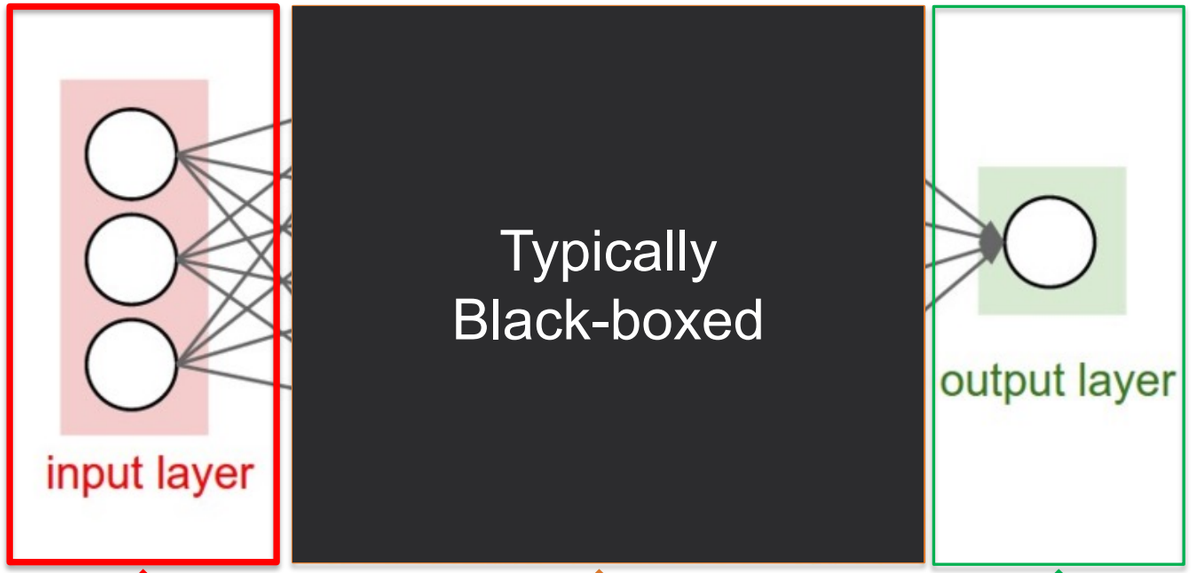
Class 0

Conda,
rdkit, numpy, pandas, matplotlib, seaborn …
Jupyter notebook installation instructions

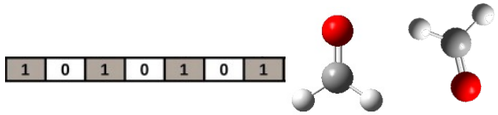
Overview – Machine Learning in Chemistry



Molecular descriptors (Class 1)
Fingerprint and similarity (Class 2)

Input features

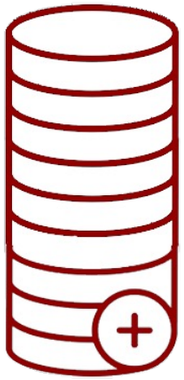
- w/ or w/o 3D coord.
- Descriptors such as graphs, fingerprints, …
- Rotation/permutation 'invariant'

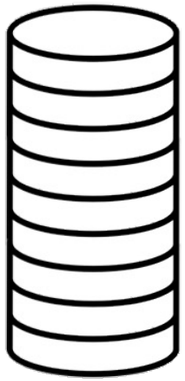


Regression
Multivariate model (Class 3)
Machine learning (Class 4)
Deep learning (Class 5 – Peter? Focusing on message-passing, RL if possible?),
etc.

Various physical/chemical properties (experimental, calculated)

Overview – Machine Learning in Chemistry





Database Size

- Data points > 10^4, 10^5

  → Minimal information is given in the input layer (SMILES, atom, bond feature, connectivity, etc.)

  → Minimal human intervention

  → Deep learning, so that the computer automatically recognizes and learns some 'patterns' that are consistent (sometimes not consistent) with our chemical intuition

- Data points < 10^3

  → Molecular descriptors can be generated (from QM calculations, Python packages, etc.) and chosen based on our intuition

  → Target property = f(chosen molecular descriptors/features)

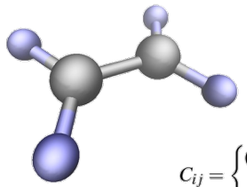
*** May contain oversimplification, the perspective in this slide is not always right.*

Class 1 – Molecular Descriptors

Colorado State University

Features/Descriptors As Input

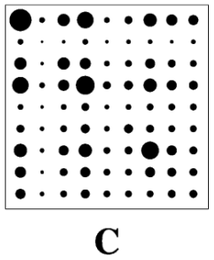
- **Input features using 3D coordinates (Continuous)**



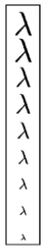
$$C_{ij} = \begin{cases} 0.5\, Z_i^{2.4} & \forall i = j \\ \frac{Z_i Z_j}{|\mathbf{R}_i - \mathbf{R}_j|} & \forall i \neq j. \end{cases}$$

$\mathbf{C} =$

	H	H	C	C	H	H
H	0.5	0.3	2.9	1.5	0.2	0.2
H	0.3	0.5	2.9	1.5	0.2	0.2
C	2.9	2.9	36.9	14.3	1.5	1.5
C	1.5	1.5	14.3	36.9	2.9	2.9
H	0.2	0.2	1.5	2.9	0.5	0.3
H	0.2	0.2	1.5	2.9	0.3	0.5



$\mathbf{C}$



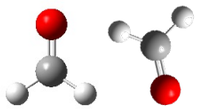
Eigenvalues

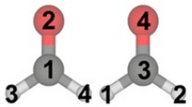
$$C_{ij} = \begin{cases} 0.5\, Z_i^{2.4} & \forall i = j \\ \frac{Z_i Z_j}{|\mathbf{R}_i - \mathbf{R}_j|} & \forall i \neq j. \end{cases}$$

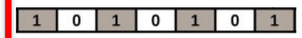
J. Chem. Theory Comput., **2013**, *9*, 3404.

Input features

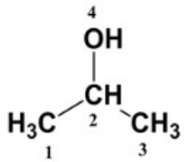
- w/ or w/o 3D coord.
- Descriptors such as graphs, fingerprints, …
- Rotation/permutation 'invariant'







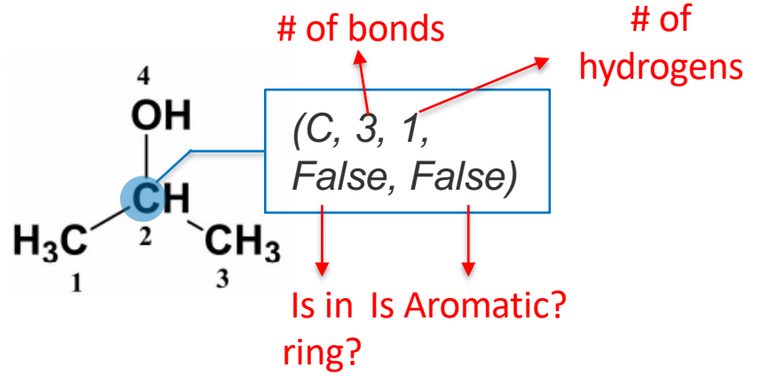
- **Input features w/o 3D coordinates (Discrete)**



$$X_1 = \begin{bmatrix} 6 \\ 3 \\ 4 \\ 0 \end{bmatrix} \quad \cdots \quad X_4 = \begin{bmatrix} 8 \\ 1 \\ 4 \\ 0 \end{bmatrix} \begin{matrix} \text{Atom type} \\ \text{\# of Hs.} \\ \text{\# Valence} \\ \text{Aromaticity} \end{matrix}$$

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

- Atom features



arXiv: 1805.10988v2

In Jupyter Notebook

Topological Descriptors

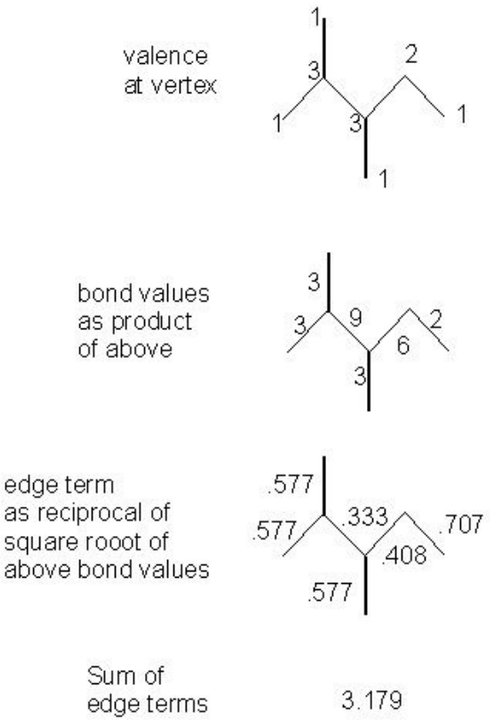
Molecular shape → Real number

- Randić indices

$$\chi_0(G) = (\text{Sum of Deg.'s for vertices } i)^{-1}$$

$$\chi_1(G) = \sum_{edges\ i-j} (\mathrm{Deg}(i)\mathrm{Deg}(j))^{-1/2}$$

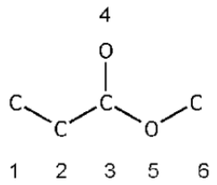
$${}^{h}\chi(G) = \sum_{\text{paths of length } h} (\mathrm{Deg}(i)\mathrm{Deg}(j)\ldots\mathrm{Deg}(h+1))^{-1/h}$$



- Balaban's J index

$$J = \frac{m}{\gamma+1} \sum_{(i,j)\in E(G)} (D_i D_j)^{-1/2},$$

(γ = m – n + 1,
m: # of bonds, n: # of atoms)



G_1

$A(G_1) =$

	1	2	3	4	5	6
1	0	1	0	0	0	0
2	1	0	1	0	0	0
3	0	1	0	1	1	0
4	0	0	1	0	0	0
5	0	0	1	0	0	1
6	0	0	0	0	1	0

$D(G_1) =$

	1	2	3	4	5	6
1	0	1	2	3	3	4
2	1	0	1	2	2	3
3	2	1	0	1	1	2
4	3	2	1	0	2	3
5	3	2	1	2	0	1
6	4	3	2	3	1	0

- Topological polar surface area

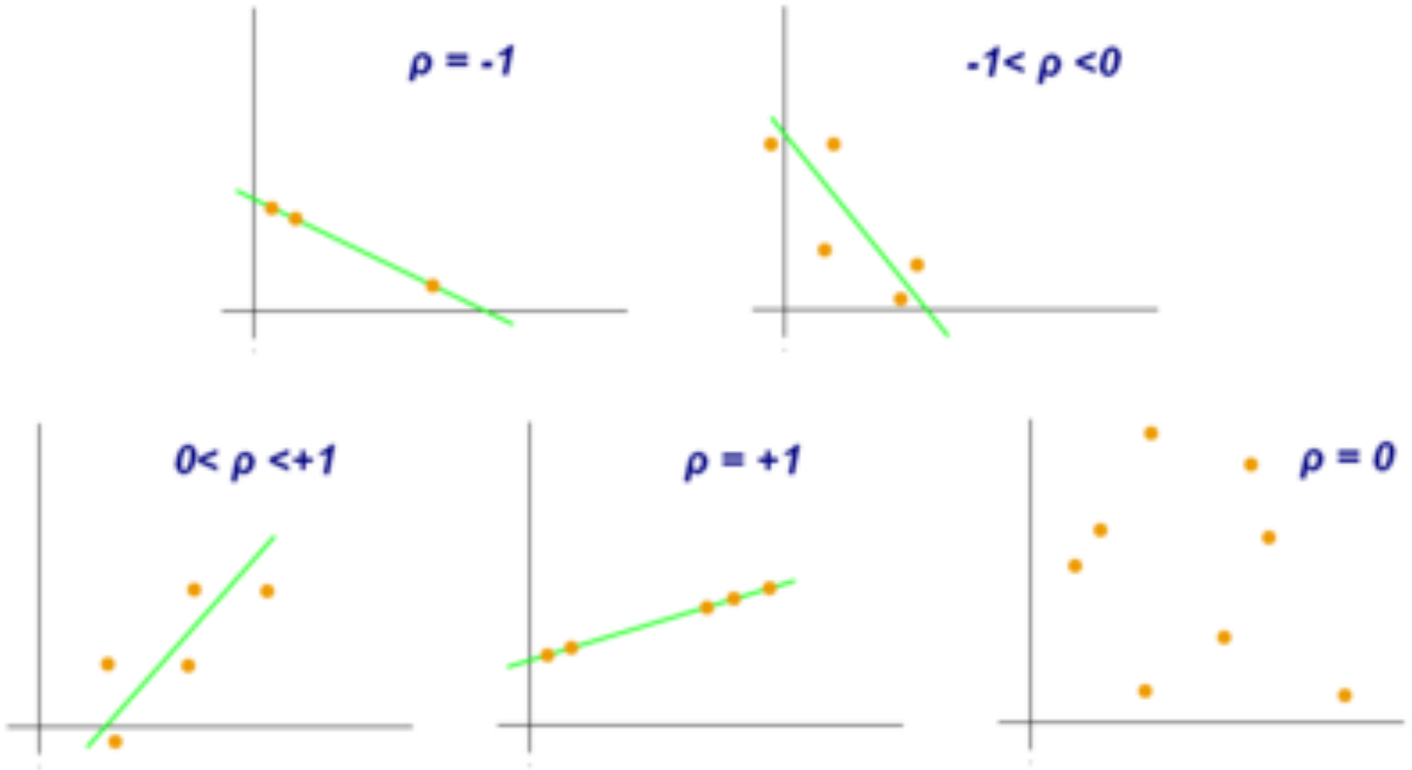
- Bertz complexity index

An index defined to quantify complexity,
extent of branching of a molecule

https://www.rdkit.org/docs/source/rdkit.Chem.GraphDescriptors.html
https://www.rdkit.org/docs/source/rdkit.Chem.rdMolDescriptors.html
And refs therein

Pearson Correlation Coefficient

np.corrcoef



Jupyter Notebook Demo – Find the correlation between Topological descriptors vs. Boiling point

Class 2 – Fingerprint and Similarity

Colorado State University

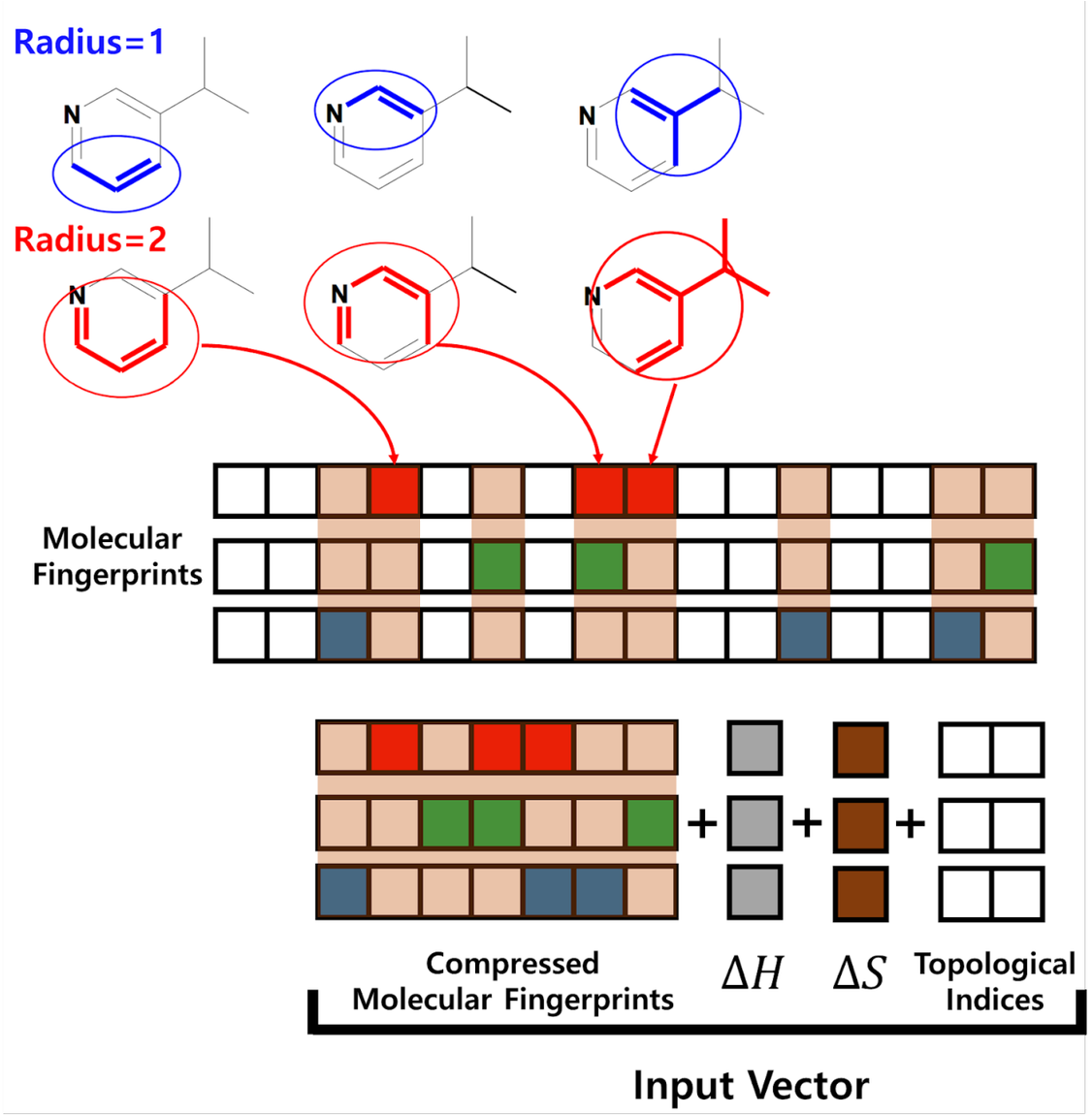
Why Molecular Fingerprints?

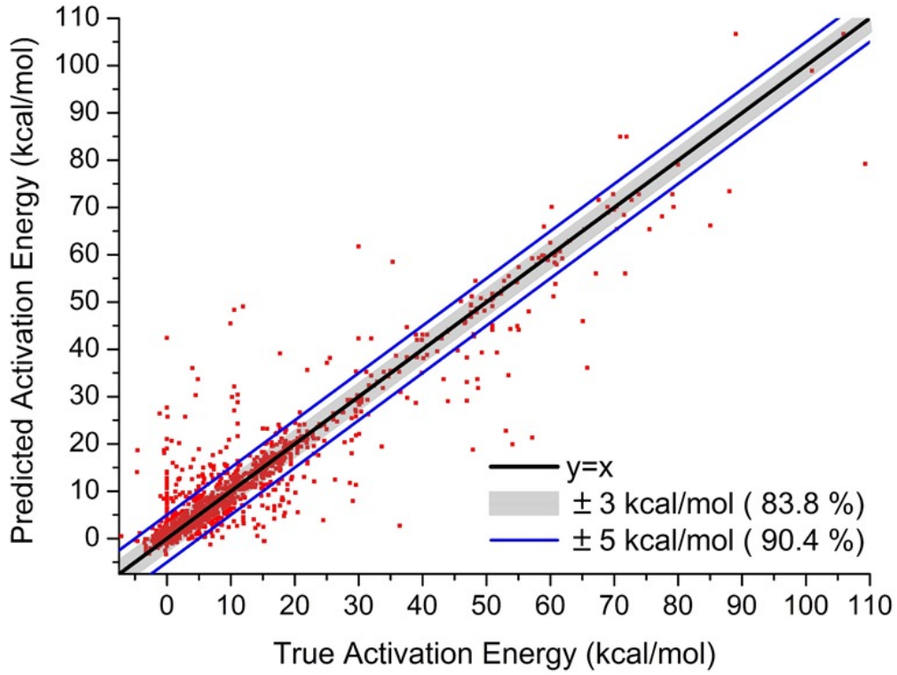
- How can we make a (complete) vector representing structural features of a molecule, without using Cartesian coordinates?

- How can we find the molecules sharing the common substructure(s)?

- How can we evaluate similarity between two molecules?

Morgan Fingerprint





MAE: 1.95 kcal/mol, RMSE: 4.49kcal/mol
R^2: 0.89

Reaction data are obtained from Reaction Mechanism Generator (RMG) Database
12,704 gas-phase reactions related to combustion (80% for training, 20% for test set)

Chem. Eur. J. 2018, 24, 12354-12358.

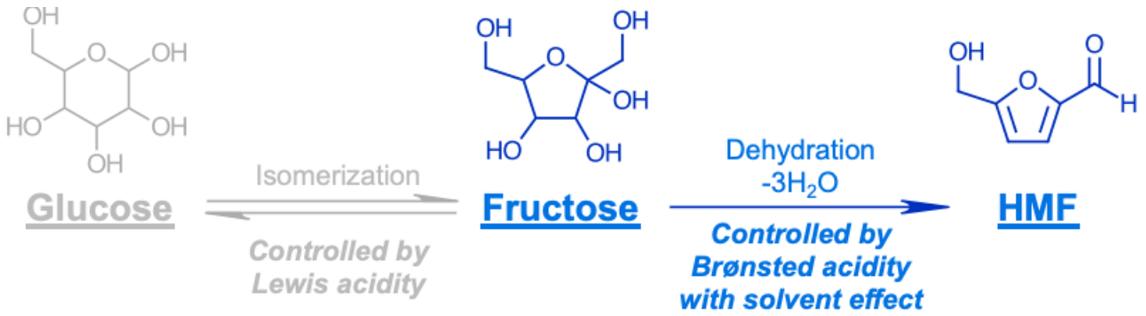
Class 3 – Multivariate Analysis

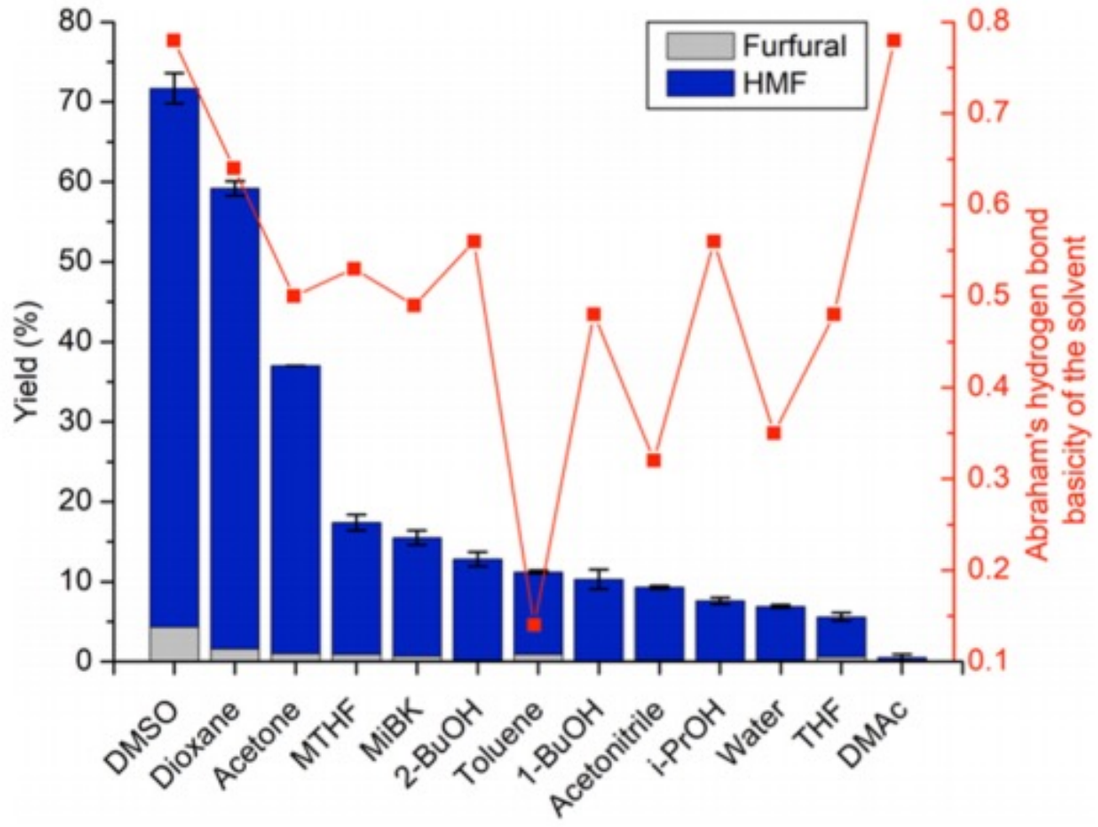
Colorado State University

Multivariate Analysis - Motivation

- How can we choose appropriate descriptors with minimizing human bias (intervention)?
- How can we overcome a limited number of data points (e.g. < 100?)
- How can we gain chemical insights from the model?
  (Although it could be *ad hoc* and subject to change with more data points)

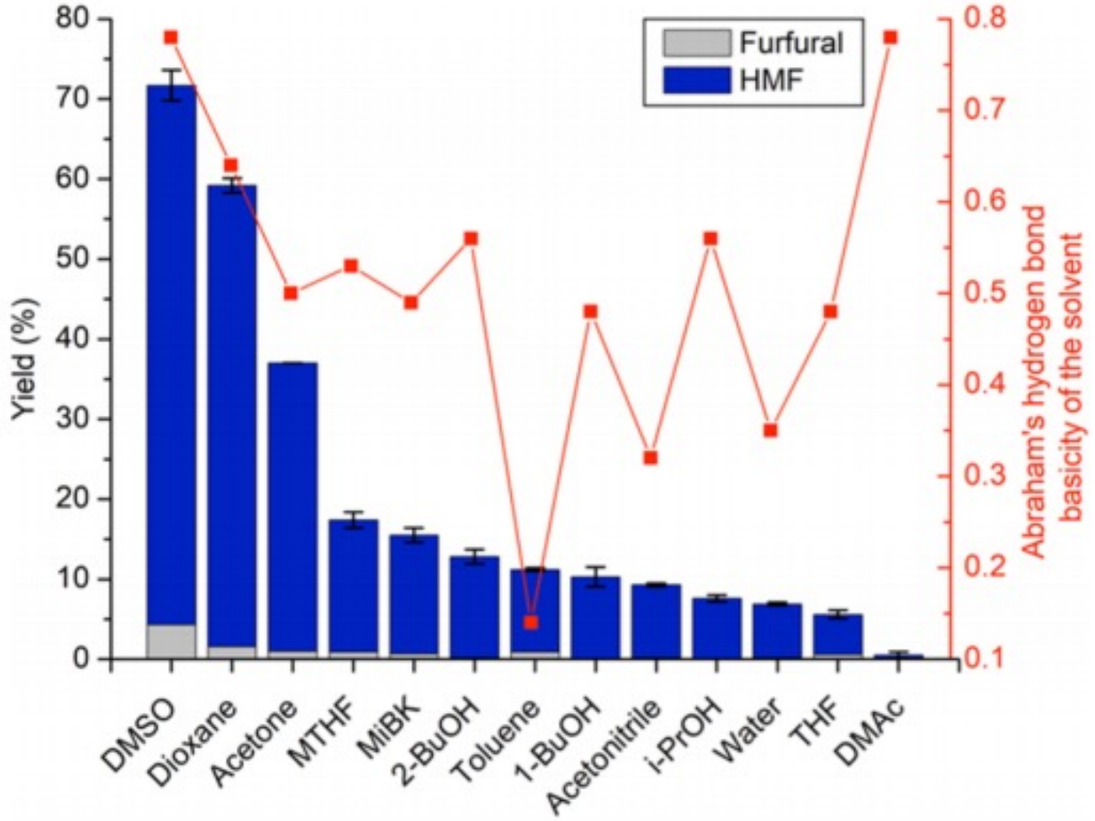
Multivariate Analysis - Motivation





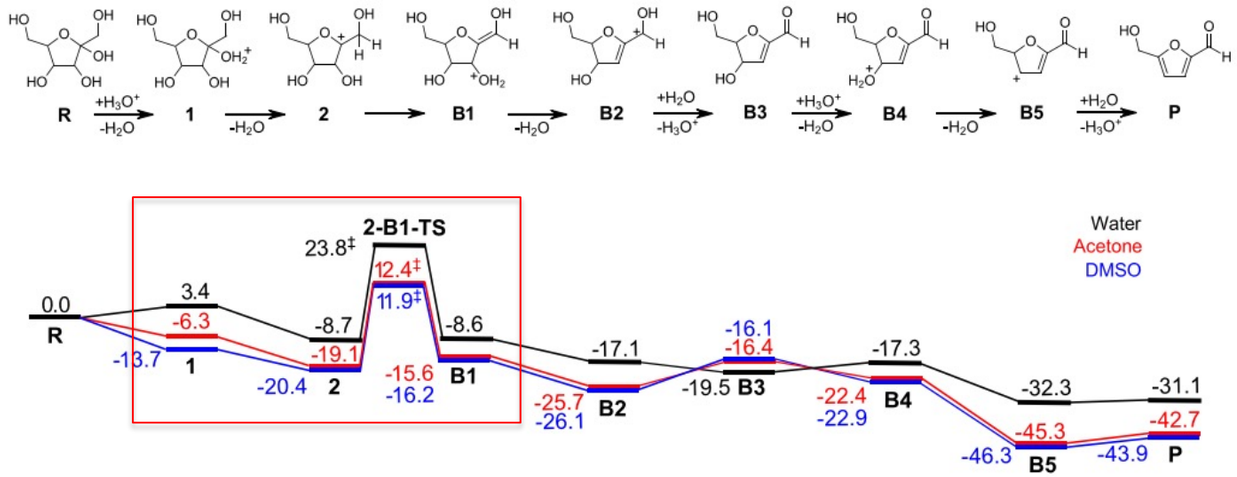
ACS Catal. 2020, 10, 14707−14721

Multivariate Analysis - Motivation

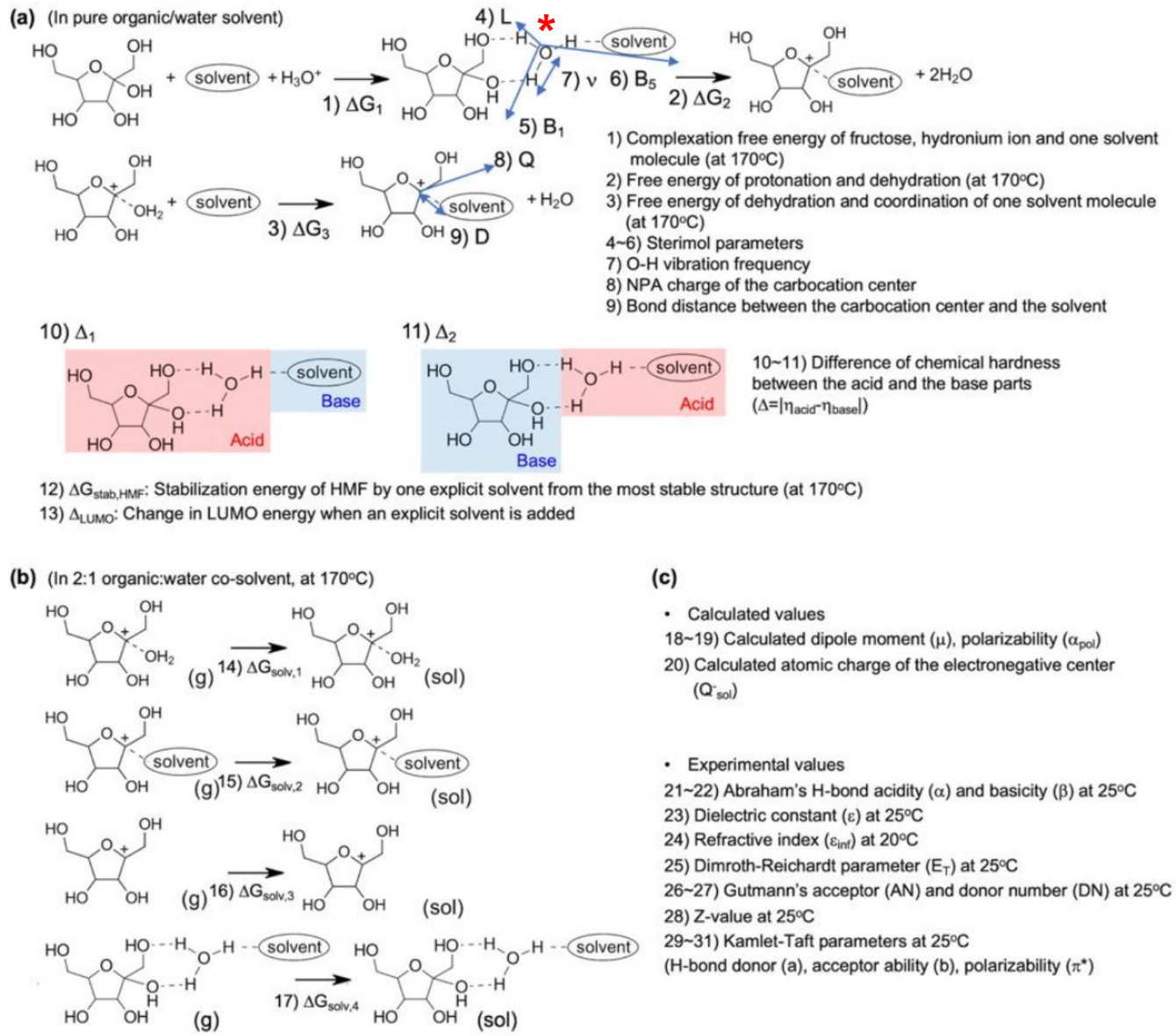


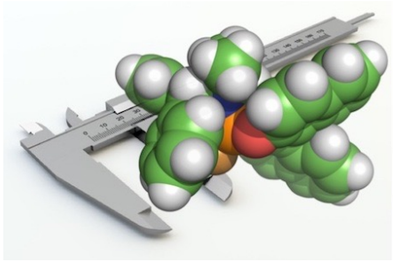
Gibbs free energy at 170°C

M06-2X/6-311++g(d,p), SMD solvation model



Multivariate Analysis





https://github.com/bobbypaton/Sterimol
https://github.com/bobbypaton/wSterimol
https://github.com/bobbypaton/DBSTEP

31 variables relevant to the solvent effect

↓

Correlation analysis of variables

↓

Multivariate linear regression against measured HMF yields in 13 solvents

↓

Candidate models

↓

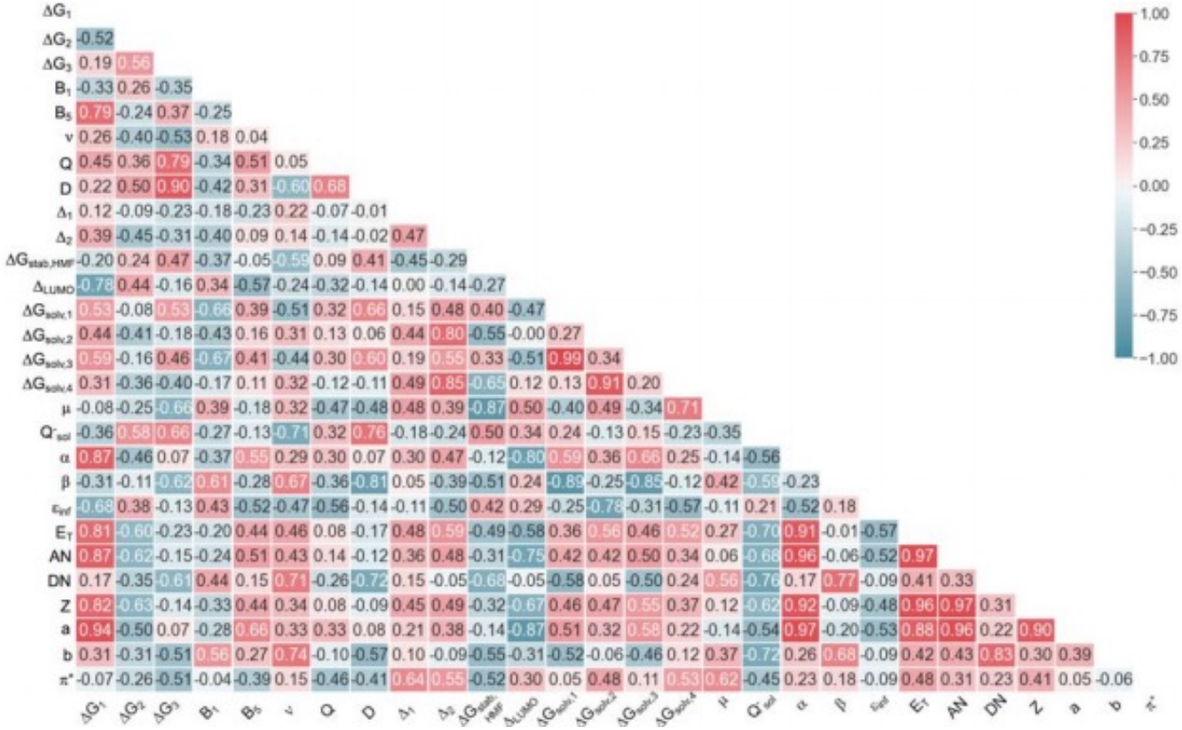
Leave-one-out cross-validation, external validation for each model

↓

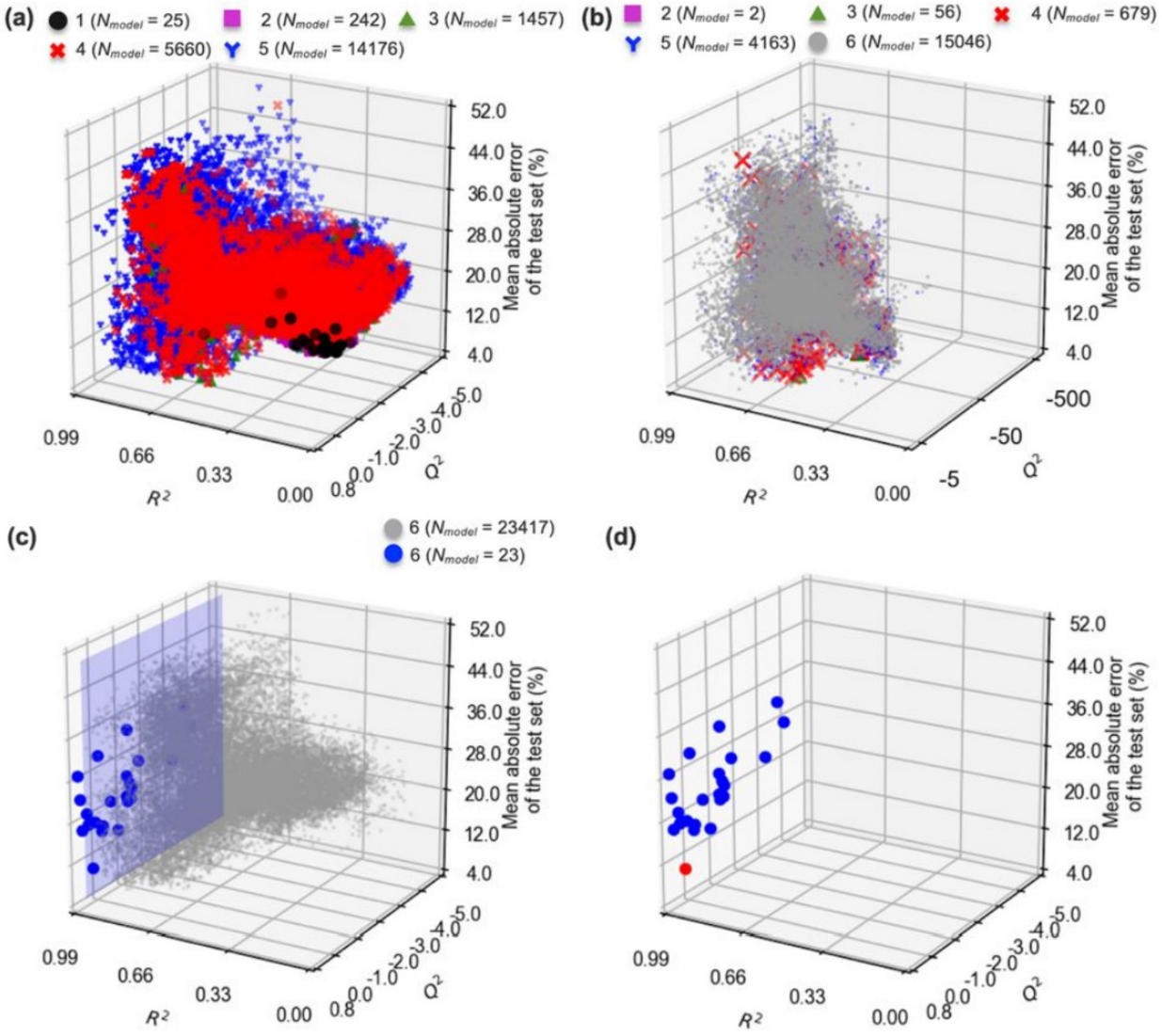
Multivariate model

Training set: 13
Test set: 4 solvents

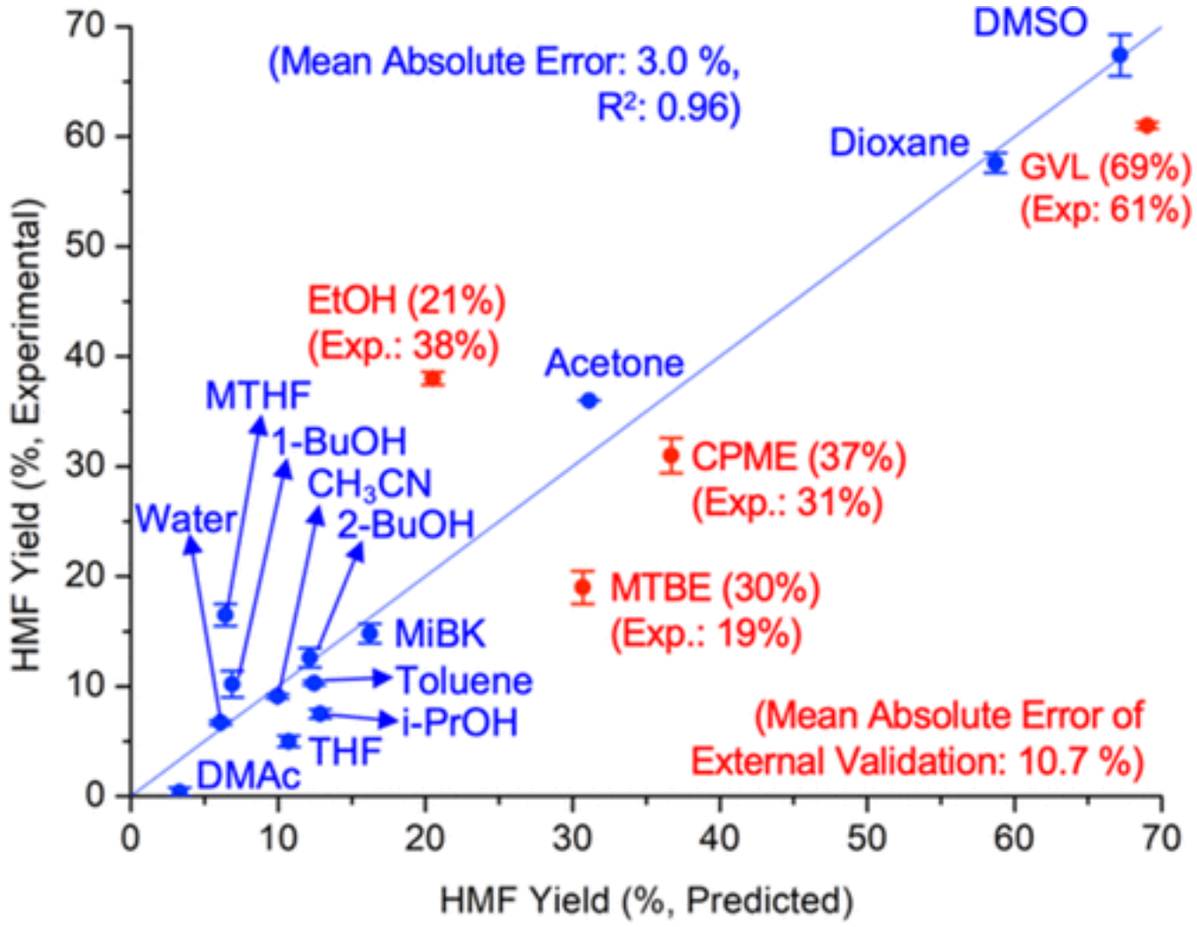
Correlation Analysis



Analysis of Candidate Models

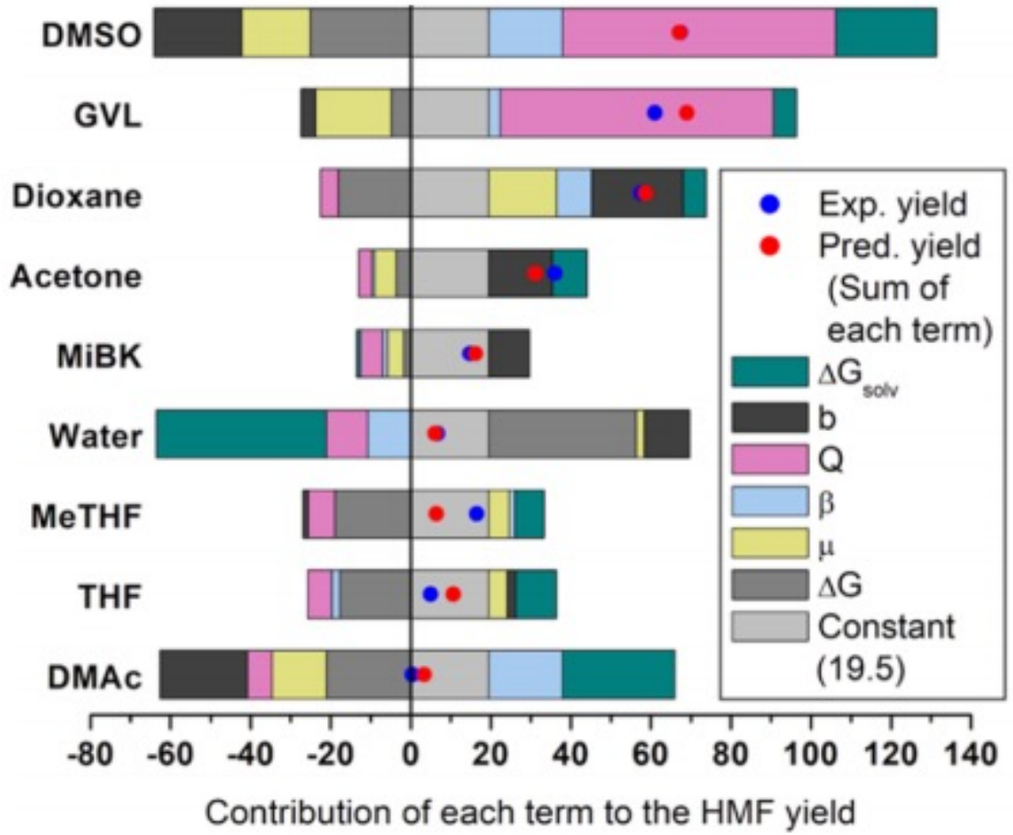


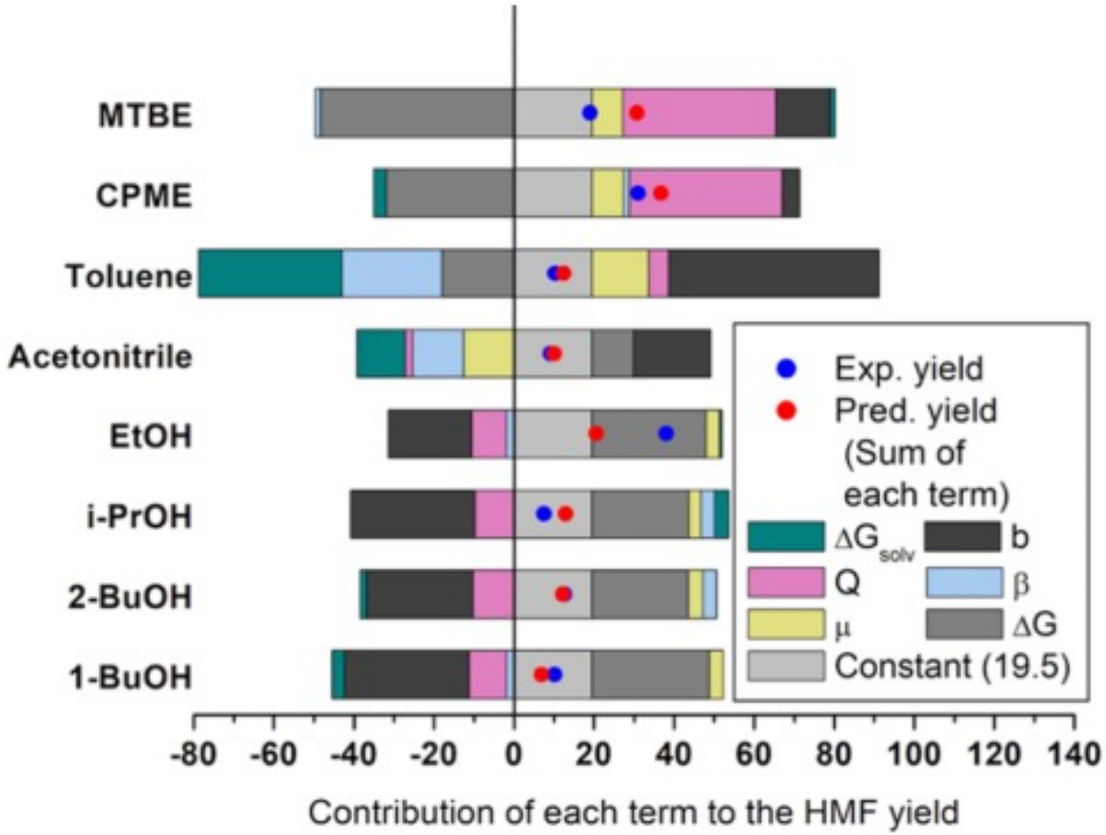
$$\text{HMF yield (\%)} = 21.3\Delta G - 9.8\mu + 11.4\beta - 20.1Q - 24.4b - 19.8\Delta G_{solv} + 19.5$$



Chemical Explanation of the Model

$$\text{HMF yield (\%)} = 21.3\Delta G - 9.8\mu + 11.4\beta - 20.1\mathbf{Q} - 24.4\mathbf{b} - 19.8\Delta G_{solv} + 19.5$$





Assignment

- Apply multivariate analysis to your ongoing research

(OR)

- Develop more efficient workflow than the code I showed you today
  (e.g., minimizing iterations, using more scikit-learn functions, etc.)

(OR)

- Try another linear regression example:
  - Pick G16 output files of one reactant and one transition state
  - Obtain k at different temperatures using GoodVibes
  - Obtain A, Ea through linear regression

$$k_{rxn} = \frac{k_B T}{h} \exp\left(-\frac{\Delta G^{\ddagger}}{RT}\right) = A \exp\left(-\frac{E_a}{RT}\right)$$

(From G16)

ln k = ln A – Ea/RT

  - If it is too easy, try fitting to modified Arrhenius equation with non-linear regression

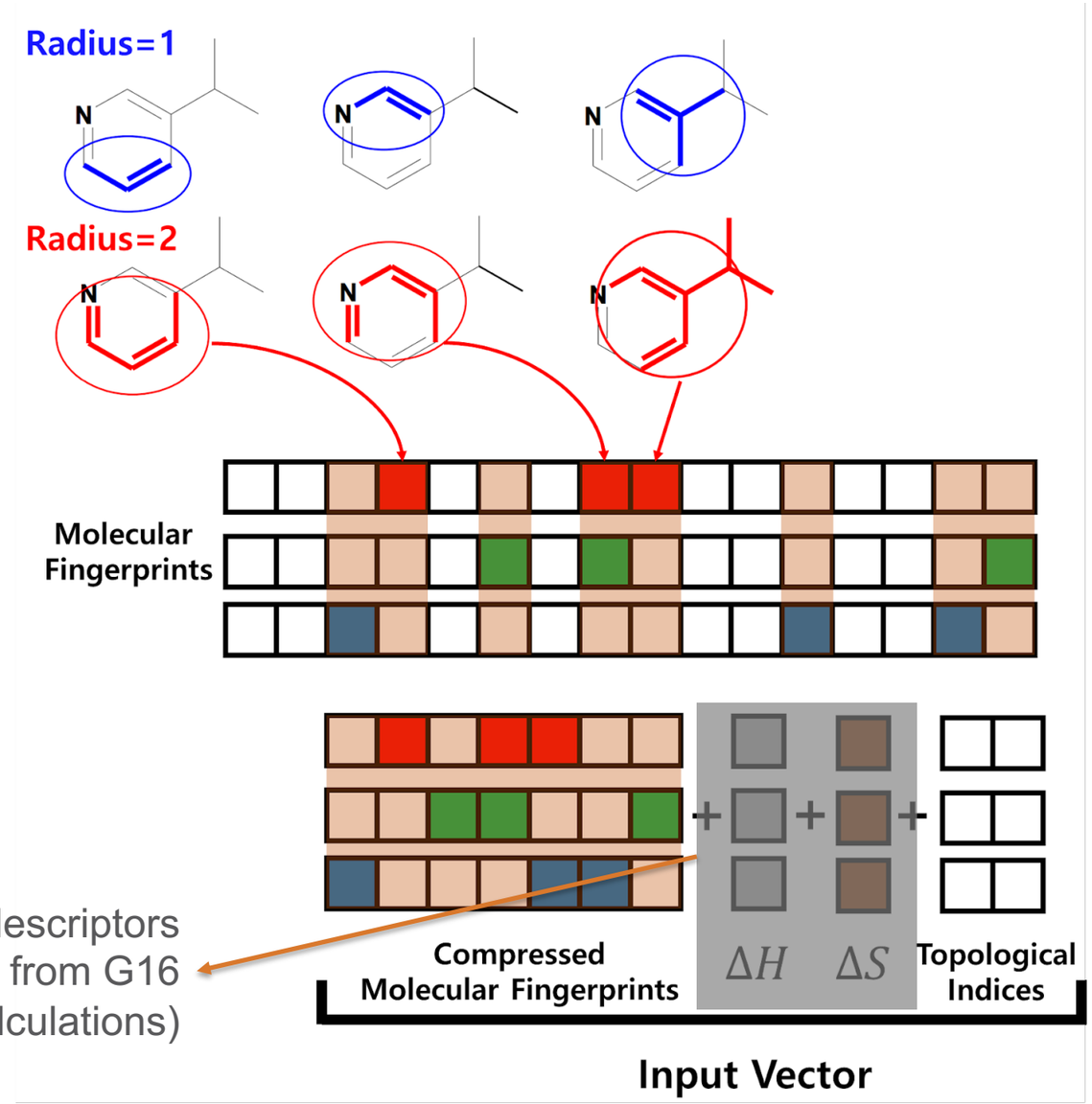
$$k = AT^n e^{-E_a/(RT)}.$$

https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.curve_fit.html

Class 4 – Machine Learning

Colorado State University

Target Property and ML Models

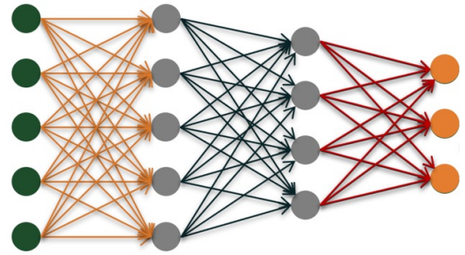


(Skip descriptors from G16 calculations)

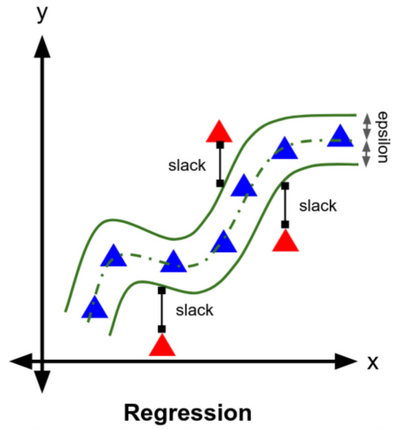
Predicting activation energy of gas phase reactions

Reaction data are obtained from Reaction Mechanism Generator (RMG) Database
2,386 gas-phase reactions related to combustion (80% for training, 20% for test set)

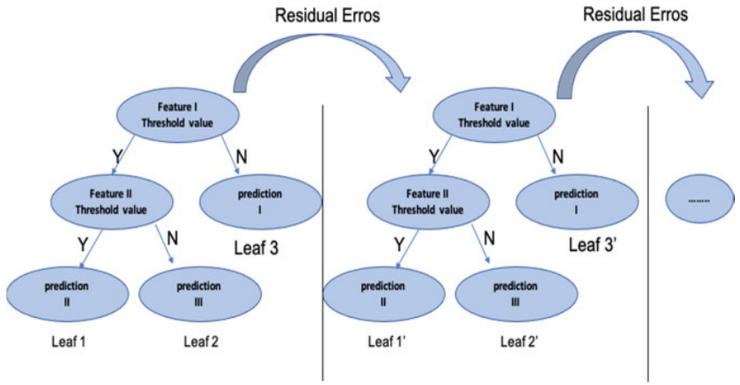
ANN (multilayer perceptron)



SVR



XGBoost



Chem. Eur. J. 2018, 24, 12354-12358.

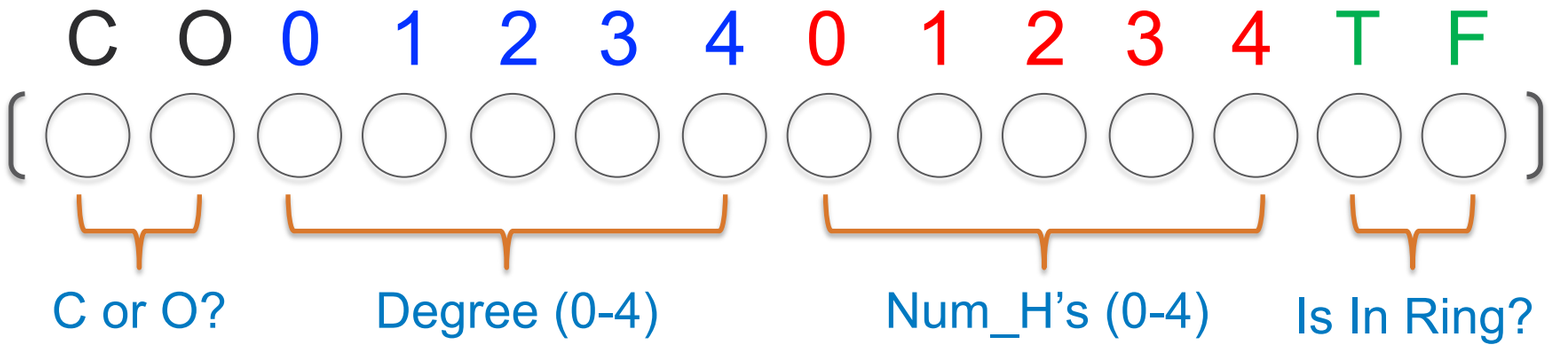
https://medium.com/it-paragon/support-vector-machine-regression-cf65348b6345

https://doi.org/10.1016/j.asej.2020.11.011

Miscellaneous

Colorado State University

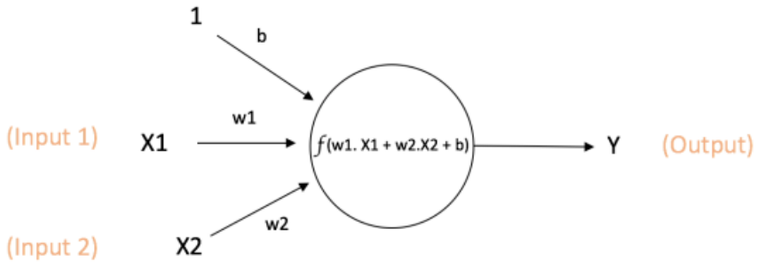
One-hot Vector - Example



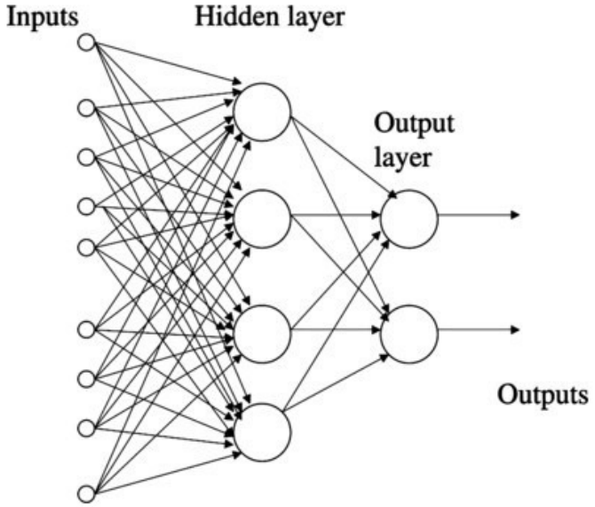
e.g.) Carbon, Degree 4, 3 hydrogens, not in the ring
(**1**,0, 0,0,0,0,**1**, 0,0,0,**1**,0, 0, 1)

A Brief Glimpse of Neural Nets

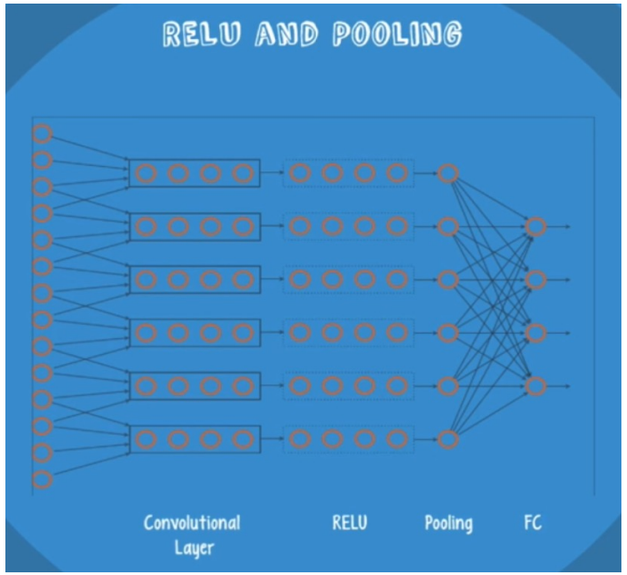
- Artificial neural network

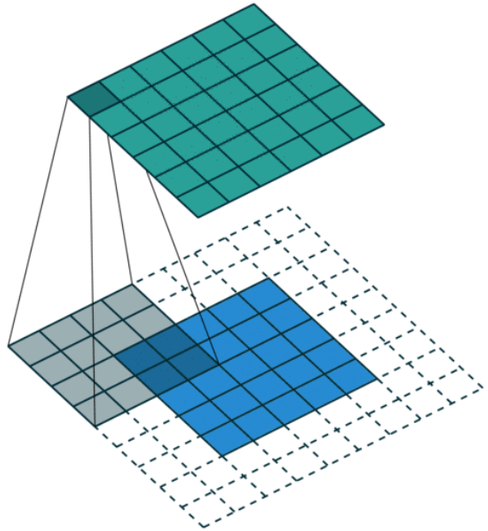


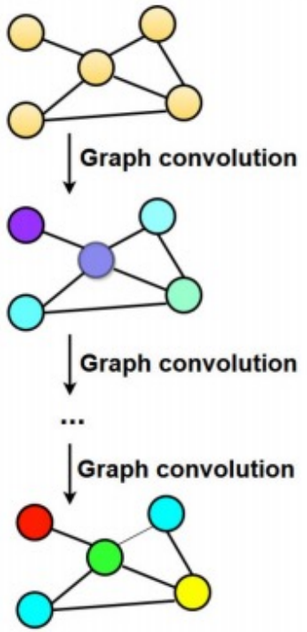
Output of neuron = Y= $f(w1.\ X1 + w2.X2 + b)$

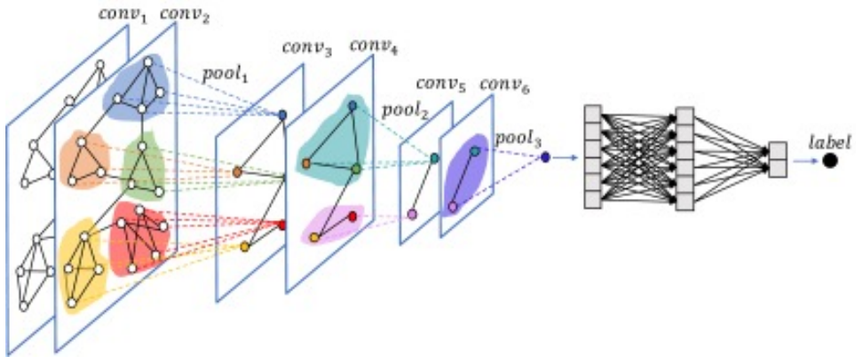


- Convolutional neural network







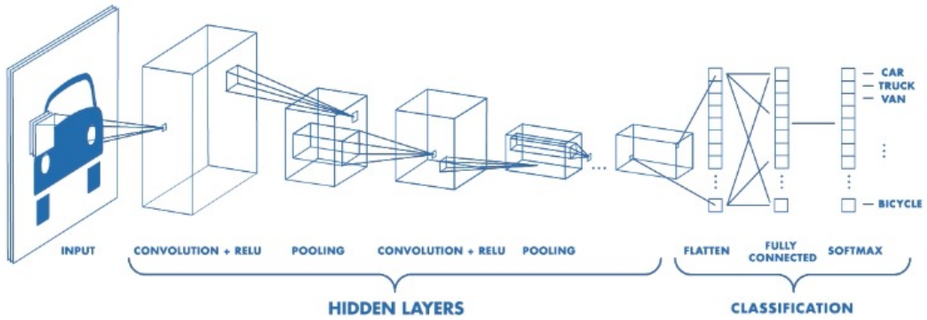


- Node – atoms, edges – bonds
- Each node and edge has its feature vector – iteratively trained to predict the desired molecular properties

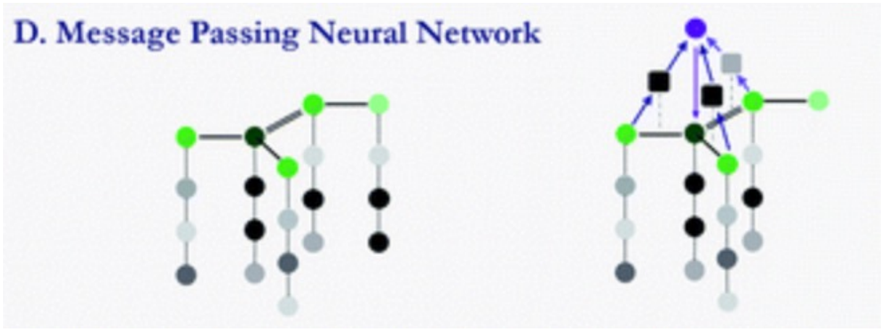
https://medium.com/@jayeshbahire/perceptron-and-backpropagation-970d752f4e44
https://www.groundai.com/project/graph-convolutional-networks-with-eigenpooling/1
arXiv: 1805.10988

CNN and GNN

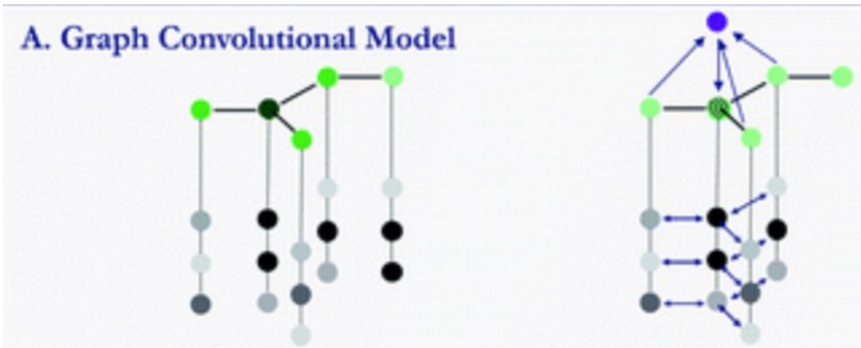
- **Various types of neural networks**
  - Convolutional neural network (CNN)



  - Message-passing neural network (MPNN)



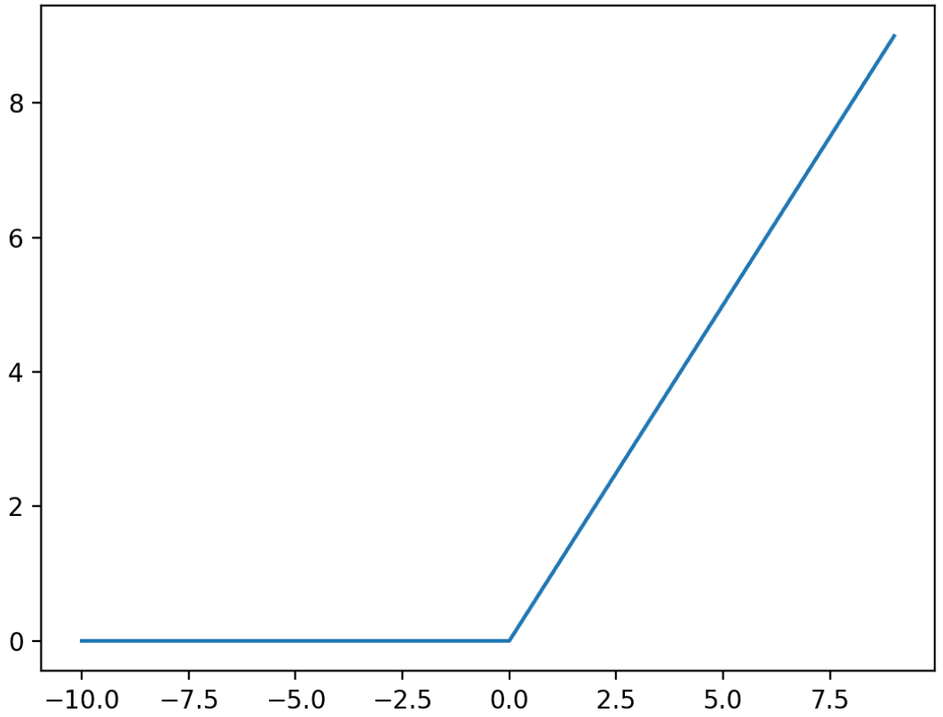
  - Graph convolutional network (GCN)
  - Graph attention network (GAT)

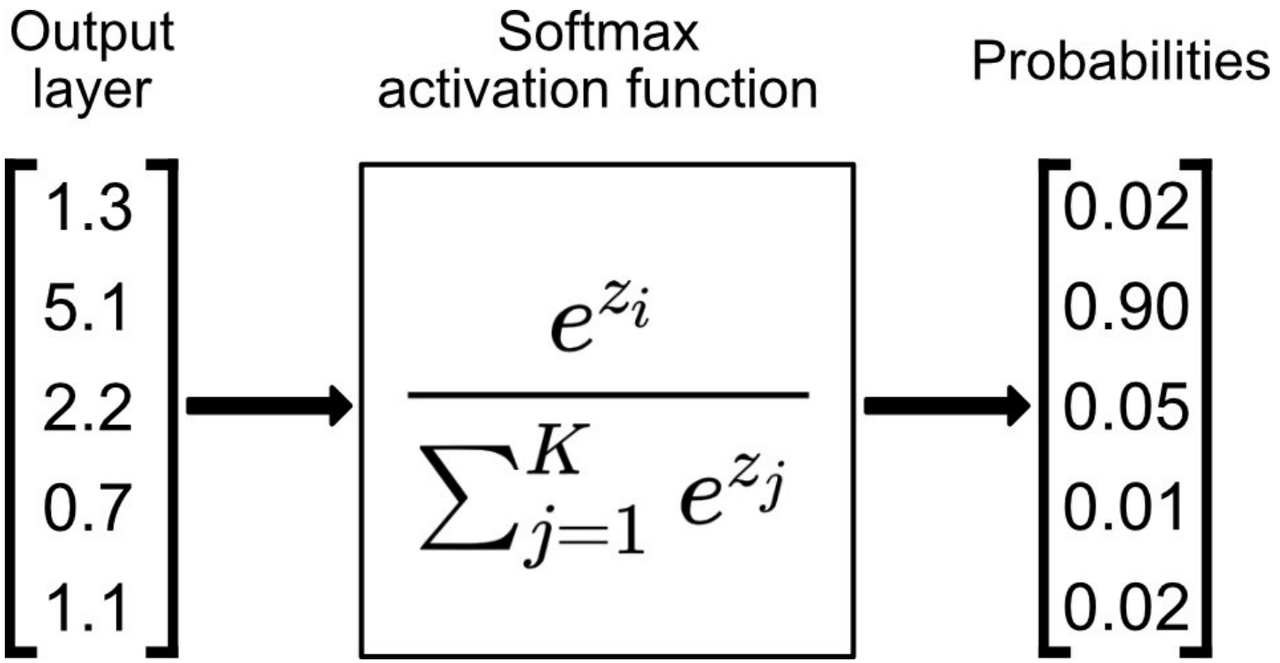


- **And other techniques…**

Chem. Sci., **2018**, *9*, 513-530.

Activation Functions





(1) To introduce non-linearity
(2) To resolve vanishing gradient problem
(3) For a specific purpose
(Regressor->classifier, target values are all positive, etc.)