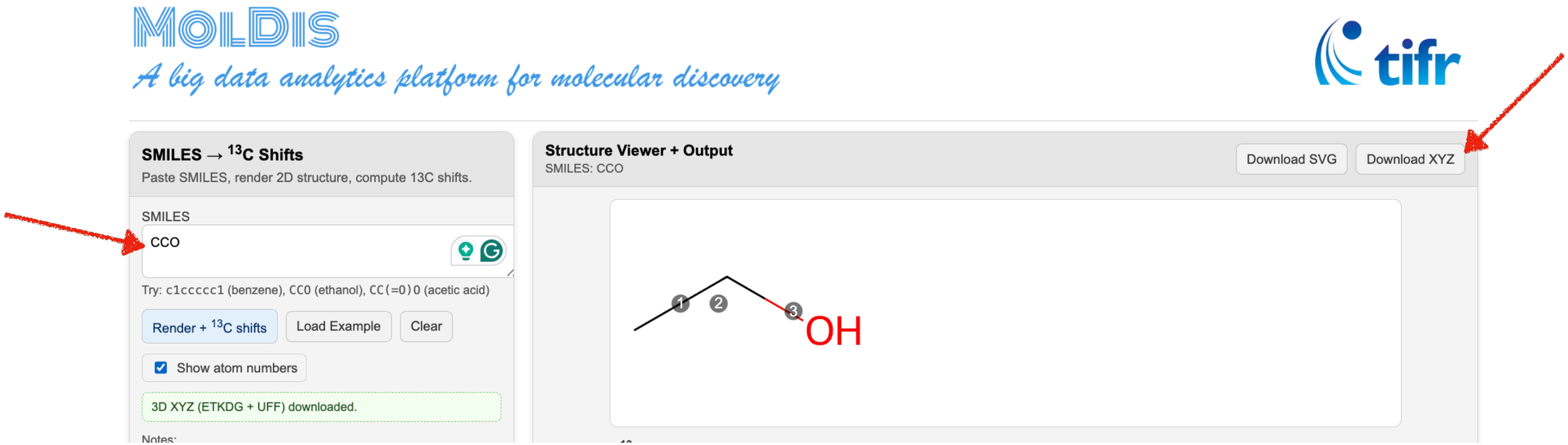
$$E\left[\rho\right] = T\left[\rho\right] + V_{\text{Coloumb}}\left[\rho\right] + \int dr\rho(r)v_{\text{Ne}}(r) + E_{\text{XC}}\left[\rho\right]$$

- ❖ The wave function in KS-DFT is a Slater determinant, defined using molecular orbitals of a hypothetical reference system of non-interacting electrons.
- ❖ The exchange–correlation functional $E_{\text{XC}}\left[\rho\right]$ is not known exactly; different approximations have well-understood strengths and limitations.
- ❖ For new applications, multiple XC functionals are often tested to rationalise method selection (benchmarking).
- ❖ Common XC functionals include: LDA, PBE, BP86, PBE0, **`B3LYP`**, CAM-B3LYP, ωB97X, M06 family.
- ❖ Typical Gaussian basis sets used in molecular DFT: STO-3G, 3-21G, 6-31G, **`6-31+G(d,p),`** cc-pVDZ, cc-pVTZ, def2-SVP, def2-TZVP
- ❖ Total energy $E[\rho]$ is the central quantity in DFT calculations.
- ❖ The equilibrium molecular structure (determined with optimization, **`Opt`**) corresponds to a stationary point on the potential energy surface where all nuclear gradients $\{dE/d\overrightarrow{R}_A\}$ vanish.
- ❖ Harmonic vibrational frequencies (determined with **`Freq`**) are obtained from the Hessian matrix; a true minimum is characterized by all real (positive) frequencies.

Exercise 1: Geometry optimization of ethanol molecule with ORCA

The first step in geometry optimization is to obtain a reasonable initial molecular structure. Visit the MolDis-Lab interactive application: https://moldis.tifrh.res.in/C13.html.

- Use the SMILES input **CCO** to generate the structure of ethanol and download the corresponding XYZ file, which will be saved locally as **`Mol_CCO_UFF.xyz`**.



- The downloaded structure is an approximate geometry obtained using the Universal Force Field (UFF) and requires further refinement. Later, we will see that an ML model uses this force-field-based geometry to predict chemical shifts.
- You can visualize the downloaded XYZ file using any molecular viewer installed on your system (e.g., PyMOL) to inspect the 3D structure.

Download the workshop materials (this presentation PDF, exercises, and their solutions) from
https://gitlab.com/raghurama123/nmrworkshop2026

- Navigate in your terminal to
  `nmrworkshop2026/exercises/ex01`,
  which contains the ORCA input file for geometry optimization (**opt.com**).
- Copy the file **`Mol_CCO_UFF.xyz`** into this directory.
- The ORCA input file **`opt.com`** specifies a DFT geometry optimization followed by a harmonic frequency calculation using the **`B3LYP`** functional and the **`6-31+G(d,p)`** basis set. Empirical dispersion (**`D4`**) and the **`RIJCOSX`** approximation (with **`def2/J`** auxiliary basis set) are used to improve accuracy and efficiency.

```
! B3LYP D4 6-31+G(d,p) def2/J RIJCOSX Opt Freq

* xyzfile 0 1 Mol_CCO_UFF.xyz
```

On Windows the last line should be replaced by

```
* xyz 0 1
…
coordinates
…
end
```

- The keywords **`Opt`** and **`Freq`** request geometry optimization and evaluation of the harmonic force constant matrix, respectively. The **`Freq`** calculation is used to verify the nature of the stationary point and to obtain harmonic vibrational frequencies.
- You are now ready to run the calculation from your terminal using:
  `/Users/rr/ORCA/orca_6_0_0_macosx_openmpi411/orca opt.com | tee out.out`
  Replace the ORCA executable path with the location of ORCA on your system.
- After the calculation finishes, the optimized geometry is written to the file **`opt.xyz`**.

These are the two key sections of the output file (**`opt.out`**) that should be inspected after a geometry optimization and frequency calculation.

```
                          .--------------------.
  ----------------------|Geometry convergence|-------------------------
  Item                value                   Tolerance       Converged
  ---------------------------------------------------------------------
  Energy change      -0.0000074133            0.0000050000      NO
  RMS gradient        0.0000878235            0.0001000000      YES
  MAX gradient        0.0002851820            0.0003000000      YES
  RMS step            0.0007460608            0.0020000000      YES
  MAX step            0.0016162100            0.0040000000      YES
  ........................................................
  Max(Bonds)      0.0006      Max(Angles)    0.05
  Max(Dihed)        0.09      Max(Improp)    0.00
  ---------------------------------------------------------------------

Everything but the energy has converged. However, the energy
appears to be close enough to convergence to make sure that the
final evaluation at the new geometry represents the equilibrium energy.
Convergence will therefore be signaled now


             ***********************HURRAY********************
             ***        THE OPTIMIZATION HAS CONVERGED     ***
             *************************************************
```

This confirms that a stationary point on the potential energy surface has been reached; however, it does not indicate whether the structure corresponds to a local minimum or a saddle point, as both satisfy the stationary-point condition.

```
-----------------------
VIBRATIONAL FREQUENCIES
-----------------------

Scaling factor for frequencies =  1.000000000  (already applied!)

   0:         0.00 cm**-1
   1:         0.00 cm**-1
   2:         0.00 cm**-1
   3:         0.00 cm**-1
   4:         0.00 cm**-1
   5:         0.00 cm**-1
   6:       263.94 cm**-1
   7:       291.29 cm**-1
   8:       421.14 cm**-1
   9:       805.32 cm**-1
  10:       885.14 cm**-1
  11:      1056.92 cm**-1
  12:      1072.89 cm**-1
  13:      1133.27 cm**-1
  14:      1272.93 cm**-1
  15:      1363.69 cm**-1
  16:      1402.52 cm**-1
  17:      1422.67 cm**-1
  18:      1491.57 cm**-1
  19:      1496.82 cm**-1
  20:      1518.76 cm**-1
  21:      2999.16 cm**-1
  22:      3028.53 cm**-1
  23:      3083.40 cm**-1
  24:      3102.58 cm**-1
  25:      3117.30 cm**-1
  26:      3818.25 cm**-1
```

For a non-linear molecule, three normal modes correspond to pure translation and three to pure rotation. These six modes have zero eigenvalues of the force constant matrix, as they do not represent internal degrees of freedom. The absence of imaginary frequencies confirms that the optimized structure is a true minimum.

Exercise 2: Calculation of the NMR shielding tensor of ethanol

- Navigate in your terminal to
  `nmrworkshop2026/exercises/ex02`,
  which contains the ORCA input file for calculation of the shielding tensor (**nmr.com**).
- Copy the file **`opt.xyz`**, which contains the DFT-optimized geometry from **`nmrworkshop2026/exercises/ex01`**, into this directory.
- Compare **`nmr.com`** with **`opt.com`** from Exercise 1. The keywords **`Opt`** and **`Freq`** are replaced with **`NMR`**. Empirical dispersion (**`D4`**), which improves the description of intermolecular interactions and equilibrium geometries, is not required for NMR shielding calculations and is therefore omitted.

```
! B3LYP 6-31+G(d,p) def2/J RIJCOSX NMR

* xyzfile 0 1 opt.xyz
```

On Windows the last line should be replaced, see slide 7.

- The keyword **`NMR`** requests the calculation of nuclear magnetic shielding tensors at the fixed, DFT-optimized geometry provided in **`opt.xyz`**.
- You are now ready to run the calculation from your terminal using:
  `/Users/rr/ORCA/orca_6_0_0_macosx_openmpi411/orca nmr.com | tee nmr.out`
  Replace the ORCA executable path with the location of ORCA on your system.
- Once the calculation finishes, you may notice that the directory contains many files with extensions **`.gbw`**, **`.densities`**, **`.properties.txt`**. These files can be used for further analysis, such as visualization and post-processing of shielding tensors. Remember, ORCA calculates the shielding tensor and not chemical shifts directly.

Key sections of the output file (**nmr.out**)

$$\sigma_{\alpha,\beta} = \frac{\partial^2 E}{\partial B_\alpha \partial m_{A,\beta}} \neq \frac{\partial^2 E}{\partial m_{A,\alpha} \partial B_\beta} = \sigma_{\beta,\alpha}$$

Non-symmetric in general

```
 ---------------
 Nucleus   0C :
 ---------------

Diamagnetic contribution to the shielding tensor (ppm) :
           253.322           2.394          4.364
             3.560         242.196          1.087
             4.913           1.224        243.794

Paramagnetic contribution to the shielding tensor (ppm):
           -63.449          -5.170          1.265
             1.595         -79.425          8.353
             6.163           7.106        -80.738

Total shielding tensor (ppm):
           189.874          -2.775          5.629
             5.155         162.771          9.440
            11.076           8.330        163.056


 Diagonalized sT*s matrix:

 sDSO           241.751          241.880          255.681  iso=      246.437
 sPSO           -88.460          -72.191          -62.960  iso=      -74.537
           ---------------  ---------------  ---------------
 Total          153.291          169.689          192.720  iso=      171.900

 Orientation:
  X          0.1471320    0.2833014   -0.9476774
  Y          0.6658981   -0.7368294   -0.1168856
  Z         -0.7313904   -0.6138590   -0.2970611
```

```python
# Total shielding tensor
S=np.array([
    [189.874,  -2.775,   5.629],
    [  5.155, 162.771,   9.440],
    [11.076,    8.330, 163.056]
])
```

```python
# Principal values via S^T S (eigenvalue approach)
STS=np.matmul(S.T,S)
E,V=np.linalg.eig(STS)
L1,L2,L3=np.sqrt(E)
print(f'{L1:.3f} {L2:.3f} {L3:.3f}')
```

```
192.789 169.749 153.292
```

```python
V # Principal axis of the shielding tensor
```

```
array([[ 0.94767951,  0.28330007, -0.14712111],
       [ 0.11688777, -0.73682026, -0.66590777],
       [ 0.29705353, -0.61387049,  0.73138377]])
```

```python
# Principal values via singular value decomposition (preferred)
U, s, Vt = np.linalg.svd(S)
L1,L2,L3=s
print(f'{L1:.3f} {L2:.3f} {L3:.3f}')
```

```
192.789 169.749 153.292
```

```python
V # Principal axis of the shielding tensor
```

```
array([[ 0.94767951,  0.28330007, -0.14712111],
       [ 0.11688777, -0.73682026, -0.66590777],
       [ 0.29705353, -0.61387049,  0.73138377]])
```

- The diamagnetic contribution is always positive (shielding), whereas the paramagnetic contribution is typically negative (deshielding).

$$\sigma_{\text{iso}} = \frac{1}{3}\left(\sigma_{11} + \sigma_{22} + \sigma_{33}\right)$$

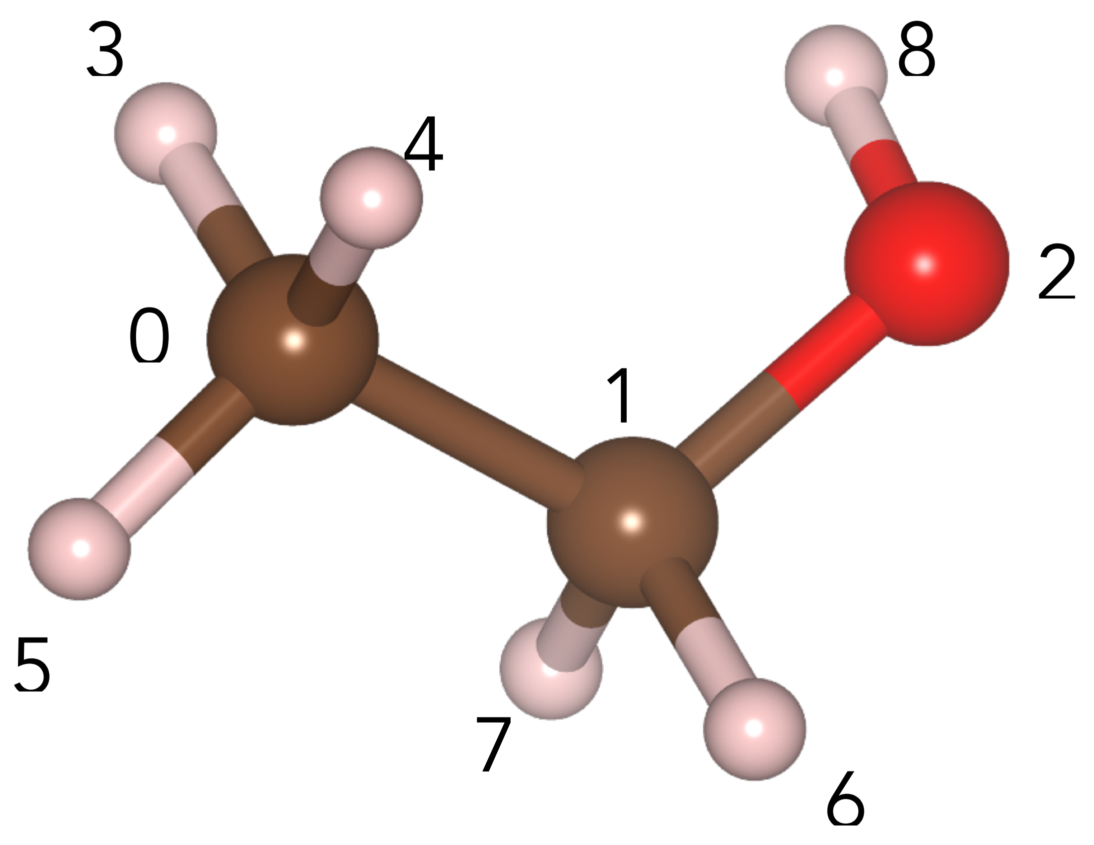
$$\sigma_{\text{aniso}} = \sigma_{33} - \frac{1}{2}\left(\sigma_{11} + \sigma_{22}\right)$$

```
---------------------------------
CHEMICAL SHIELDING SUMMARY (ppm)
---------------------------------
```

Nucleus	Element	Isotropic	Anisotropy
0	C	171.900	31.230
1	C	131.349	62.948
2	O	283.056	40.440
3	H	30.730	6.935
4	H	30.217	7.393
5	H	30.544	9.089
6	H	27.972	6.311
7	H	27.770	6.830
8	H	31.670	15.080

```python
import numpy as np

S = np.array([
    [189.874,  -2.775,   5.629],
    [  5.155, 162.771,   9.440],
    [ 11.076,   8.330, 163.056]
])

# Principal values via singular value decomposition
U, s, Vt = np.linalg.svd(S)

# Sort descending: sigma_33 >= sigma_22 >= sigma_11
sigma_33, sigma_22, sigma_11 = np.sort(s)[::-1]
print(sigma_33, sigma_22, sigma_11 )

# Isotropic shielding
sigma_iso = (sigma_11 + sigma_22 + sigma_33) / 3.0

# Anisotropy (ORCA convention)
delta_sigma = sigma_33 - 0.5 * (sigma_11 + sigma_22)

print("sigma_iso   =", sigma_iso)
print("anisotropy  =", delta_sigma)
```

```
192.78912167510836 169.74904569824645 153.29236793876893
sigma_iso   = 171.94351177070791
anisotropy  = 31.268414856600685
```

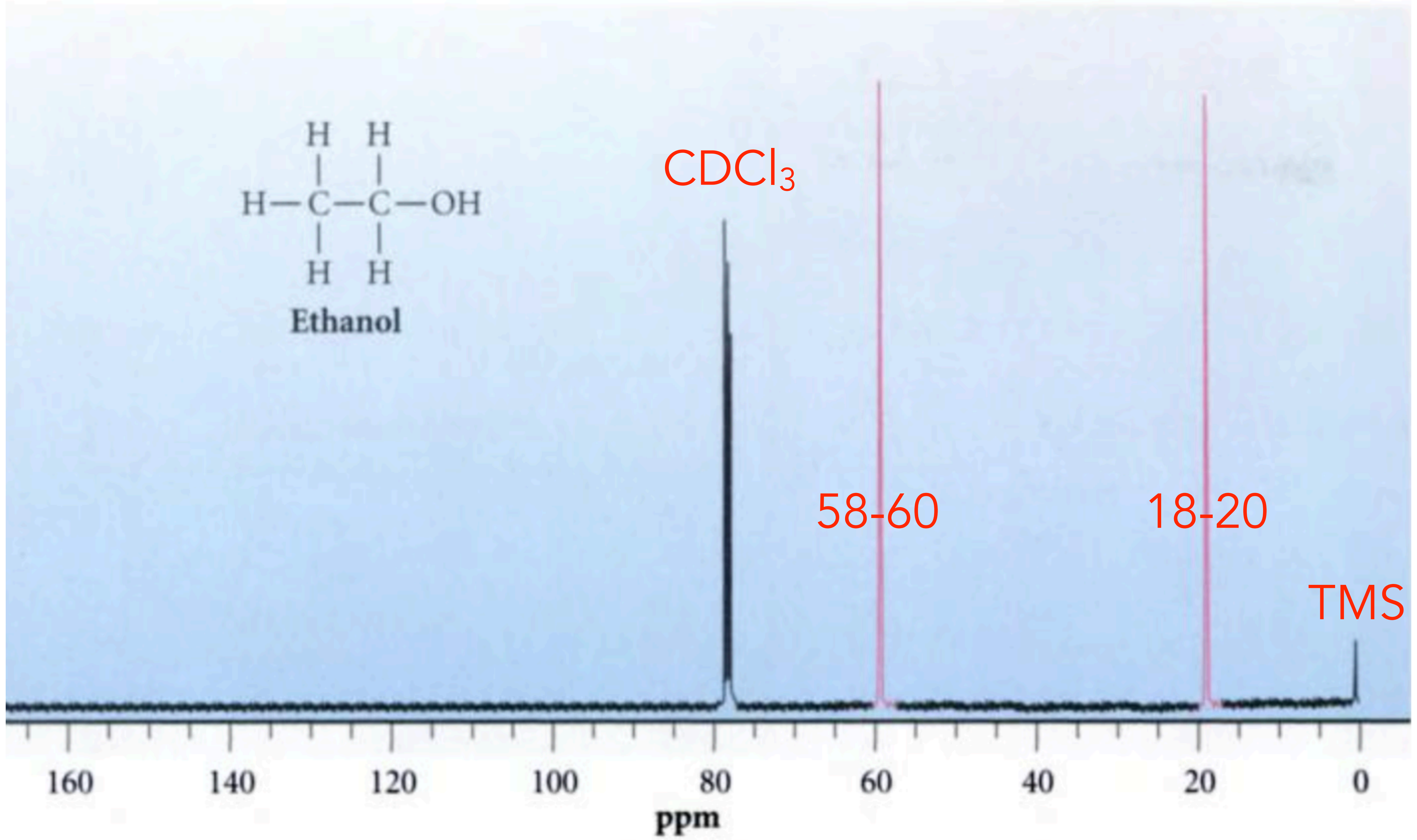


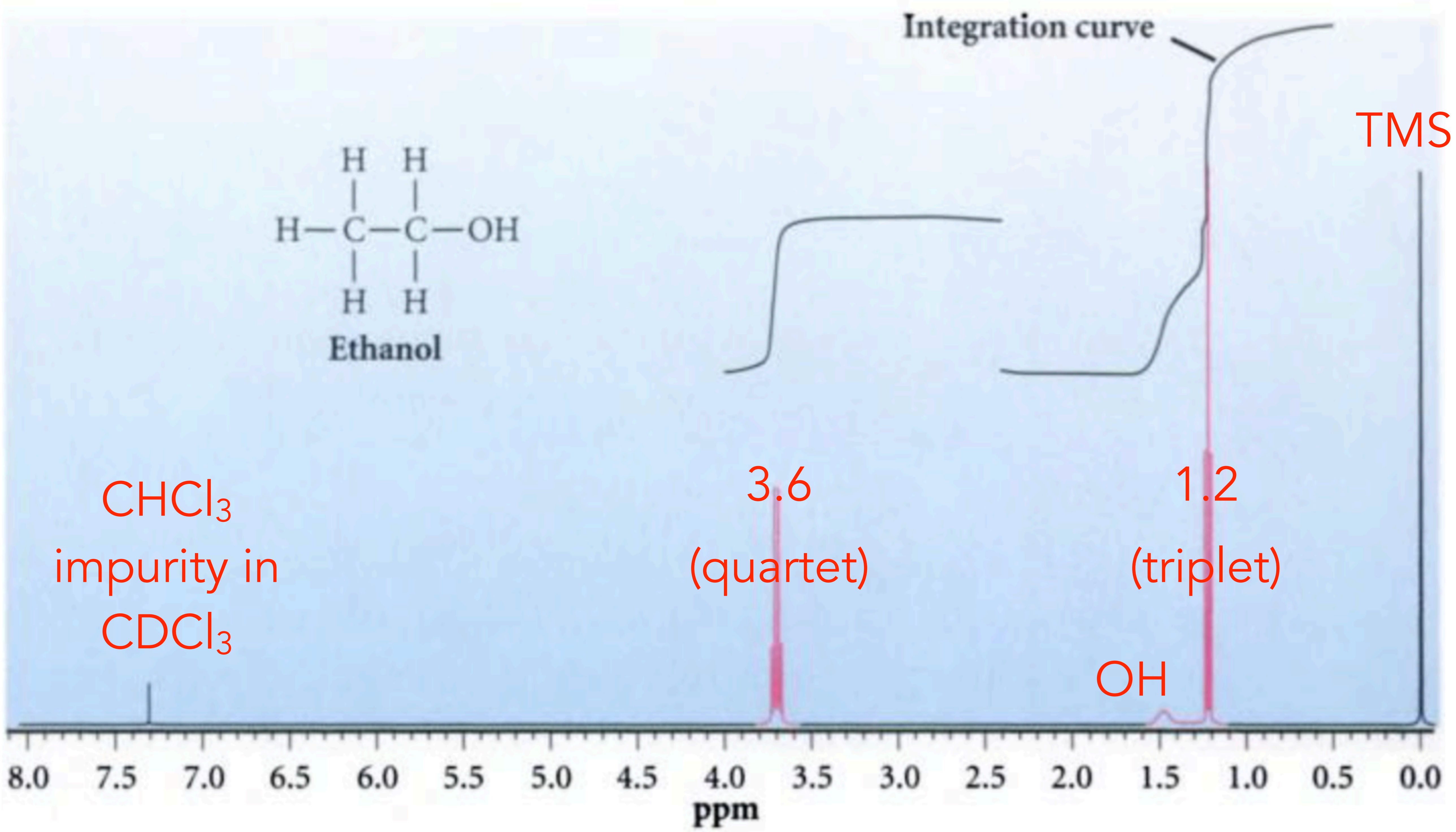
Ethanol (optimized geometry)

- NOTE: σ_{11}, σ_{22}, and σ_{33} are eigenvalues of the shielding tensor; $\sigma_{33} \geq \sigma_{22} \geq \sigma_{11}$.
- In solution, rapid molecular tumbling averages the chemical shielding tensor.
- Only the isotropic shielding σ_{iso} is observed.
- Anisotropy → directionality of electronic response
- C, O: large anisotropy (directional bonding, lone pairs)
- H: small anisotropy (isotropic environment)

Exercise 3: Calculation of the ^{1}H and ^{13}C chemical shifts of ethanol

- As done in Exercise 1, visit MolDis-Lab and use the SMILES input `[Si](C)(C)(C)C` to generate the structure of the reference compound tetramethylsilane (TMS). Navigate in your terminal to **`nmrworkshop2026/exercises/ex03`**, and perform geometry optimization (**`opt.com`**) and calculate the isotropic shielding values (**nmr.com**).
- Calculate ^{1}H and ^{13}C chemical shifts using isotropic shielding values of ethanol from Exercise 2 and TMS from this exercise as $\delta_X = \sigma_X^{\mathrm{TMS}} - \sigma_X^{\mathrm{ethanol}}$, where X is the nucleus ^{1}H or ^{13}C.
- Compare the DFT-based predictions of ^{1}H and ^{13}C chemical shifts and compare them against the experimental values shown in the image below.





M. A. Fox, J. A. Whitesell, *"Organic Chemistry"*

TMS from Exercise 3

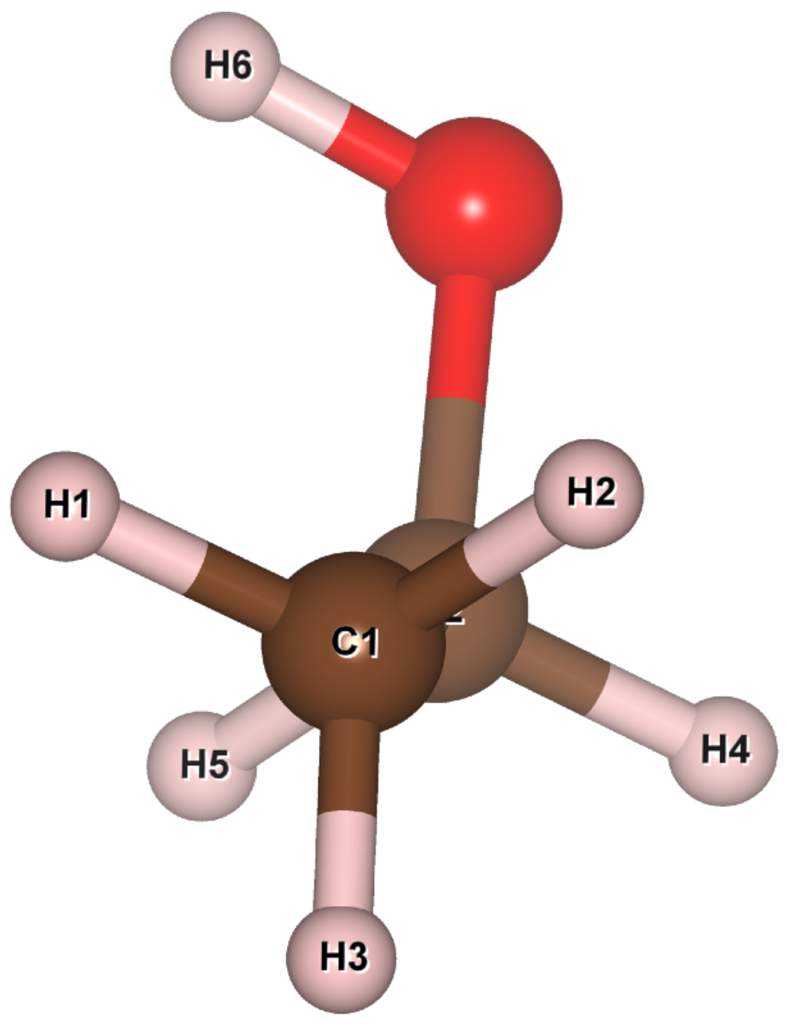
```
---------------------------------
CHEMICAL  SHIELDING  SUMMARY  (ppm)
---------------------------------


  Nucleus   Element     Isotropic
  -------   -------   ------------
     0        Si         408.100
     1        C          192.880
     2        C          192.901
     3        C          192.887
     4        C          192.882
     5        H           31.634
     6        H           31.633
     7        H           31.631
     8        H           31.630
     9        H           31.632
    10        H           31.633
    11        H           31.632
    12        H           31.633
    13        H           31.633
    14        H           31.635
    15        H           31.630
    16        H           31.632
```

$$\sigma^{\mathrm{TMS}}_{^{13}\mathrm{C}} = 192.9\,\mathrm{ppm} \quad \sigma^{\mathrm{TMS}}_{^{1}\mathrm{H}} = 31.6\,\mathrm{ppm}$$

will be used for all problems (unless stated otherwise) as long as the geometry optimization and shielding tensor calculation were done with the same method.

Ethanol from Exercise 2

```
---------------------------------
CHEMICAL  SHIELDING  SUMMARY  (ppm)
---------------------------------


  Nucleus   Element     Isotropic
  -------   -------   ------------
     0        C          171.900
     1        C          131.349
     2        O          283.056
     3        H           30.730
     4        H           30.217
     5        H           30.544
     6        H           27.972
     7        H           27.770
     8        H           31.670
```



Conformer structure of Ethanol

```python
TMS_C=192.9; TMS_H=31.6

C_ethanol=np.array([171.900, 131.349])
H_ethanol=np.array([30.730, 30.217, 30.544,
                    27.972, 27.770, 31.670])

C_shifts=TMS_C-C_ethanol
H_shifts=TMS_H-H_ethanol

for shift in C_shifts:
    print(f'{shift:.3f} ppm')
```

```
21.000 ppm
61.551 ppm
```

```python
for shift in H_shifts:
    print(f'{shift:.3f} ppm')
```

```
0.870 ppm
1.383 ppm
1.056 ppm
3.628 ppm
3.830 ppm
-0.070 ppm
```

For comparison with experiment, average the computed shieldings over symmetry-equivalent nuclei (e.g., CH_3 and CH_2 groups).

```python
H_shifts_mean=[ (0.870+1.383+1.056)/3, (3.628+3.830)/2]

for shift in H_shifts_mean:
    print(f'{shift:.3f} ppm')
```

```
1.103 ppm
3.729 ppm
```

Exercise 4: Convergence of ^{13}C chemical Shifts in Alicyclic Rings

Let us verify the textbook trends for ^{13}C chemical shifts of non-aromatic monocyclic rings using the interactive app provided on MolDis-Lab: https://moldis.tifrh.res.in/C13.html.

Use the following SMILES inputs to generate predicted ^{13}C shifts:

Molecule	**SMILES**
Cyclopropane	C1CC1
Cyclobutane	C1CCC1
Cyclopentane	C1CCCC1
Cyclohexane	C1CCCCC1
Cycloheptane	C1CCCCCC1

You may notice small numerical differences compared to standard textbook tables, since the app uses simplified empirical parameters.

Q1. From the app, determine from which ring size onward the predicted ^{13}C chemical shift for the ring CH_2 carbon no longer changes (i.e., has effectively converged).

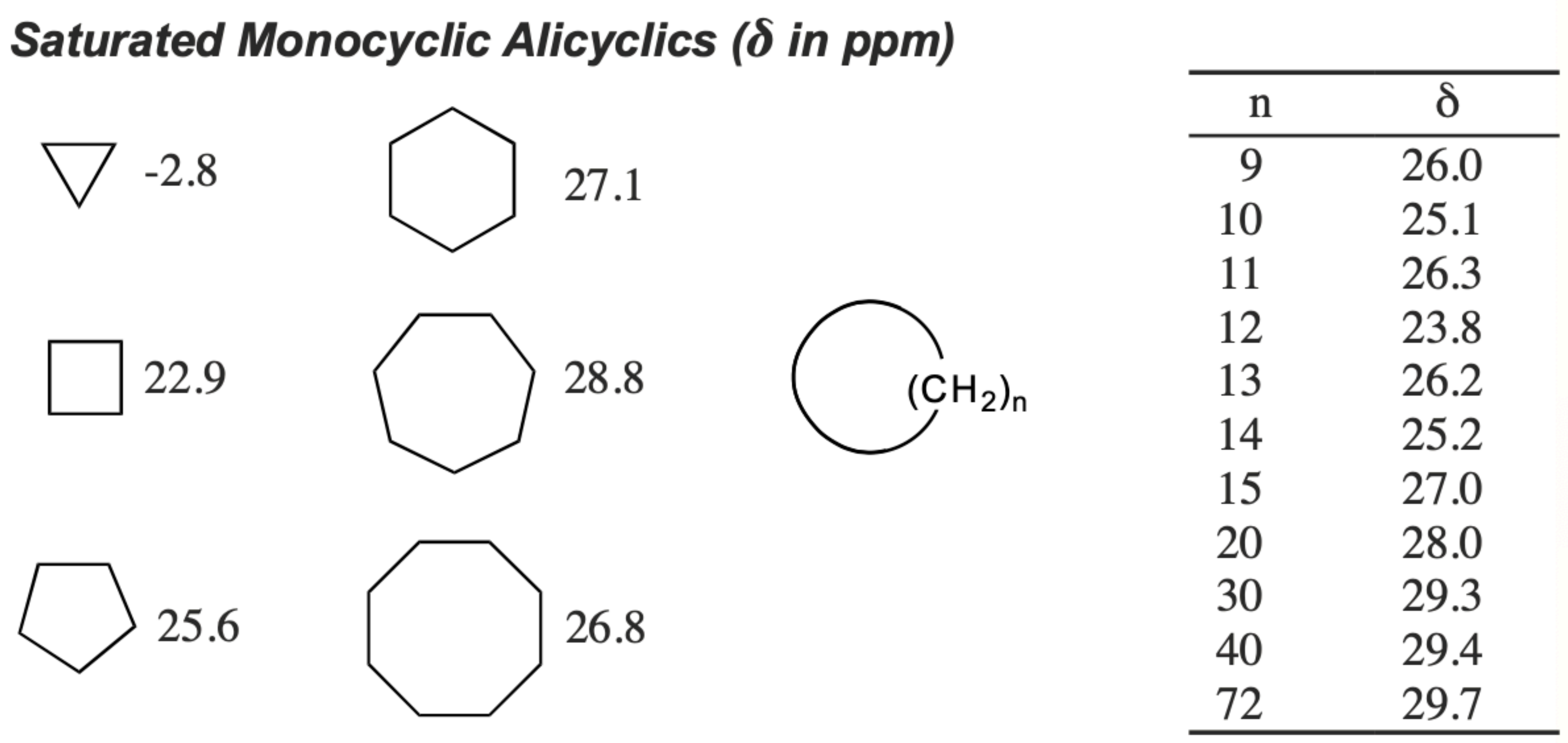
Q2. Now consider the simple empirical model, $\delta_N = \delta_\infty - A/N$, where N is the ring size. Use this model (with $\delta_\infty = 30$ ppm and $A = 25$ ppm) to estimate δ_N for $N = 10, 20, 30$ and check if the shift has converged to within 0.1 ppm of the limiting value. Compare your conclusions from (Q1) (the app) and (Q2) (the model).

Q3. Why is the carbon signal in cyclopropane extremely shielded, even appearing at a negative chemical shift?

Molecule	SMILES	$\delta^{13}C$ in ppm
Cyclopropane	C1CC1	-2.6
Cyclobutane	C1CCC1	23.3
Cyclopentane	C1CCCC1	26.5
Cyclohexane	C1CCCCC1	27.8
Cycloheptane	C1CCCCCC1	29.4
Cyclooctane	C1CCCCCCC1	30.0
Cyclononane	C1CCCCCCCC1	30.0

```
import numpy as np

delta_inf=30
A=25

def delta_N(N):
    return delta_inf - A/N

N=np.array([10,20,30])

delta_N(N)
```

```
array([27.5        , 28.75       , 29.16666667])
```



E. Pretsch, P. Buehlmann, A. Affolter,
"Structure determination of organic compounds"

Atoms-in-molecule machine learning for chemical shifts: Kernel-ridge regression

Property prediction

$$\delta_q = \sum_t c_t k_{qt}; \quad k_{qt} = \exp\left(-|\mathbf{d}_q - \mathbf{d}_t|/\sigma\right)$$

Training

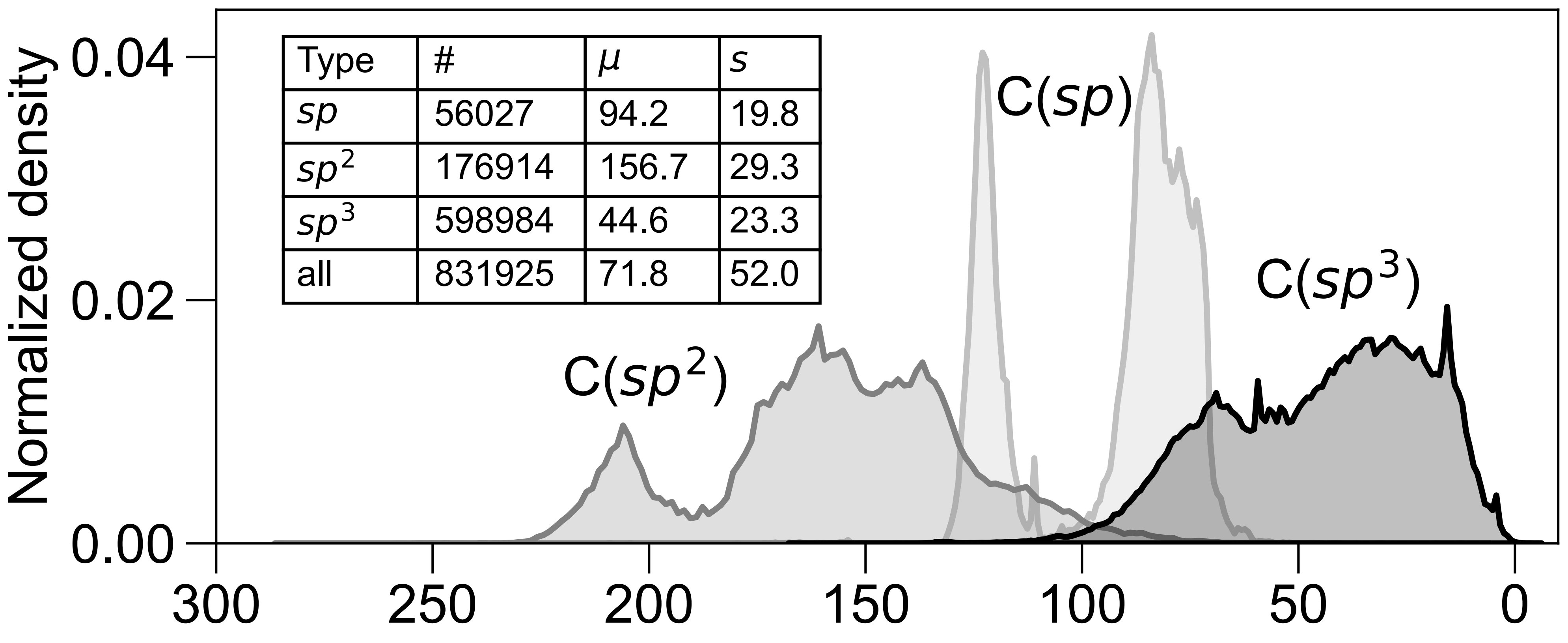
$$[\mathbf{K} + \lambda \mathbf{I}]\, \mathbf{c} = \mathbf{p}$$

chemical shift of query atom, q

hyperparameter

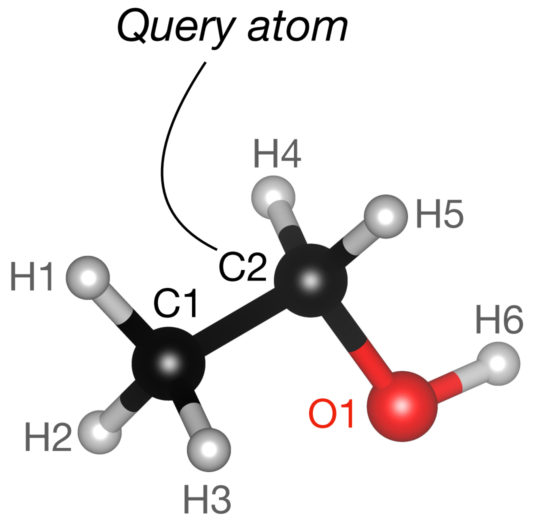
- NMR properties calculated using mPW1PW91/6-311+G(2d,p) for geometries determined with the B3LYP/6-31G(2df,p)

C shifts of 130,831 small organic molecules in QM9NMR dataset (with up to 9 CONF atoms)



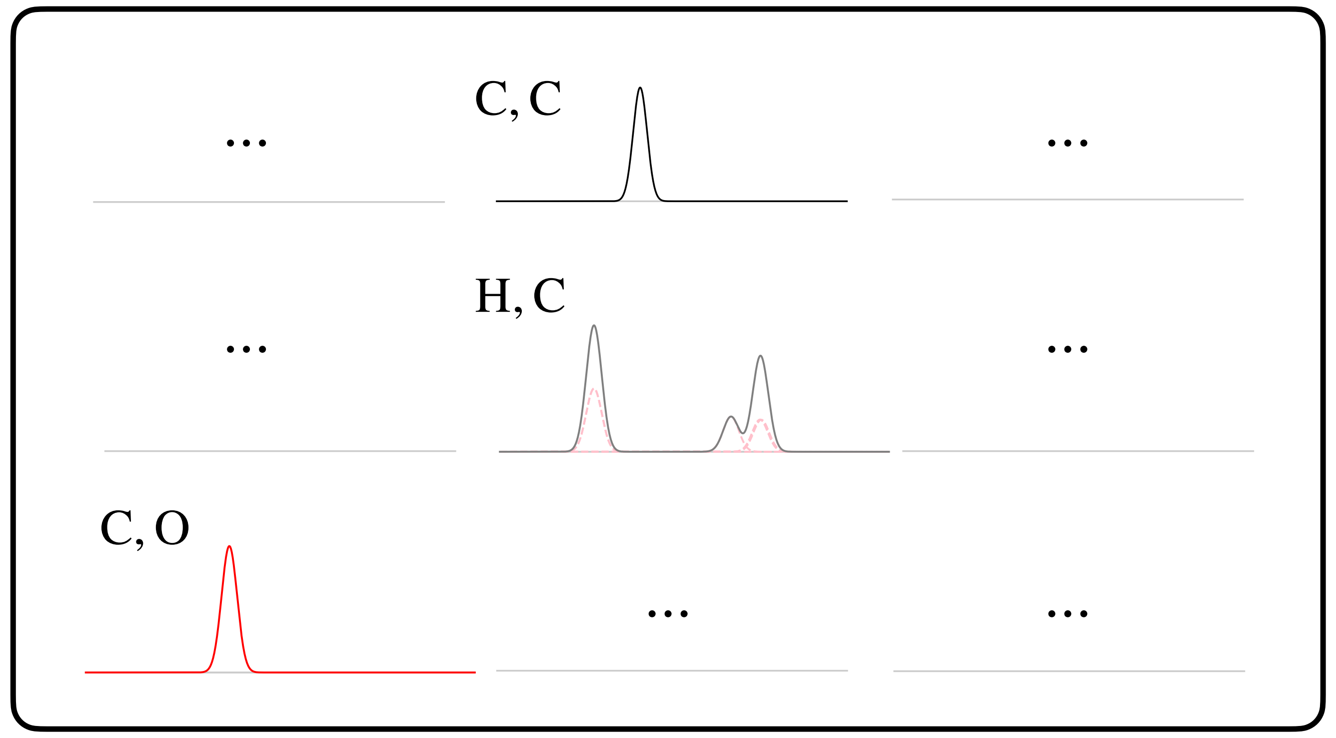
Type	#	μ	s
sp	56027	94.2	19.8
sp^2	176914	156.7	29.3
sp^3	598984	44.6	23.3
all	831925	71.8	52.0

Atomic descriptor: atomic bag-of-bonds based on radial basis functions, aBoB-RBF(4)



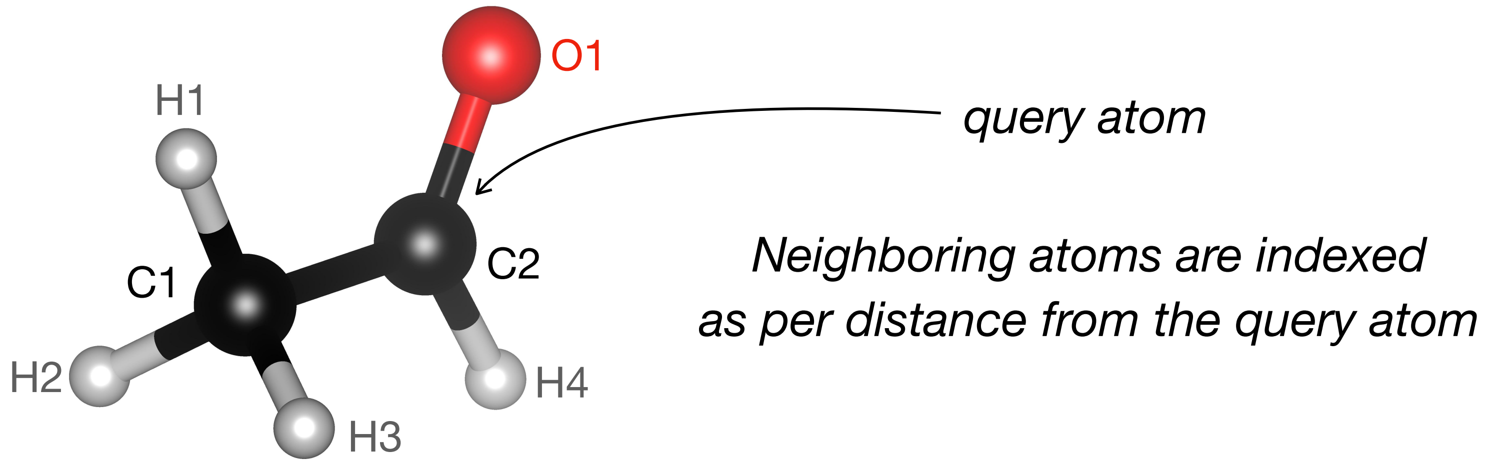
Pairwise functions:

$$\mathbf{d}^{(A,B)}(r) = \sum_{J \neq I, I \in A, J \in B} g_{IJ}(r) \cdot \frac{Z_I Z_J}{R_{IJ}} \cdot s(R_{IJ})$$



Concatenated pairwise functions:

$$\mathbf{d}(r) = \left[\mathbf{d}^{(\mathrm{H,H})}(r), \mathbf{d}^{(\mathrm{C,C})}(r), \cdots, \mathbf{d}^{(\mathrm{H,C})}(r), \cdots, \mathbf{d}^{(\mathrm{C,N})}(r), \mathbf{d}^{(\mathrm{C,O})}(r), \cdots\right]$$



$\mathbf{d}(0) = \mathbf{d}_{\mathrm{C2}}$ *Query atom's descriptor vector*

$\mathbf{d}(1) = \mathbf{d}_{\mathrm{C2}}\ \mathbf{d}_{\mathrm{H4}}$ *First neighbour's descriptor is padded*

$\mathbf{d}(2) = \mathbf{d}_{\mathrm{C2}}\ \mathbf{d}_{\mathrm{H4}}\ \mathbf{d}_{\mathrm{O1}}$ *Second neighbour*

$\mathbf{d}(3) = \mathbf{d}_{\mathrm{C2}}\ \mathbf{d}_{\mathrm{H4}}\ \mathbf{d}_{\mathrm{O1}}\ \mathbf{d}_{\mathrm{C1}}$ *Third neighbour*

$\mathbf{d}(4) = \mathbf{d}_{\mathrm{C2}}\ \mathbf{d}_{\mathrm{H4}}\ \mathbf{d}_{\mathrm{O1}}\ \mathbf{d}_{\mathrm{C1}}\ \mathbf{d}_{\mathrm{H1}}$ *Fourth neighbour*

$\mathbf{d}(5) = \mathbf{d}_{\mathrm{C2}}\ \mathbf{d}_{\mathrm{H4}}\ \mathbf{d}_{\mathrm{O1}}\ \mathbf{d}_{\mathrm{C1}}\ \mathbf{d}_{\mathrm{H1}}\ \mathbf{d}_{\mathrm{H2}}$ *Fifth neighbour*

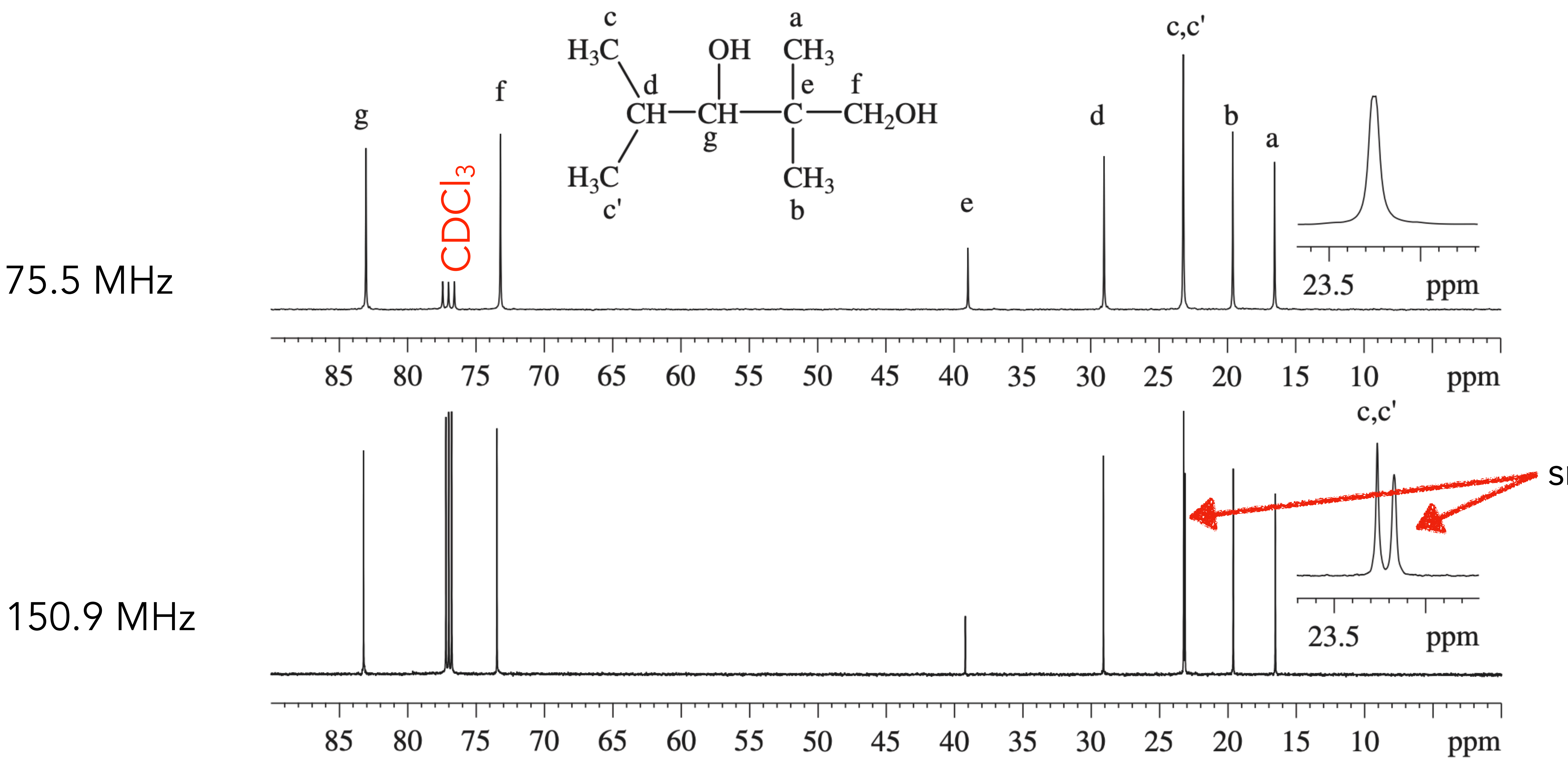
S. Das et al., J. Chem. Phys. (2026)

Exercise 5: ^{13}C Chemical shifts of 2,2,4-trimethyl-1,3-pentanediol

The SMILES strings of propane, isobutane, and neopentane are CCC, CC(C)C, and CC(C)(C)C, respectively.

Q1. Write the SMILES string for 2,2,4-trimethyl-1,3-pentanediol and verify the structure by entering it into the MolDis-Lab ^{13}C NMR app. The app will first display the ^{13}C chemical shifts estimated using the additivity rule.

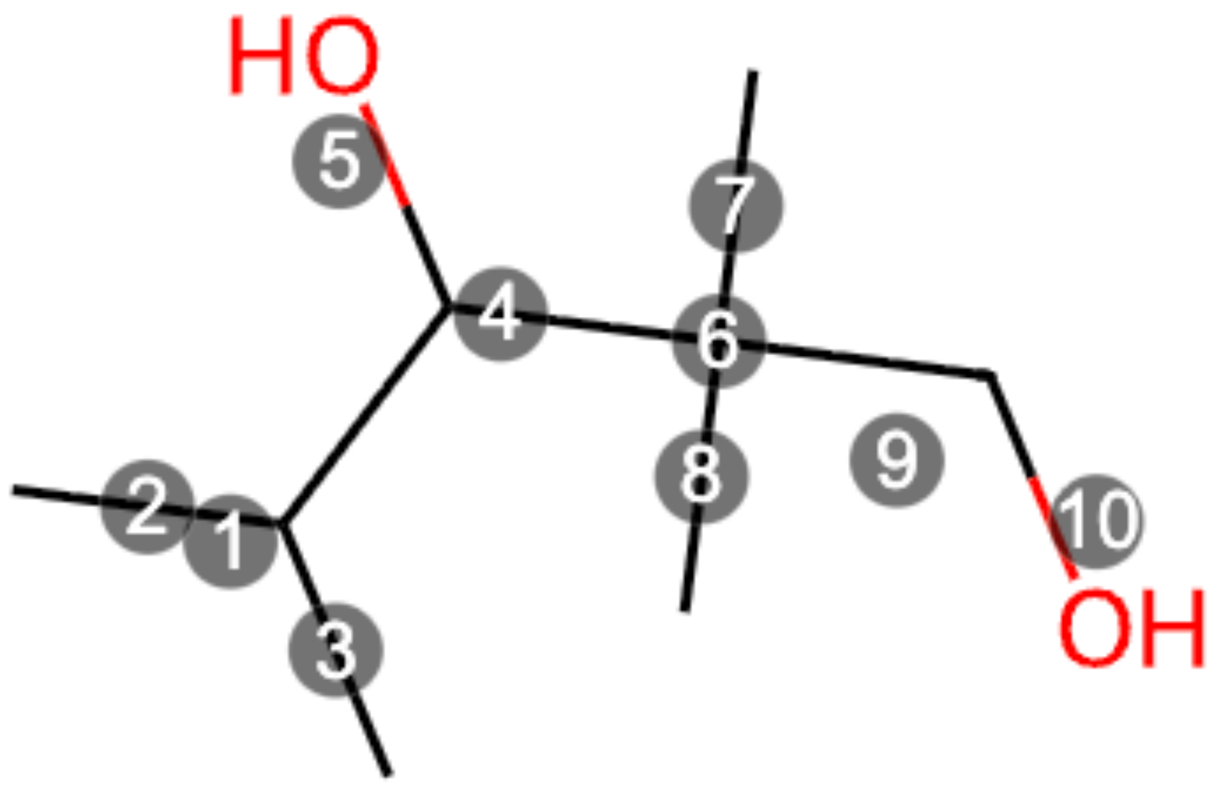
Q2. Next, click **`Predict ¹³C shifts using KRR-ML`** to obtain predictions from a pre-trained kernel ridge regression (KRR) machine-learning model, and compare these values with the reference data provided below.

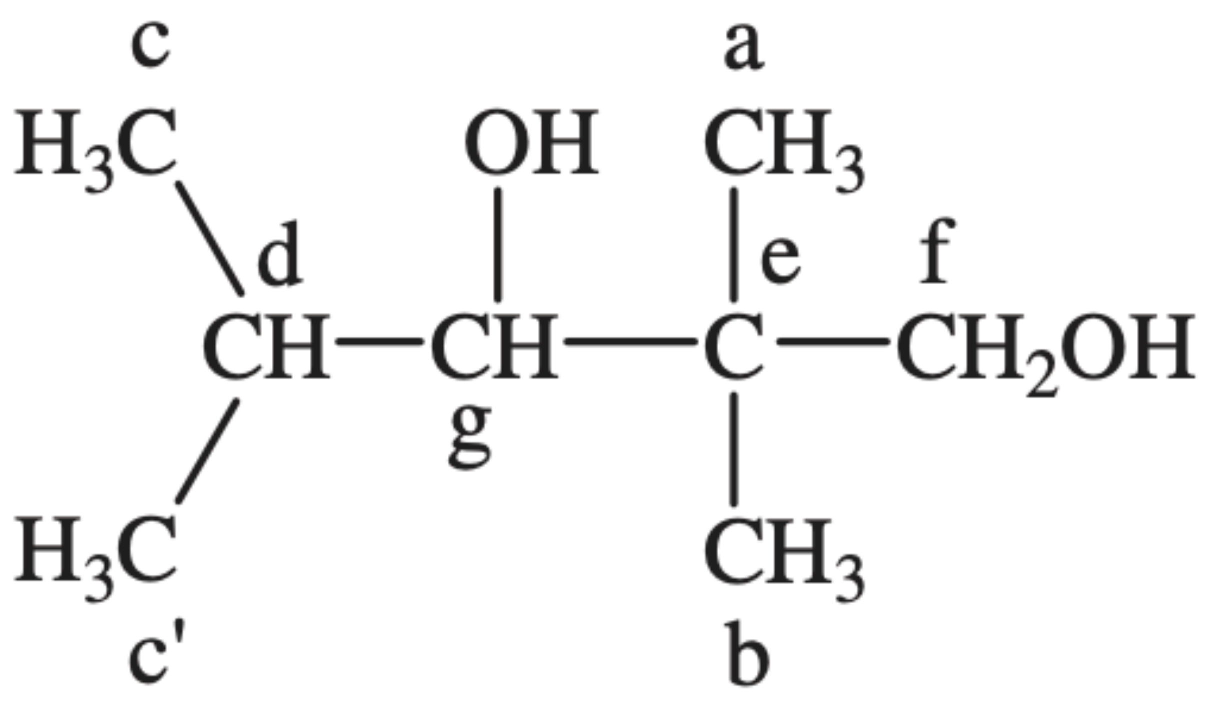


- The structural descriptor used to train the KRR-ML is based on molecular geometries calculated with universal force field (UFF).
- For a new query, MolDis-Lab generates such a structure on the fly from a SMILES string.

R. M. Silverstein et al., *Spectrometric identification of organic compounds* (2015)

17





^{13}C Nuclei	Exp.	Emp. Model	ML
a (7)	19-20	23.2	18.3
b (8)	22	23.2	22.6
c (2)	23-24	18.9	16.9
c’ (3)	23-24	18.9	19.8
d (1)	30-32	32	41.7
e (6)	37-39	36.8	42.3
f (9)	72-74	74.3	71.8
g (4)	83-85	86.8	79.5

Q1. Why is the empirical model unable to differentiate the inequivalent ^{13}C nuclei a (7) and b (8)?

Q2. What physical effects lead to very close chemical shift values for the inequivalent nuclei c (2) and c′ (3) in the experimental spectrum?

Q3. Why does the ML model predict a larger separation for the atom pairs a (7)/b (8) and c (2)/c′ (3) compared to experiment?

Exercise 6: Structure assignment using computed ^{13}C Chemical shifts

Two candidate structures (1 and 2), shown below, are proposed for a compound whose experimental ^{13}C NMR spectrum exhibits four resonances (four chemically non-equivalent ^{13}C nuclei) at 23, 34, 66, and 110 ppm.

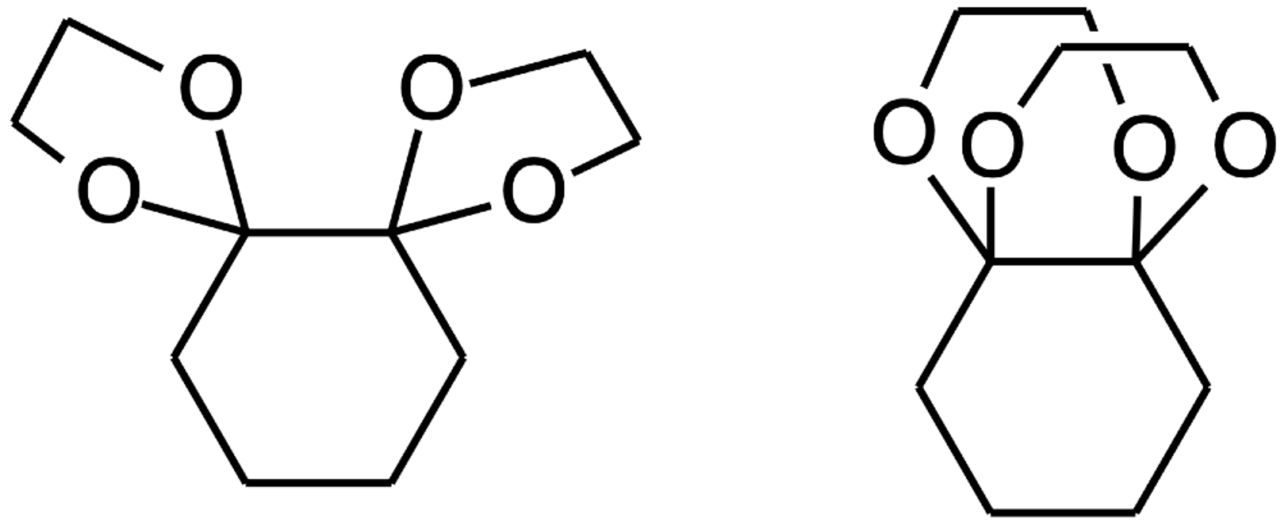
Q1. The SMILES strings of both structures are as follows.

Structure 1: **`C12(OCCO2)C2(OCCO2)CCCC1`**, Structure 2: **`C123C(OCCO2)(OCCO3)CCCC1`**

Enter each SMILES string into MolDis-Lab and estimate the ^{13}C chemical shifts using the additivity (empirical) model. Tabulate the predicted shifts for both structures.

Q2. Use the ML model in MolDis-Lab to predict the ^{13}C chemical shifts for both structures. Compare the empirical and ML predictions with the experimental values and determine which structure provides the better match. To quantify the comparison, you may compute the mean absolute error (MAE) and standard deviation of the errors (SDE).

Q3. Perform geometry optimizations of both structures and calculate the ^{13}C shielding tensors using ORCA. Use the computed isotropic shieldings of TMS as a reference and estimate the corresponding chemical shifts. Compare the DFT-based shifts with those obtained from the empirical and ML models. Do the DFT results support the same structural assignment? Comment on the agreement or discrepancies among the three approaches.



1 **2**

L. J. Tilley et al., *J. Chem. Educ. 79, 593* (2002)

Structure	Type	Emp. Model	ML
1	CH_2	21.4	21.3 (21.2, 21.3)
	CH_2	36.4	28.5 (26.5, 33.5)
	CH_2	60.8	66.6 (65.9, 66.2, 66.4, 67.8)
	C	120.1	123.5 (121.4, 125.6)
2	CH_2	21.4	22.0 (20.6, 23.4)
	CH_2	36.4	32.4 (34.0, 30.8)
	CH_2	62.1	66.4 (63.9, 65.8, 67.0, 68.8)
	C	121.4	111.6 (111.0, 112.1)

Symmetry averaging must be performed after ML prediction (or prediction based on one structure) and before comparison with experiment.

Experimental peaks at 23, 34, 66, and 110 ppm

```python
import numpy as np

exp = np.array([23.0, 34.0, 66.0, 110.0]) # Experimental 13C peaks (ppm)

# Structure 1
emp_1 = np.array([21.4, 36.4, 60.8, 120.1])
ml_1  = np.array([21.3, 28.5, 66.6, 123.5])

# Structure 2
emp_2 = np.array([21.4, 36.4, 62.1, 121.1])
ml_2  = np.array([22.0, 32.4, 66.4, 111.6])

mae, sde = np.mean(np.abs(emp_1-exp)), np.std(emp_1-exp)
print(f"{'Empirical, structure 1':25s}  MAE = {mae:5.1f} ppm   STD = {sde:5.1f} ppm")
mae, sde = np.mean(np.abs(emp_2-exp)), np.std(emp_2-exp)
print(f"{'Empirical, structure 1':25s}  MAE = {mae:5.1f} ppm   STD = {sde:5.1f} ppm")

mae, sde = np.mean(np.abs(ml_1-exp)), np.std(ml_1-exp)
print(f"{'ML, structure 1':25s}  MAE = {mae:5.1f} ppm   STD = {sde:5.1f} ppm")
mae, sde = np.mean(np.abs(ml_2-exp)), np.std(ml_2-exp)
print(f"{'ML, structure 2':25s}  MAE = {mae:5.1f} ppm   STD = {sde:5.1f} ppm")
```

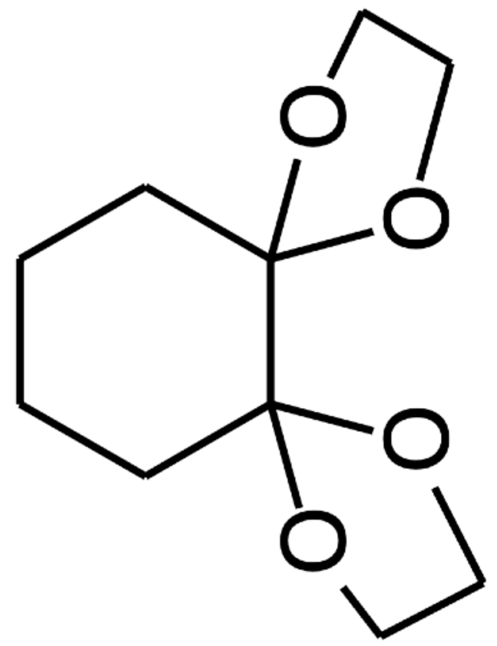
```
Empirical, structure 1     MAE =   4.8 ppm   STD =   5.7 ppm
Empirical, structure 2     MAE =   4.7 ppm   STD =   5.7 ppm
ML, structure 1            MAE =   5.3 ppm   STD =   7.1 ppm
ML, structure 2            MAE =   1.2 ppm   STD =   1.2 ppm
```

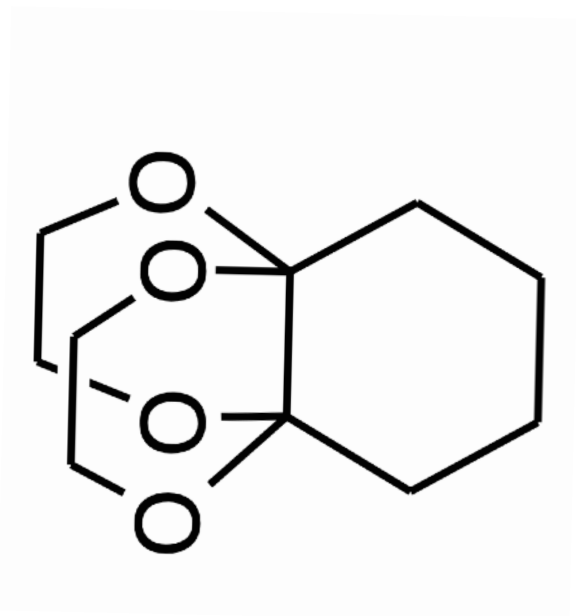
Empirical model based on connectivity fails

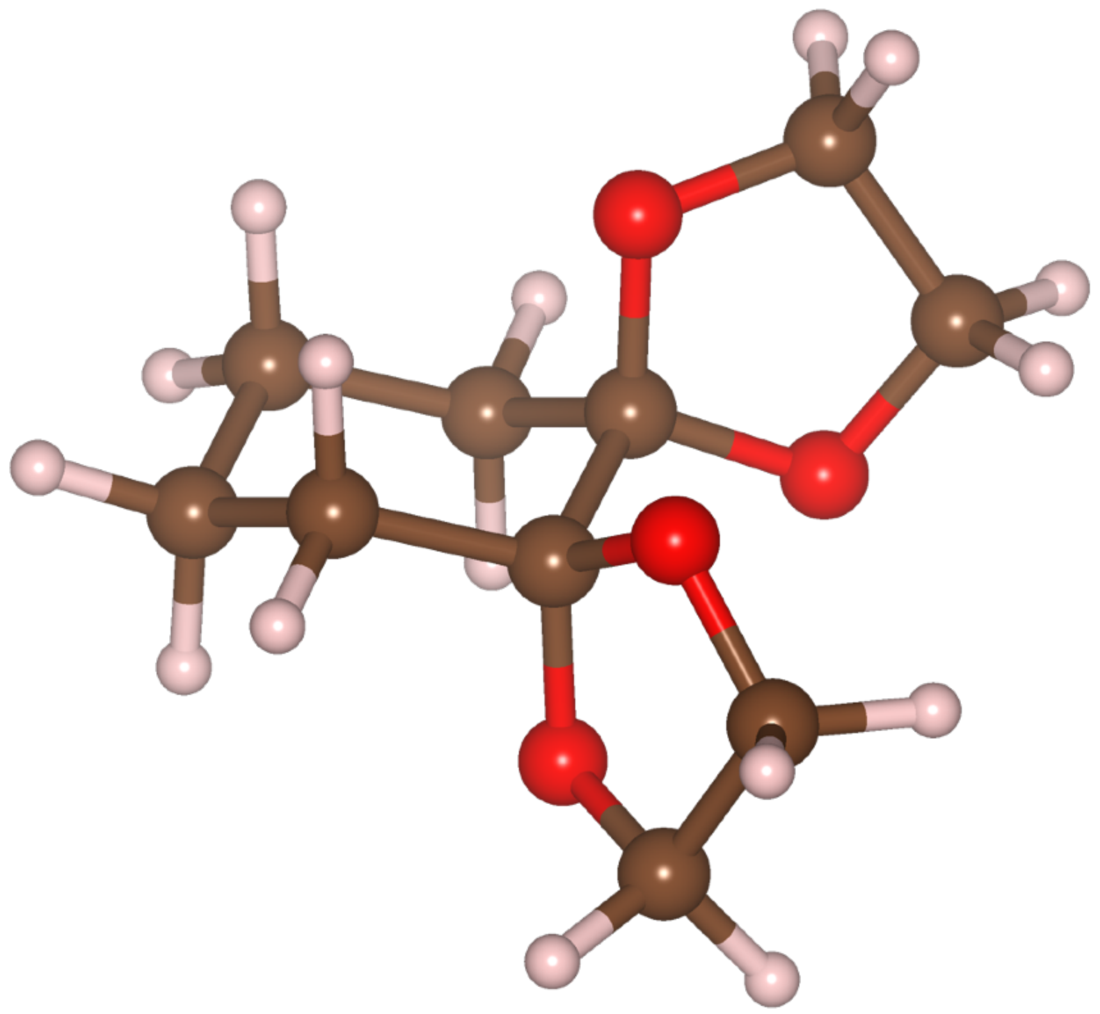
ML suggests structure 2 (small deviation)

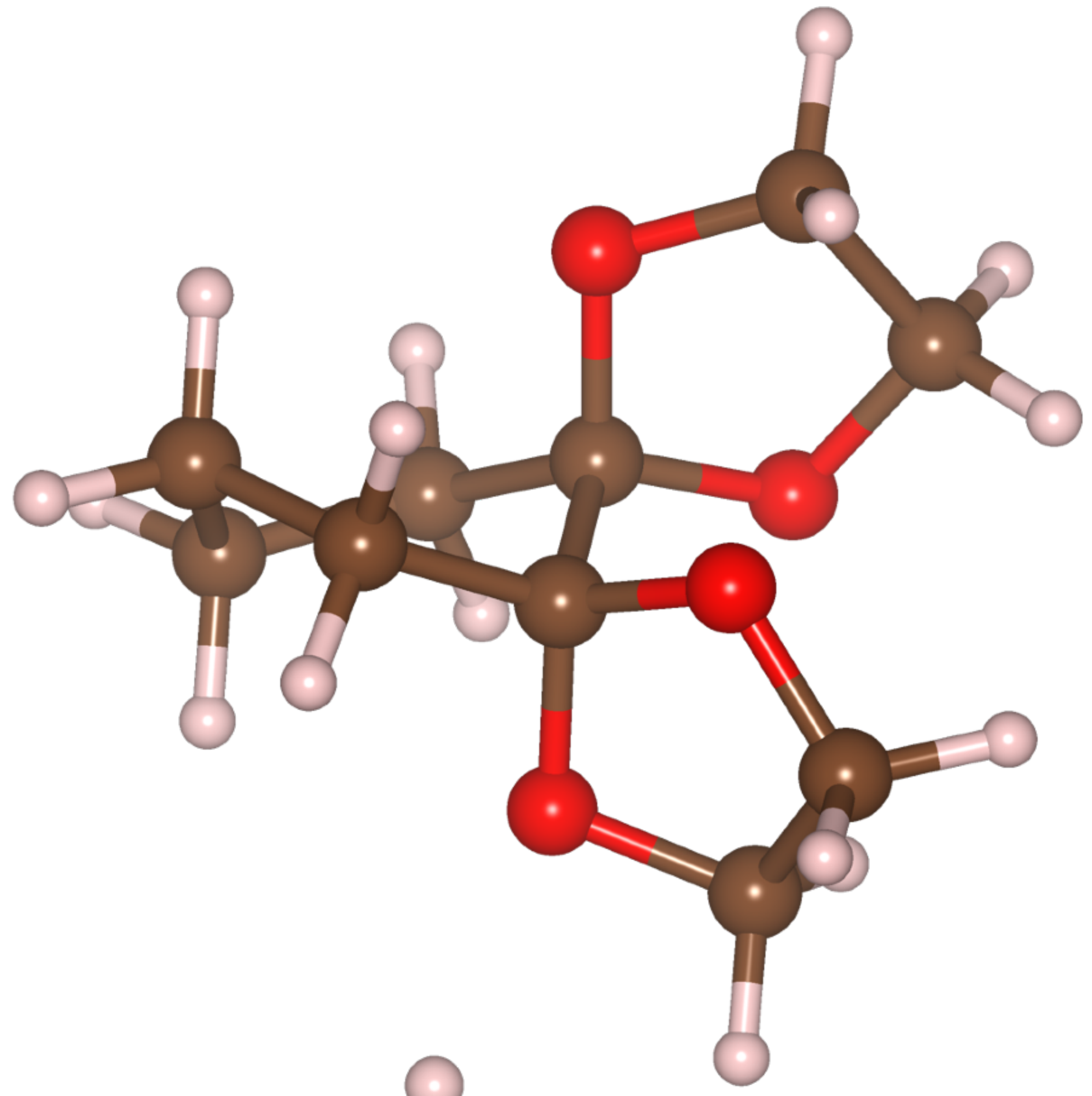
Recall that ML prediction is based on UFF geometry

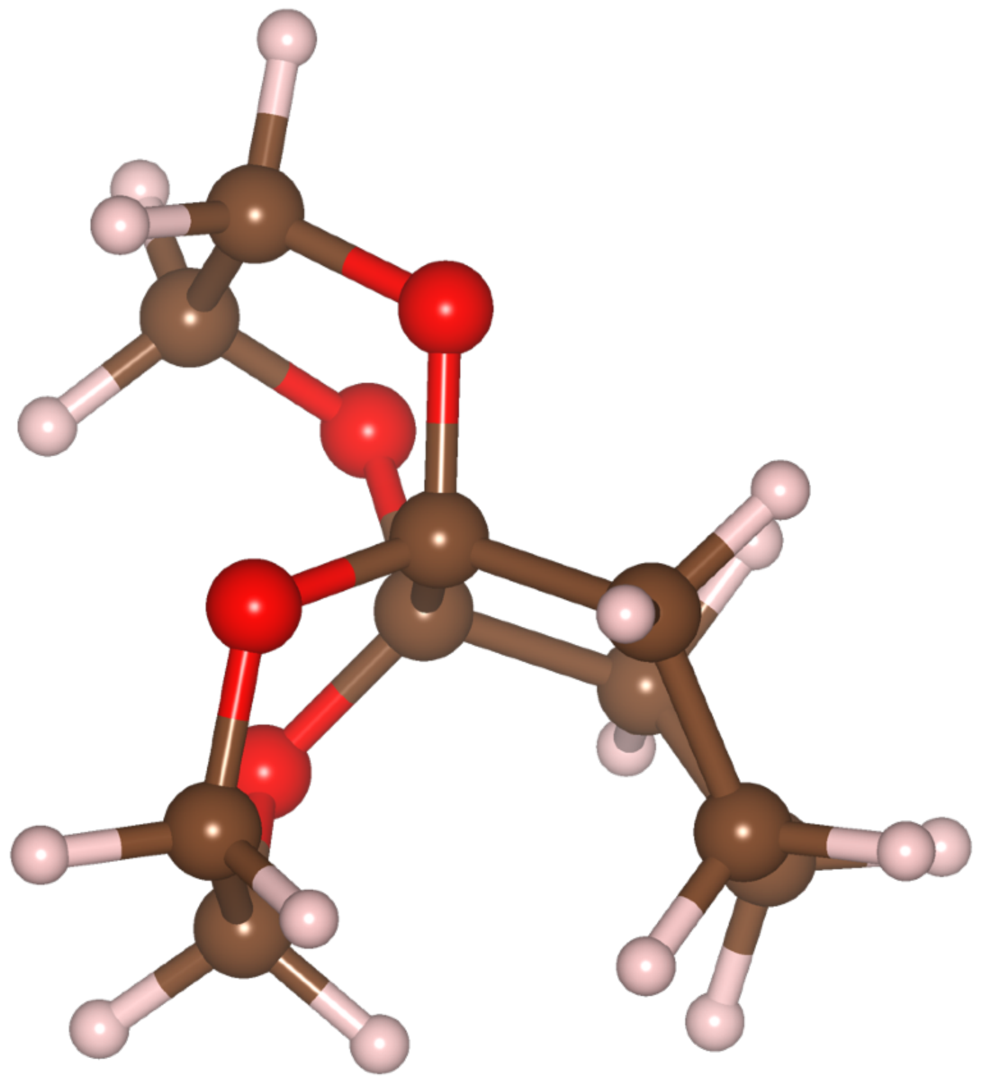
Force field and DFT geometries differ significantly

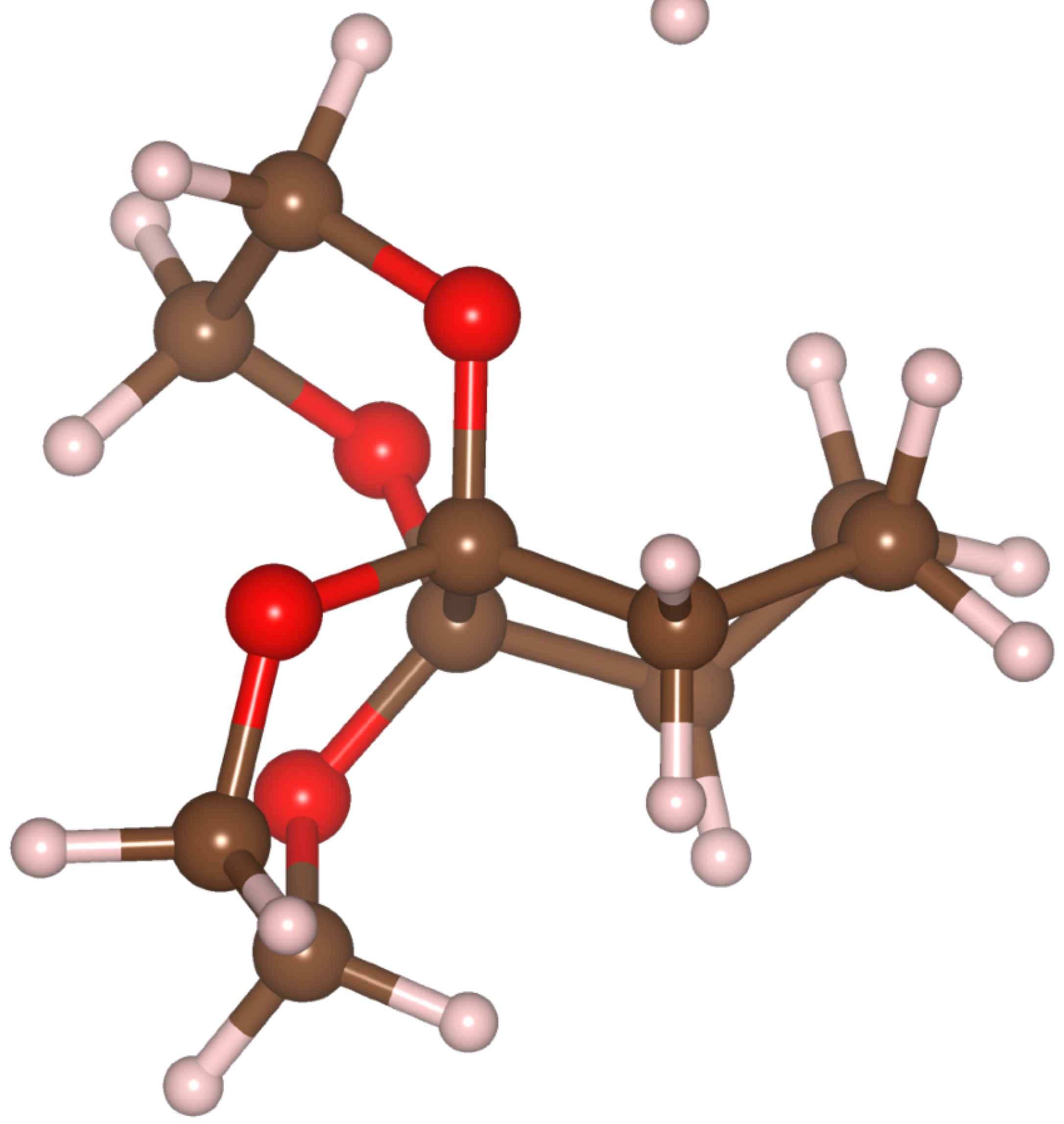












Force field (UFF)

DFT (B3LYP/6-31+G(d,p) D4)

Structure	Type	Emp. Model	ML	DFT
1	CH_2	21.4	21.3 (21.2, 21.3)	24.2 (24.3, 24.1)
	CH_2	36.4	28.5 (26.5, 33.5)	37.2 (37.5, 24.1)
	CH_2	60.8	66.6 (65.9, 66.2, 66.4, 67.8)	67.5 (68.1, 67.2, 68.0, 66.5)
	C	120.1	123.5 (121.4, 125.6)	113.5 (113.2, 113.8)
2	CH_2	21.4	22.0 (20.6, 23.4)	21.5 (20.7, 22.4)
	CH_2	36.4	32.4 (34.0, 30.8)	32.5 (31.9, 33.2)
	CH_2	62.1	66.4 (63.9, 65.8, 67.0, 68.8)	60.7 (57.8, 62.0, 59.3, 63.7)
	C	121.1	111.6 (111.0, 112.1)	99.3 (96.1, 102.6)

Experimental peaks at 23, 34, 66, and 110 ppm

```python
str1_DFT=[113.167,68.073,67.245,113.783,67.996,66.525,37.465,24.31,24.127,36.928]

a=np.mean([24.31,24.127])
b=np.mean([37.465,36.928])
c=np.mean([68.073,67.245,67.996,66.525])
d=np.mean([113.167,113.783])

print(a,b,c,d)

dft_1=np.array([a,b,c,d])
mae, sde = np.mean(np.abs(dft_1-exp)), np.std(dft_1-exp)
print(f"{'DFT, structure 1':25s}  MAE = {mae:5.1f} ppm   STD = {sde:5.1f} ppm")
```

```
24.2185 37.1965 67.45974999999999 113.475
DFT, structure 1          MAE =   2.3 ppm   STD =   1.0 ppm
```

DFT suggests structure 1 (small deviation), which is correct

```python
str2_DFT=[96.103,102.581,57.78,61.96,59.313,63.731,31.878,20.677,22.365,33.161]

a=np.mean([20.677,22.365])
b=np.mean([31.878,33.161])
c=np.mean([57.78,61.96,59.313,63.731])
d=np.mean([96.103,102.581])

print(a,b,c,d)

dft_2=np.array([a,b,c,d])
mae, sde = np.mean(np.abs(dft_2-exp)), np.std(dft_2-exp)
print(f"{'DFT, structure 2':25s}  MAE = {mae:5.1f} ppm   STD = {sde:5.1f} ppm")
```

```
21.521 32.5195 60.696 99.342
DFT, structure 2          MAE =   4.7 ppm   STD =   3.8 ppm
```

- Low MAE and SDE between the experiment and the model are not sufficient.
- Agreement with multiple (^{1}H, IR, etc.) experiments is necessary.
- High-level theoretical modeling, combined with conformational sampling, can provide a reliable reference.

L. J. Tilley et al., *J. Chem. Educ. 79, 593* (2002)

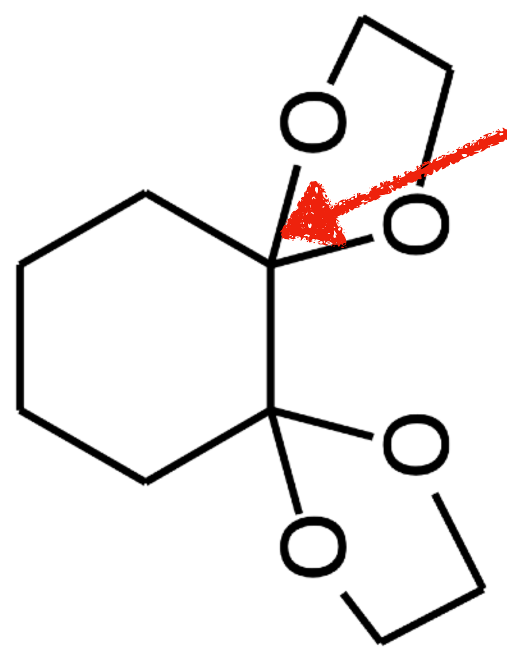
Q4. Let's diagnose the source of ML error. The ML model uses descriptors computed from a UFF-optimized geometry. How can one determine whether discrepancies between ML-predicted and experimental ^{13}C chemical shifts arise primarily from

(i) inaccuracies in the UFF geometry used to generate the descriptors, or

(ii) residual model error due to limitations of the ML training set or model architecture?

To address this, perform DFT NMR shielding calculations on the UFF geometries of Structures 1 and 2 (without further geometry optimization) and compare the resulting chemical shifts with experiment.

Exp. shifts (ppm)	ML (UFF geometry)	DFT (UFF geometry)	DFT (DFT geometry)
23	21.3	25.8	24.2
34	28.5	39.9	37.2
66	66.6	65.7	67.5
110	123.5	112.7	113.5
MAE	5.3	2.9	2.3



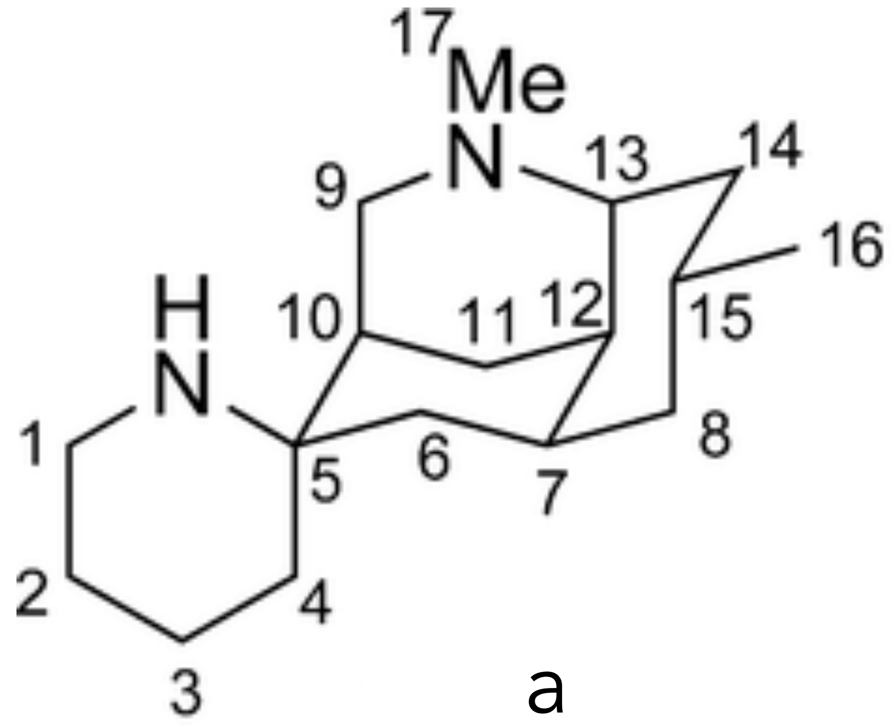
Training set lacks such C environment

2.4 ppm increase in error due to ML model limitations

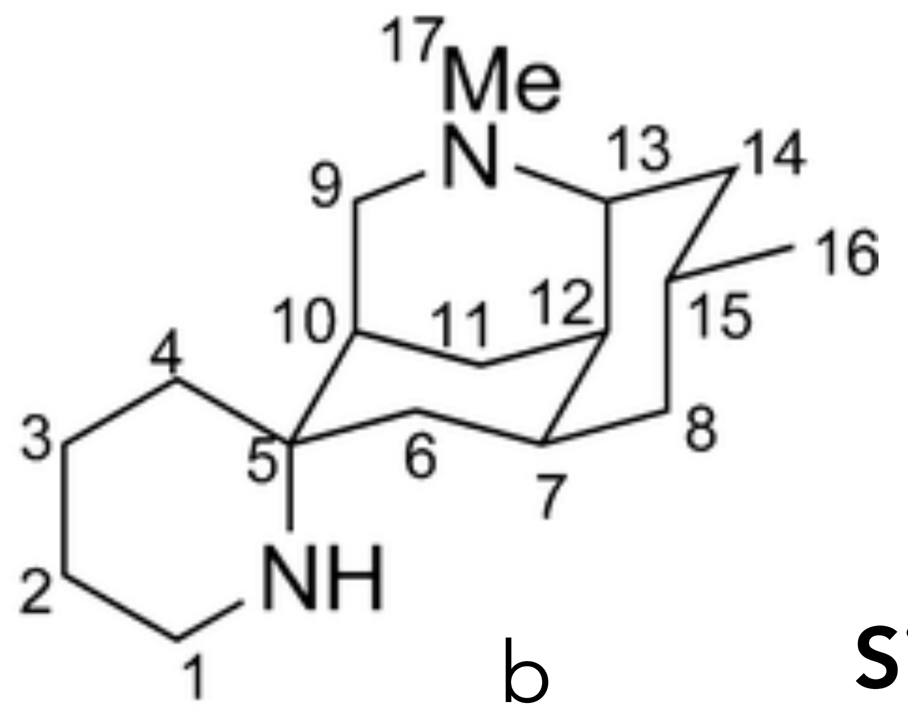
0.6 ppm increase in error due to geometry quality (UFF vs. DFT)

Error in DFT (method error, missing conformer sampling)

Exercise 7: Statistical vs probabilistic (DP4) error metrics for structure assignment



a



b

position	^{13}C calcd 15a	15b	^{13}C expt
1	40.56	40.97	41.0
2	27.05	28.56	26.3
3	22.15	22.53	20.9
4	34.93	37.92	34.6
5	58.27	58.34	56.1
6	41.39	39.27	40.0
7	36.06	34.78	34.5
8	41.01	41.11	41.9
9	56.48	57.31	58.5
10	40.18	41.75	37.4
11	33.74	32.35	32.5
12	39.09	39.35	36.9
13	63.03	63.21	65.1
14	39.54	39.5	40.0
15	24.46	24.43	22.0
16	24.25	24.33	23.0
17	41.07	41.39	43.4

Simple statistical comparison (MAE, SD): The structure of nankakurine was originally assigned as b. Using the data below, compute the mean absolute error (MAE) and standard deviation (SD) of the error between the calculated ^{13}C chemical shifts for structures a and b and the experimental values.

Based on these statistical metrics alone, which structure appears to be the most likely?

Note: In this step, calculated and experimental shifts are compared directly, without any scaling.

S. G. Smith, J. M. Goodman, *J. Am. Chem. Soc.* 132, 12946 (2010).

Probabilistic assignment using the DP4 method: In the DP4 approach, the Bayesian probability that a candidate structure i (with N nuclei) corresponds to the experimental spectrum is given by $P\left(i\,|\,\delta_1, \delta_2, \cdots, \delta_N\right) = \dfrac{\Pi_{k=1}^{N}\left[1 - T_\nu\left(t_k\right)\right]}{\sum_{j=1}^{m}\Pi_{k=1}^{N}\left[1 - T_\nu\left(t_k\right)\right]}$, where $T_\nu\left(t\right)$ is the cumulative distribution function of Student's t-distribution with ν degrees of freedom and t_k is the standardized error for nucleus k given by $t_k = \dfrac{|\left(\delta^i_{\text{scaled},k} - \delta_{\exp,k}\right) - \mu|}{\sigma}$.

Before computing DP4 probabilities, the calculated shifts must be linearly corrected. A linear regression of calculated shifts (y-axis) against experimental shifts (x-axis) is performed $\delta_{\text{calc}} = m\,\delta_{\text{exp}} + b$. The regression is then inverted to map the calculated shifts back onto the experimental scale: $\delta_{\text{scaled}} = \dfrac{\delta_{\text{calc}} - b}{m}$. This step removes systematic offset and slope errors.

Use $\mu = 0$ ppm, $\sigma = 2.306$ ppm, and $\nu = 11.38$ from Smith and Goodman's article.

Perform the linear scaling separately for structures a and b**.** Compute the DP4 probabilities using the scaled errors. Decide which structure is most consistent with the experimental ^{13}C NMR spectrum.

S. G. Smith, J. M. Goodman, *J. Am. Chem. Soc.* 132, 12946 (2010).

^{13}C NMR (from Smith and Goodman)

	a	b
Mean absolute error	1.50	1.62
Standard deviation of the error	1.59	1.83
DP4 probability	79.5%	20.5%

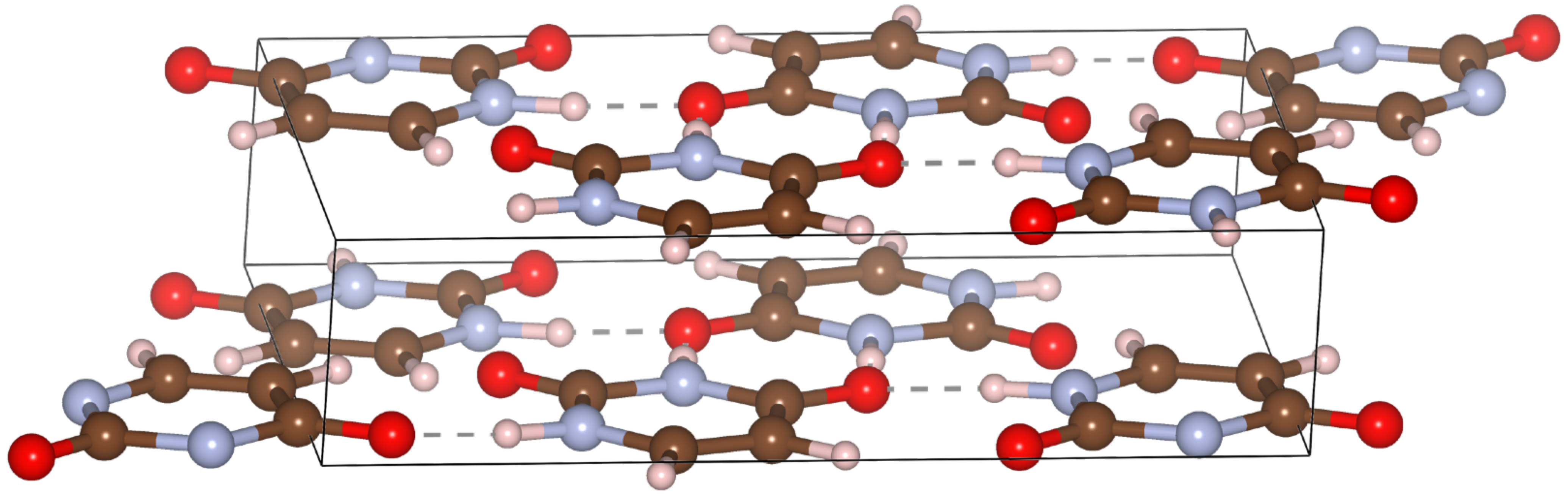
Do your results agree with the values from Smith and Goodman's article? If there is any disagreement, what do you think is the reason? You can also repeat the exercise for the ^{1}H NMR data from Smith and Goodman.

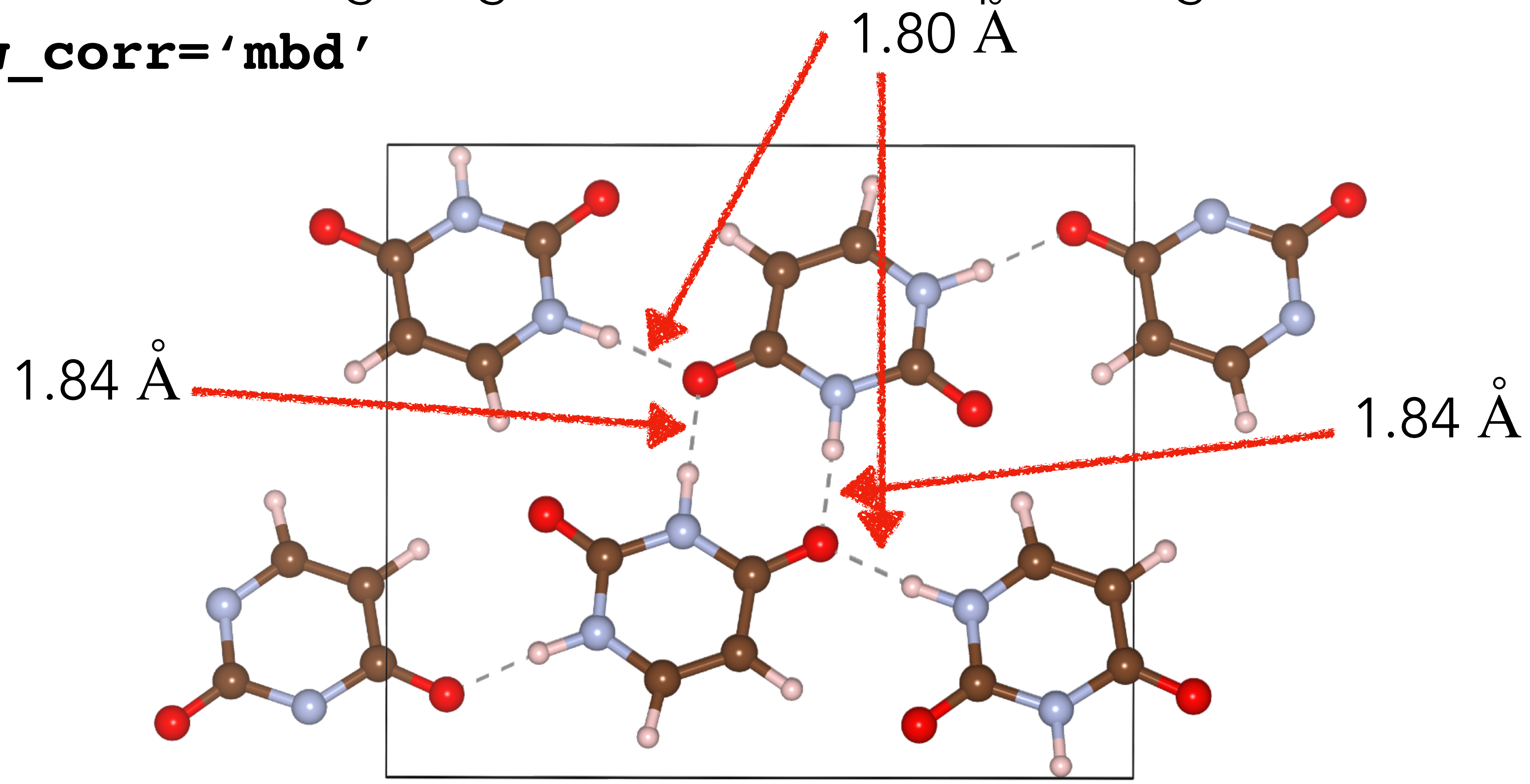
Take-home message: MAE averages errors, and DP4-like scores penalize unlikely errors.

S. G. Smith, J. M. Goodman, *J. Am. Chem. Soc.* 132, 12946 (2010).

Exercise 8: Solid state NMR spectrum of uracil

Input/Output files for the program Quantum Espresso for geometry relaxation of uracil crystal with a guess initial structure are provided in the folder **`nmrworkshop2026/exercises/ex08/opt`**, and **`nmrworkshop2026/solution/ex08/opt.`** The DFT method PBE was used along with many-body dispersion correction to long-range interactions for optimizing the structure of uracil crystal using the keywords **`input_dft = pbe`** and **`vdw_corr='mbd'`**



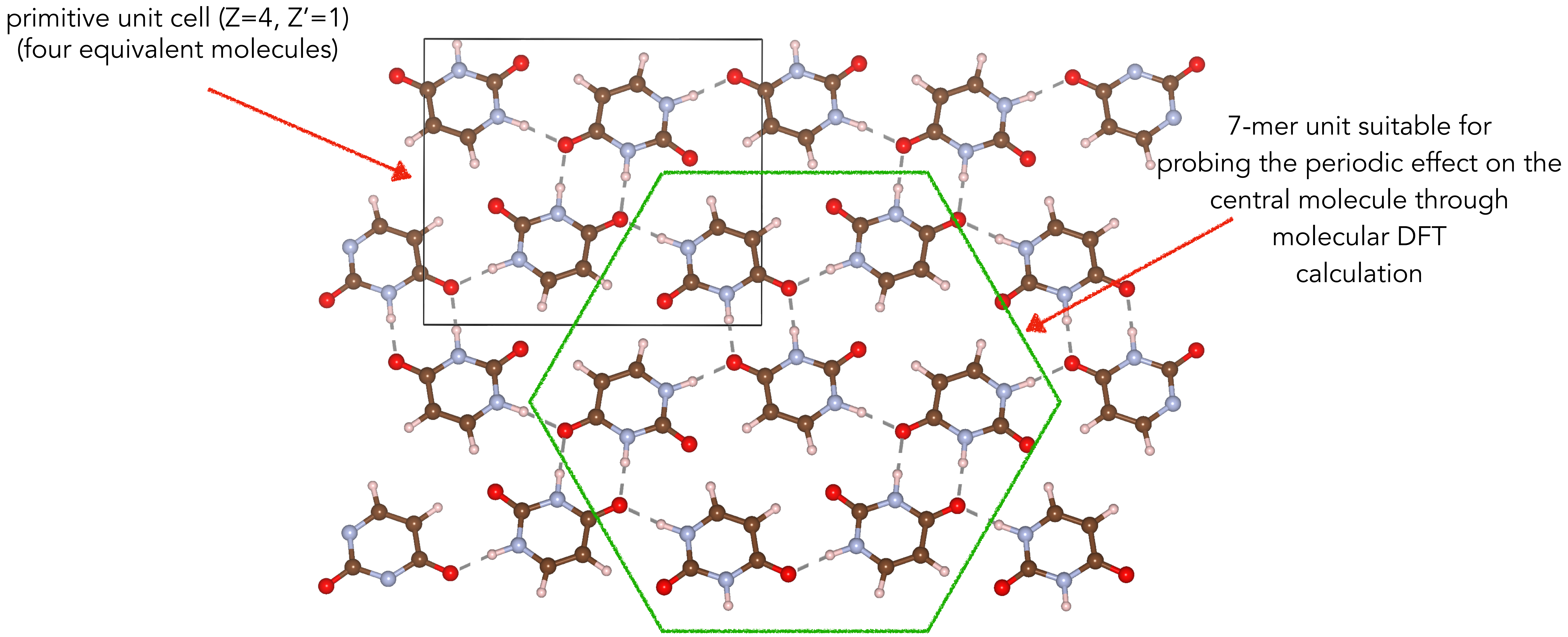


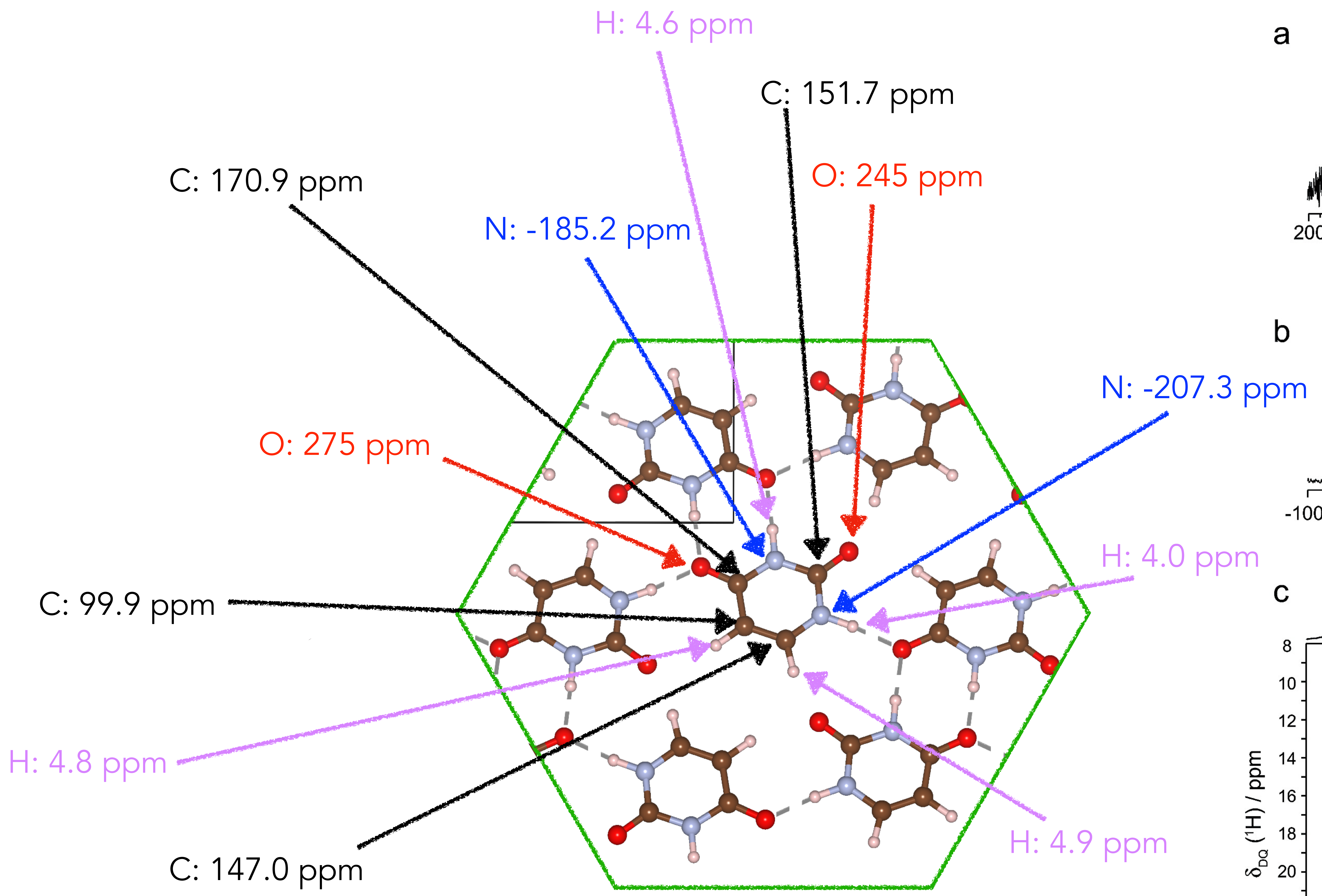
Space group (No. 14): $P2_1/a$

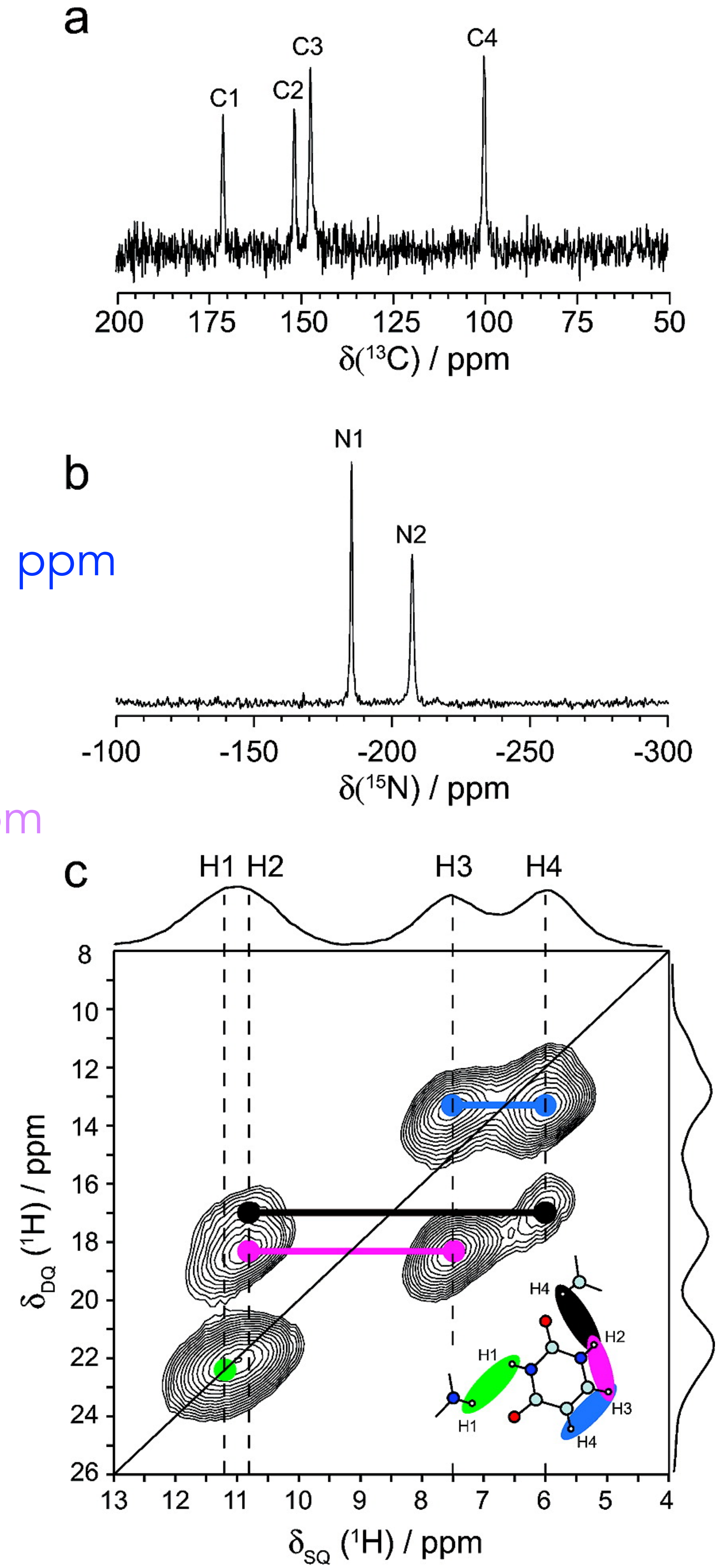
Source	a (Å)	b (Å)	c (Å)	α (deg)	β (deg)	γ (deg)
PBE-MBD	12.10	12.39	3.81	90	120.2	90
Exp.	11.94	12.38	3.66	90	120.5	90

R. F. Stewart, L. H. Jensen, *Acta Crystallographica*, 23, 1102, (1967)

Q1. Which nuclei in uracil are expected to be most sensitive to crystal packing and hydrogen bonding, and why?







Experimental chemical shifts

R. Uldry, et al., *J. Am. Chem. Soc.*, 130, 945 (2008)

Input/Output files for NMR calculations are provided in the folder **nmrworkshop2026/exercises/ex08/nmr**, and **nmrworkshop2026/solution/ex08/nmr.** This is a two-step process. First is to do a single-point energy calculation (**scf.in**) with the keyword **restart_mode = 'from_scratch'** to save the wavefunctions. In scf.in, we will use the optimized coordinates and lattice vectors taken from opt.out. The keyword **restart_mode = 'from_scratch'** is used to do an NMR calculation (**nmr.in**). This latter part of the calculation is done using Quantum Espresso's extension **gipaw.x**

key sections of the output file (**nmr.out**)

$\sigma_C^{\mathrm{uracil}}$ $\qquad \delta_C = \sigma_C^{\mathrm{TMS}} - \sigma_C^{\mathrm{ethanol}}$

```
 Total NMR chemical shifts in ppm: ---------------------------------------
 (adopting the Simpson convention for anisotropy and asymmetry)-----------

 Atom   1  C    pos: (  0.148060  0.217129 -0.003254)  Total sigma:       14.38
       -25.9104        -54.2685          -0.6696
       -50.3738         -3.7642           1.8066
        -2.1644          3.7688          72.8027

 C    1    anisotropy:   -124.04    eta:   -0.4197
 C    1    sigma_11=      38.37    axis=( -0.627615  0.773482 -0.088454)
 C    1    sigma_22=      73.08    axis=( -0.052115  0.071621  0.996069)
 C    1    sigma_33=     -68.32    axis=(  0.776777  0.629758 -0.004640)

 Atom   2  C    pos: (  0.686360  0.807131  0.270738)  Total sigma:       14.38
       -25.9104        -54.2685          -0.6696
       -50.3738         -3.7642           1.8066
        -2.1644          3.7688          72.8027

 C    2    anisotropy:   -124.04    eta:   -0.4197
 C    2    sigma_11=      38.37    axis=( -0.627615  0.773482 -0.088454)
 C    2    sigma_22=      73.08    axis=( -0.052115  0.071621  0.996069)
 C    2    sigma_33=     -68.32    axis=(  0.776777  0.629758 -0.004640)

 Atom   3  C    pos: (  0.186622  0.729259  0.286926)  Total sigma:       14.38
       -25.9104         54.2685          -0.6696
        50.3738         -3.7642          -1.8066
        -2.1644         -3.7688          72.8027
```

As an approximation, we use the reference shielding values adopted in the experimental study. This avoids introducing separate reference calculations, but can contribute to discrepancies.

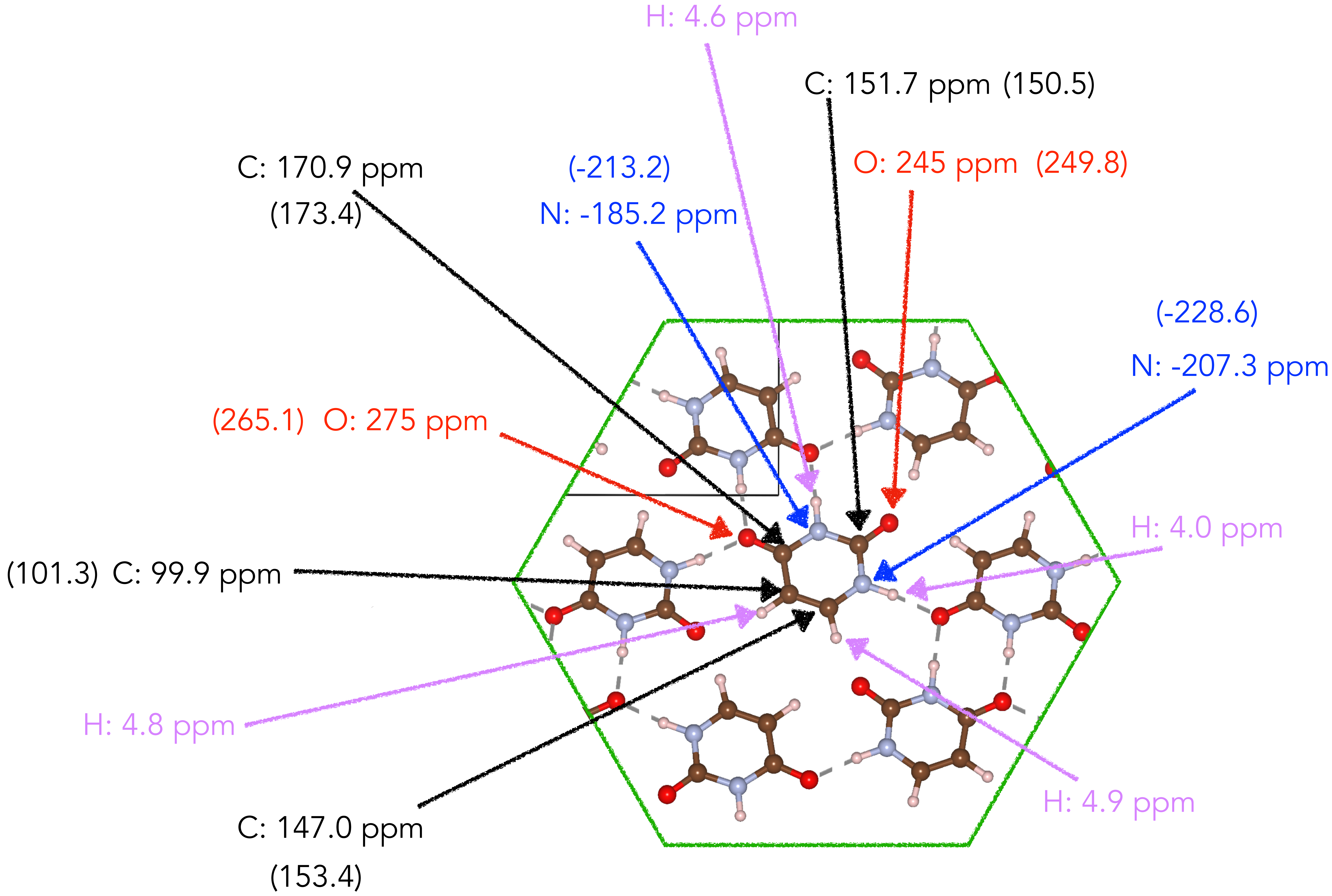
$$^{13}\mathrm{C}: \quad \sigma_C^{\mathrm{TMS}} = 167.8\,\mathrm{ppm}$$

$$^{1}\mathrm{H}: \quad \sigma_H^{\mathrm{TMS}} = 29.7\,\mathrm{ppm}$$

$$^{15}\mathrm{N}: \quad \sigma_N^{\mathrm{nitro\ methane}} = -154.3,\mathrm{ppm}$$

$$^{17}\mathrm{O}: \quad \sigma_O^{\mathrm{water}} = 261.5\,\mathrm{ppm}$$

R. Uldry, et al., *J. Am. Chem. Soc.*, 130, 945 (2008)



Q2. Which nuclei show the largest discrepancies between experiment and calculation?Are these differences more likely due to

(i) residual structural errors in the crystal geometry,

(ii) missing effects such as nuclear motion or temperature averaging,

(iii) limitations of the exchange–correlation functional, or

(iv) the use of experimental reference shielding values rather than reference shieldings computed at the same DFT level?

What did we learn?

- Empirical models capture trends
- ML interpolates learned chemistry
- DFT provides physical grounding but is sensitive to structure
- Probabilistic methods (DP4) encode confidence
- Solid-state NMR requires thinking beyond isolated molecules