

# **DELHI TECHNOLOGICAL UNIVERSITY**

## **DEPARTMENT OF APPLIED CHEMISTRY**



### **Estimating Aqueous Solubility Directly from Molecular Structure using Machine Learning**

Made under supervision of:

**Dr. Richa Srivastava**

Submitted by:

**Varsha Rani (2K20/B18/36)**

**Utkarsh Sahu (2K20/B18/27)**

## 1. Introduction

---

Aqueous solubility is one of the key physical properties of interest to a medicinal or agrochemical chemist. Solubility affects the uptake/distribution of biologically active compounds in living material and the environment, thus affecting their potential efficacy and marketability. Accurate equilibrium solubility determination is a time-consuming experiment, and it is useful to be able to assess solubility in the absence of a physical sample. Aqueous solubility is a key factor in drug discovery. *If a compound is not soluble, it will typically be poorly bioavailable, making it difficult to use in in-vivo studies, and ultimately to deliver to patients.* Solubility can even be a problem in early discovery. In some cases, *poorly soluble compounds can create problems for biochemical and cellular assays.* Aqueous solubility is incredibly important for formulation selection and subsequent development processes. Many metabolic activities do not put adequate efforts if not ensuring compounds are soluble and achieve reasonable exposure. Low aqueous solubility leads to many potential complications. For instance, in biochemical-based assays and cell-based assays, low solubility may lead compounds precipitate from screening buffer which may create a high risk of erroneous results, costly setbacks, and false leads.

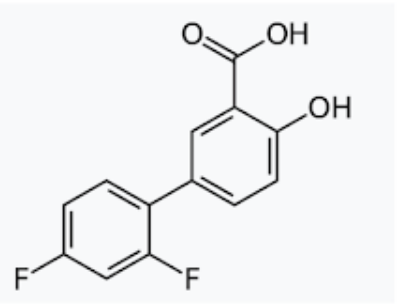
As originally defined by **Yalkowsky** <sup>[1]</sup>, aqueous solubility is a function of both lipophilicity and crystal packing forces. **Yalkowsky** <sup>[1]</sup> defined the **General Solubility Equation (GSE)**.

$$\text{Log (S)} = 0.5 - 0.01(\text{MP} - 25) - \text{Log (K}_{\text{ow}})$$

Where **Log(S)** is the aqueous solubility, **MP** is the melting point in degrees Celsius and **Log (K<sub>ow</sub>)** is the log of the octanol-water partition coefficient of the un-ionized species.

So far this sounds simple. All we need to do to predict solubility is predict **Log (K<sub>ow</sub>)** and melting point. Octanol partition can be calculated with reasonable accuracy from a compound's structure but estimating melting point is far harder. Where a measured melting point is

available, **GSE** becomes the method of choice, while other methods, based solely on structure, must be used in situations where **MP** is not available. In order to effectively predict melting point, we would probably have to know how a molecule stacks in a crystal structure. While the field of crystal structure prediction has made some progress, the technique is far from routine. Some of the best structure prediction methods can require months to generate a prediction for a single molecule. The problem is further compounded by the fact that molecules can be crystallized in different crystal forms, better known as polymorphs. These polymorphs can have dramatically different melting points, and as a result, vastly different solubilities. For instance, aqueous solubility for **4 different forms for Diflunisal**. Note that there is a **100-fold difference** in solubility between the most and least soluble forms.


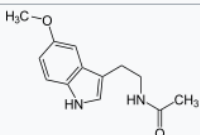
 Diflunisal	Form	Solubility $\mu\text{g/ml}$	LogS mol/L
	1	26	-3.9
	2	7.6	-4.5
	3	0.93	-5.4
	4	0.29	-5.9

We have probably seen the application of machine learning in one form or another. For instance, machine learning has been used together with computer vision in self-driving cars and self-checkout convenience stores, in entertainment for recommendation systems and the list goes on. In this will explore how machine learning is being used for drug discovery particularly using a simple regression model in **Python** for predicting the solubility of molecules (Log(**S**) values). Herein, we will be reproducing a research article entitled [\*“ESOL: Estimating Aqueous Solubility Directly from Molecular Structure”\*](#) by **John S. Delaney**<sup>[2]</sup>.

## 2. Working

### 2.1 Materials & Resources used

- The **RDkit** library, an open-source toolkit for **cheminformatics** that allows us to handle chemical structures and the calculation of their molecular properties (*i.e.*, for quantitating the molecular features of each molecule that we can subsequently use in the development of a machine learning model).
- **SMILES** (**S**implified **M**olecular **I**nput **L**ine **E**ntry **S**ystem) is a specification in the form of a **line notation** for describing the structure of chemical species using short **ASCII strings**. SMILES strings can be imported by most molecule editors for conversion back into two-dimensional drawings or three-dimensional models of the molecules. Few examples of SMILES notation is given here.

Molecule	Structure	SMILES formula
Dinitrogen	$\text{N}\equiv\text{N}$	<chem>N#N</chem>
Methyl isocyanate (MIC)	$\text{CH}_3\text{-N=C=O}$	<chem>CN=C=O</chem>
Copper(II) sulfate	$\text{Cu}^{2+}\text{SO}_4^{2-}$	<chem>[Cu+2].[O-]S(=O)(=O)[O-]</chem>
Vanillin		<chem>O=Cc1ccc(OC)c(O)c1</chem> <chem>COc1cc(C=O)ccc1O</chem>
Melatonin ( $\text{C}_{13}\text{H}_{16}\text{N}_2\text{O}_2$ )		<chem>CC(=O)NCCC1=CNc2cc(OC)cc2</chem> <chem>CC(=O)NCCc1c[nH]c2ccc(OC)cc12</chem>

- We have used the **Delaney<sup>[2]</sup> solubility dataset** that is available as a [Supplementary file](#) of the paper [ESOL: Estimating Aqueous Solubility Directly from Molecular Structure](#).

### 2.2 Adding Chemical Descriptors to Dataset (🔗 : Colab Notebook link)

To predict **Log(S)** (log of the aqueous solubility), the study by Delaney<sup>[2]</sup> makes use of 4 molecular descriptors:

1. **Log ( $K_{ow}$ )** (Octanol-water partition coefficient)
2. **MW** (Molecular weight)
3. **RB** (Number of rotatable bonds)
4. **AP** (Aromatic proportion = number of aromatic atoms/numbers of heavy atoms)

Unfortunately, RDkit readily computes the first 3. As for the AP descriptor, we have calculated this by manually computing the ratio of the *number of aromatic atoms* to the *total number of heavy atoms* (which RDkit can compute).

### Calculating Aromatic Proportion Descriptor for Dataset

```
#A custom function to calculate aromatic proportion
def AromaticAtoms(m):
    aromatic_atoms = [m.GetAtomWithIdx(i).GetIsAromatic() for i in range(m.GetNumAtoms())]
    aa_count = []
    for i in aromatic_atoms:
        if i==True:
            aa_count.append(1)
    sum_aa_count = sum(aa_count)
    return sum_aa_count

#calculating AP of every compound in Dataset
desc_AromaticProportion = [AromaticAtoms(Chem.MolFromSmiles(element))/Descriptors.HeavyAtom
Count(Chem.MolFromSmiles(element)) for element in sol.SMILES]

#Converting the list obtained to a panda dataframe
df_desc_AromaticProportion = pd.DataFrame(desc_AromaticProportion,columns=['Aromatic propo
rtion'])

#concating the data frames to get X matrix. df data frame has been already been calculated.
refer to the COLAB NOTEBOOK
X = pd.concat([df,df_desc_AromaticProportion], axis=1)

#Generating Y matrix
Y = sol.iloc[:,1]
```

This is how are X and Y matrices would look like after all the computations.

	MolLogP	MolWt	NumRotatableBonds	Aromatic proportion
0	2.59540	167.850	0.0	0.000000
1	2.37650	133.405	0.0	0.000000
2	2.59380	167.850	1.0	0.000000
3	2.02890	133.405	1.0	0.000000
4	2.91890	187.375	1.0	0.000000
...	...	...	...	...
1139	1.98820	287.343	8.0	0.000000
1140	3.42130	286.114	2.0	0.333333
1141	3.60960	308.333	4.0	0.695652
1142	2.56214	354.815	3.0	0.521739
1143	2.02164	179.219	1.0	0.461538

1144 rows x 4 columns

0	-2.180
1	-2.000
2	-1.740
3	-1.480
4	-3.040
...	...
1139	1.144
1140	-4.925
1141	-3.893
1142	-3.790
1143	-2.581

## 2.3 Model Building and comparing equations( : Colab Notebook link)

```
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2)

from sklearn import linear_model
from sklearn.metrics import mean_squared_error, r2_score
model = linear_model.LinearRegression()
model.fit(X_train, Y_train)

#Predicting Log(S) value of from X_test data
Y_pred_test = model.predict(X_test)
print('Coefficients:', model.coef_)
print('Intercept:', model.intercept_)
print('Mean squared error (MSE): %.2f' % mean_squared_error(Y_test, Y_pred_test))
print('Coefficient of determination (R^2): %.2f' % r2_score(Y_test, Y_pred_test))
```

The output of the following code snippet is:

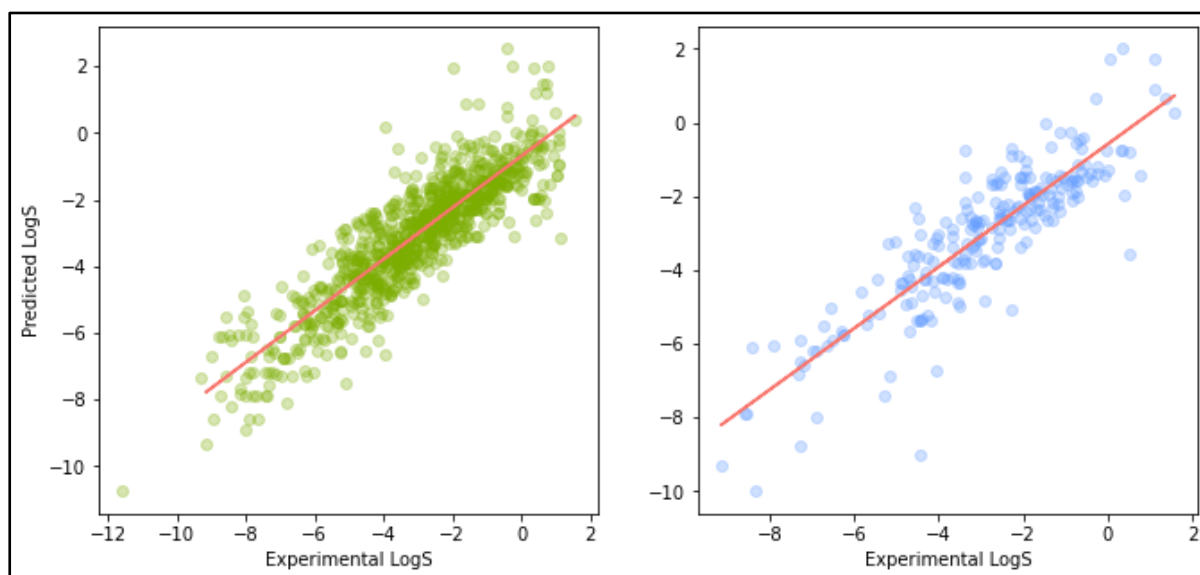
```
Coefficients: [-0.72827793 -0.00683165 -0.00112886 -0.37325709]
Intercept: 0.3114022544878621
Mean squared error (MSE): 1.02
Coefficient of determination (R^2): 0.77
```

```
#generating Eqn for Log(S)
yintercept = '%.2f' % model.intercept_
LogKow = '%.2fLog(Kow)' % model.coef_[0]
MW = '%.4fMW' % model.coef_[1]
RB = '%.4fRB' % model.coef_[2]
AP = '%.2fAP' % model.coef_[3]
print('LogS = ' + ' ' + yintercept + ' ' + LogKow + ' ' + MW + ' ' + RB + ' ' + AP)
```

Thus, the Equation obtained by **Linear Regression** is:

```
LogS = 0.31 -0.75Log(Kow) -0.0069MW 0.0005RB -0.38AP
```

### 3. Figures or Graphs



Scatter plot of *Experimental vs. Predicted Log(S)* for training and testing dataset respectively

User Input Features

SMILES input

C1CCc2ccccc2C1

Clc1cccc(Cl)c1Cl

O=N(=O)c1cccc1N(=O)=O

CC(=C)C(=C)C

Input SMILES

```
0 : "C1CCc2ccccc2C1"
1 : "Clc1cccc(Cl)c1Cl"
2 : "O=N(=O)c1cccc1N(=O)=O"
3 : "CC(=C)C(=C)C"
```

Computed molecular descriptors

	MolLogP	MolWt	NumRotatableBonds	AromaticProportion
1	2.5654	132.2060	0	0.6000
2	3.6468	181.4490	0	0.6667
3	1.5030	168.1000	2	0.5000
4	2.1386	82.1460	1	0

Predicted LogS values

```
0
0 : -2.7727
1 : -3.9280
2 : -2.1729
3 : -1.8687
```

*Utkarsh Sahu(2K20/B18/27) has made the User Interface for our project using STREAMLIT library.*

## 4. Conclusion

---

The aim of this work was to produce a robust alternative to solubility estimation by **GSE** where the melting point of the compound was unknown. **ESOL** seems to be a **viable alternative** to **GSE** for predicting the solubility of pesticide/drug-like molecules. The fact that it works so well is something of a surprise and begs the question why. GSE divides the solubility prediction problem into a liquid-liquid partition term and a solid-liquid state change term. **The three non-Log( $K_{ow}$ ) terms in the ESOL equation could be acting in either or both of these terms.** In summary, ESOL provides a fast and robust method for estimating the solubility of drugs and agrochemicals without recourse to physical measurements.

## 5. References

---

1. Jain, N.; Yalkowsky, S. H. [Estimation of the Aqueous Solubility 1: Application to Organic Nonelectrolytes](#). J. Pharm. Sci. 2001, 90(2), 234-252.
2. Delaney, John. (2004). [ESOL: Estimating Aqueous Solubility Directly from Molecular Structure](#). Journal of chemical information and computer sciences. 44. 1000-5. 10.1021/ci034243x.
3. [Predicting Aqueous Solubility - It's Harder Than It Looks](#) by Pat Walters.
4. Stephen J. Franklin, Usir S. Younis, Paul B. Myrdal. [Estimating the Aqueous Solubility of Pharmaceutical Hydrates](#). Journal of Pharmaceutical Sciences.
5. [How to Use Machine Learning for Drug Discovery](#) by Chanin Nantasenamat.
6. Utkarsh's [GitHub Repository: AC102](#) containing colab notebook in which we have trained the model, pickled model and app.py file launching the User Interface.