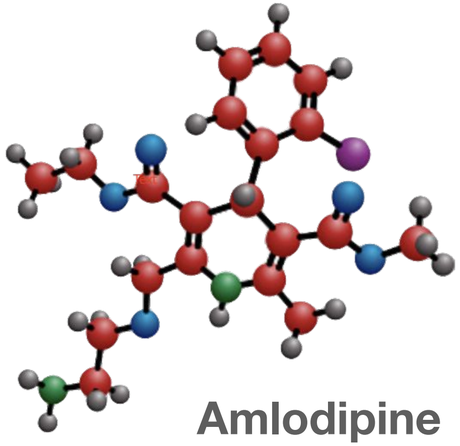
BioSolveAI

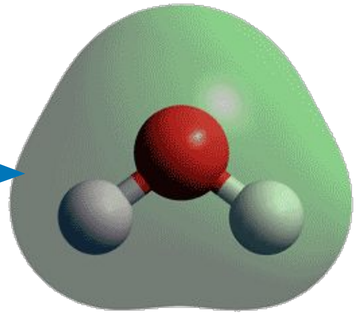
Predicting Molecule Solubility

Raiyann Jacob, Shahd Abu Gharbieh, Tanish Sharma

Problem

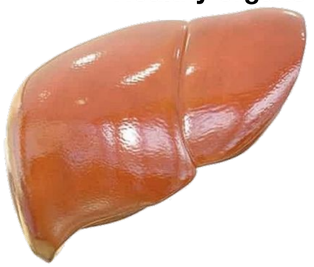


Amlodipine

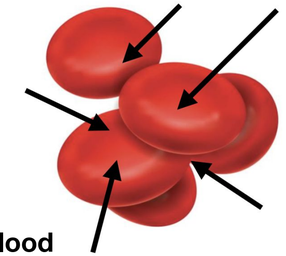


Water Soluble

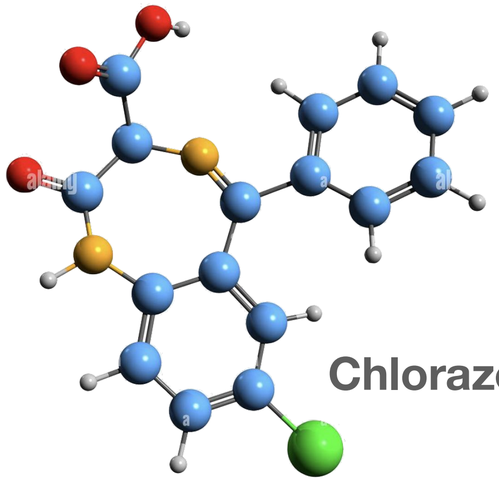
Healthy organs



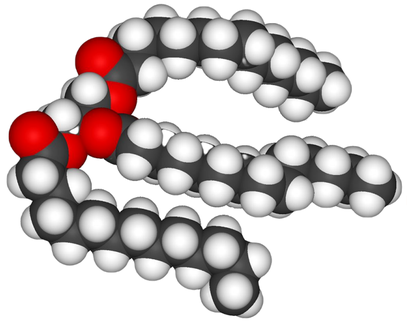
High energy



Fast diffusion into blood



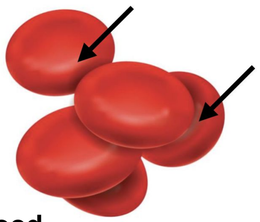
Chlorazepate



Fat Soluble

Fatty organs

Low energy



Slow diffusion into blood

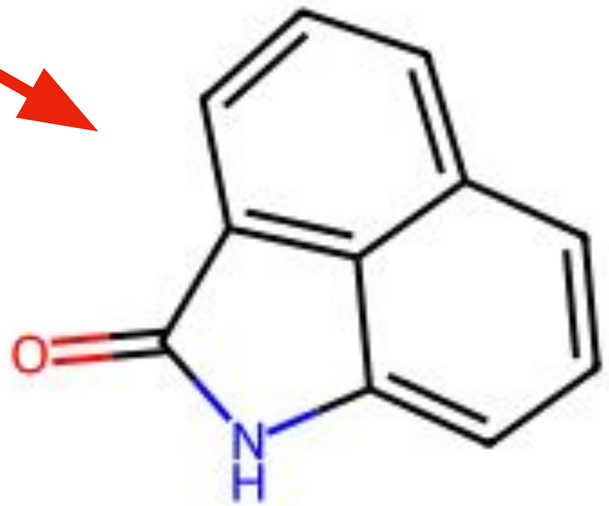
Data and Preprocessing

	1
ID	A-4
Name	Benzo[cd]indol-2(1H)-one
InChI	InChI=1S/C11H7NO/c13-11-8-5-1-3-7-4-2-6-9(12-1...
InChIKey	GPYLCFQEKPUWLD-UHFFFAOYSA-N
SMILES	O=C1Nc2cccc3cccc1c23
Solubility	-3.254767
SD	0.0
Ocurrences	1
Group	G1
MolWt	169.183
MolLogP	2.4055
MolMR	51.9012
HeavyAtomCount	13.0
NumHAcceptors	1.0
NumHDonors	1.0
NumHeteroatoms	2.0

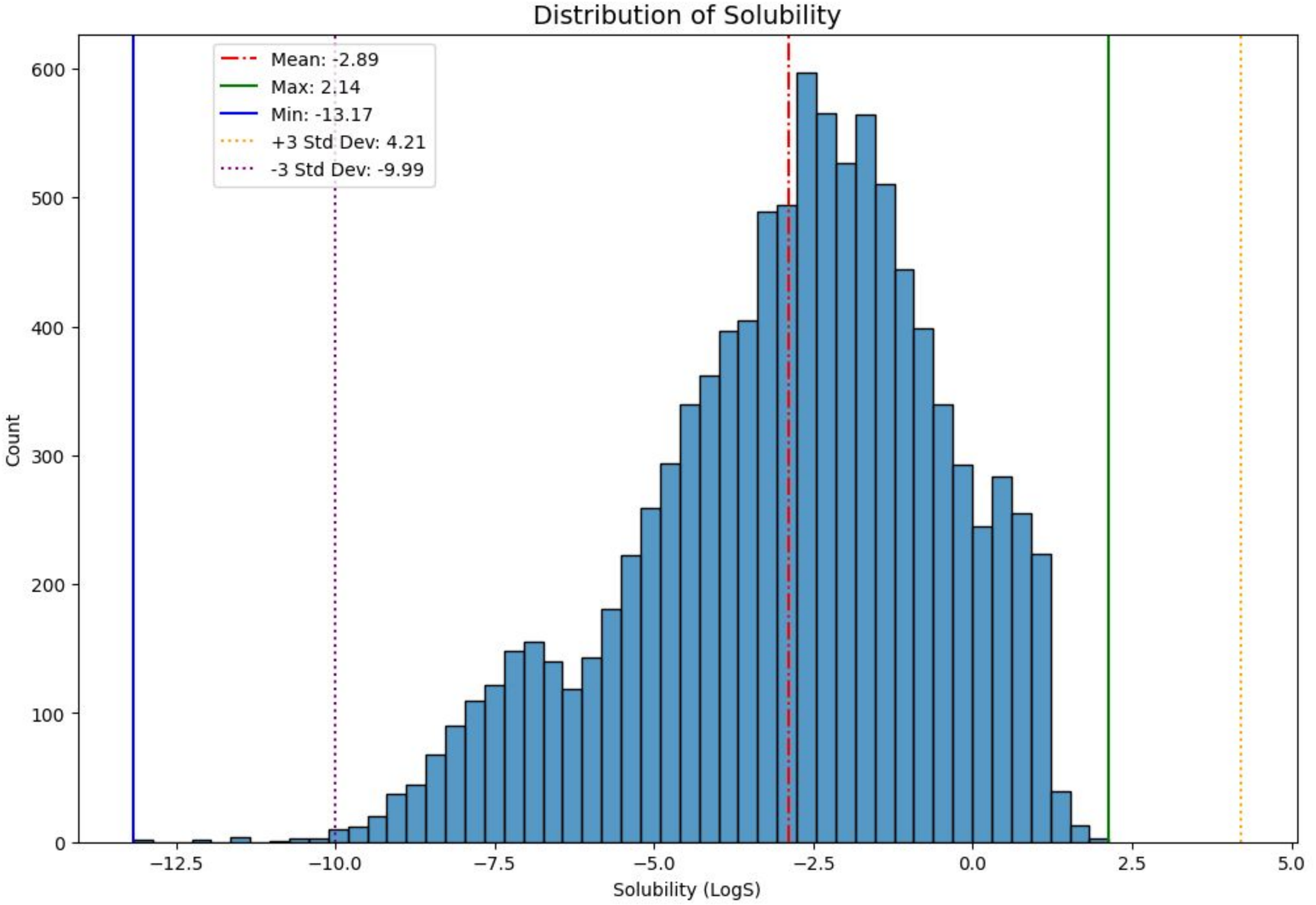
AQSolDB: 9,982 molecular compounds, 26 attributes

SMILES: Compact String Representation

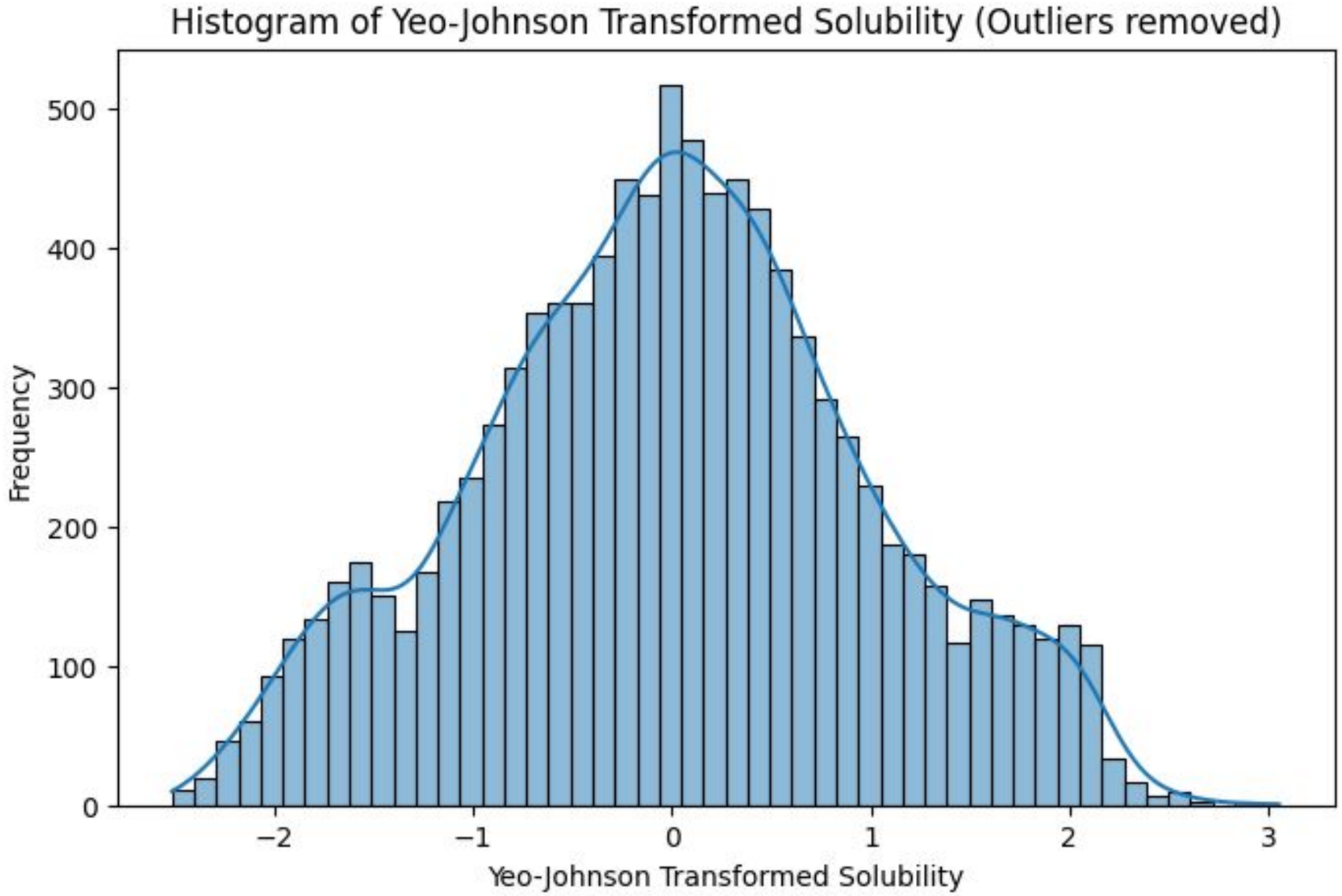
SMILES to RDKit "**Mol**" Object



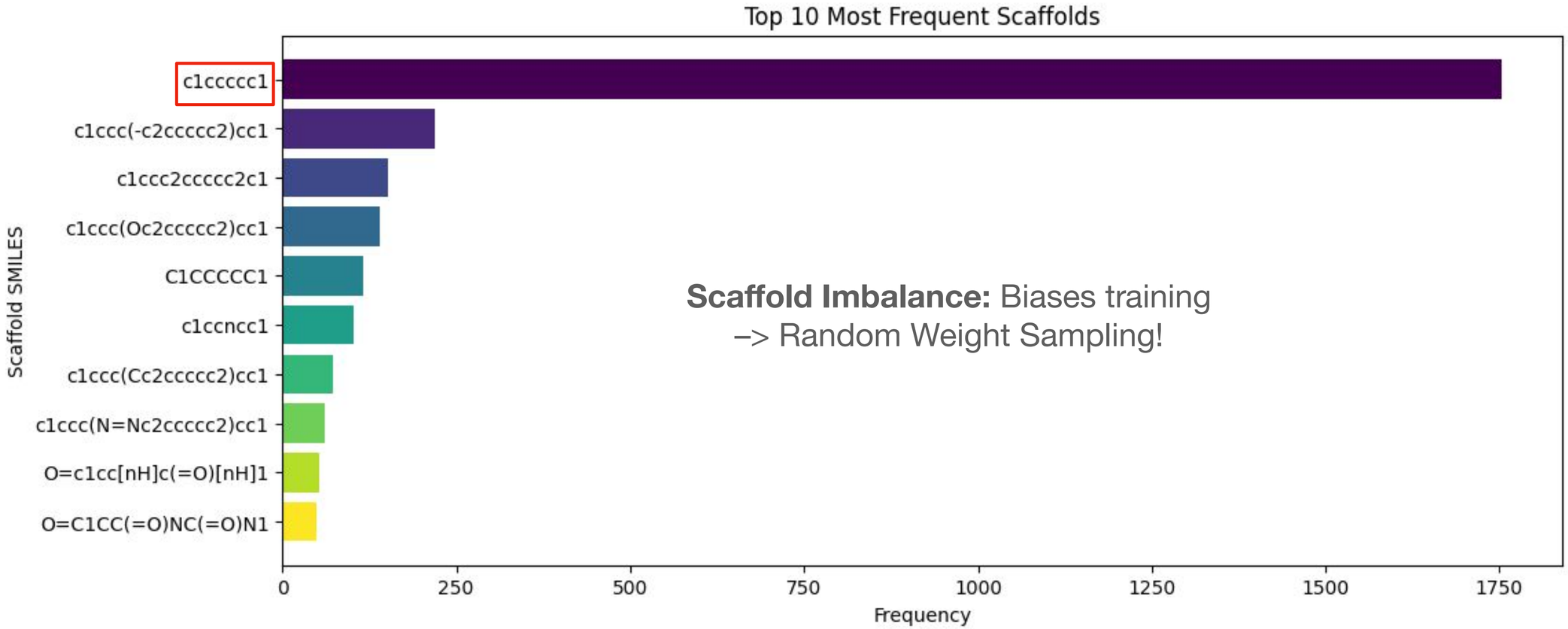
Data and Preprocessing



Data and Preprocessing



Data and Preprocessing



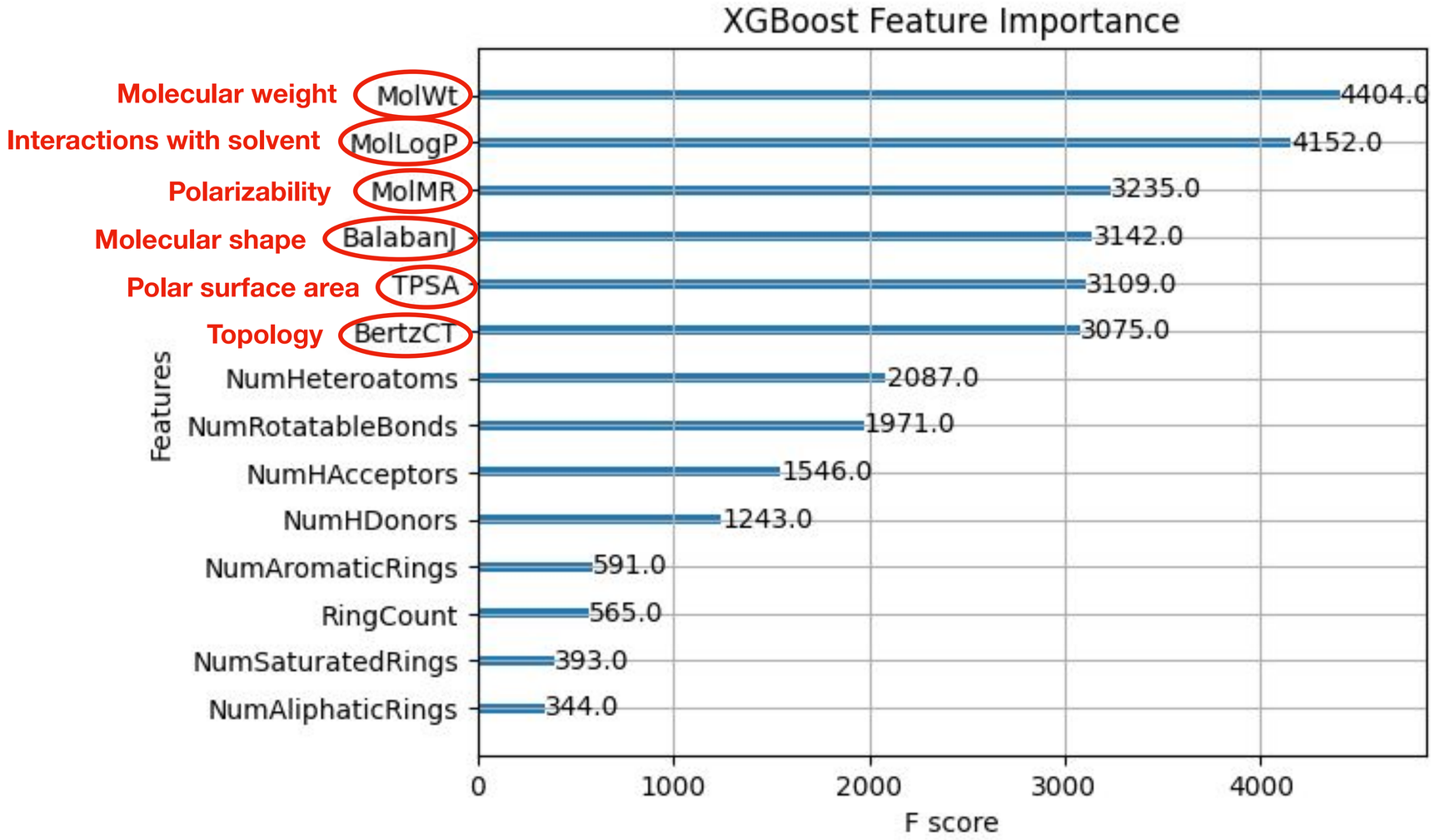
Scaffold Imbalance: Biases training
–> Random Weight Sampling!

Murcko Scaffolds
c1ccccc1

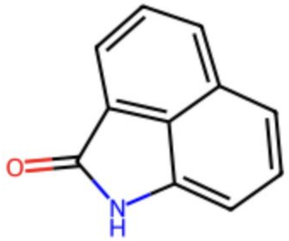
Scaffold Splits: 8/1/1 split with no scaffold overlap
–> Mimics real life!

Baseline Model

XGBoost Regression Model

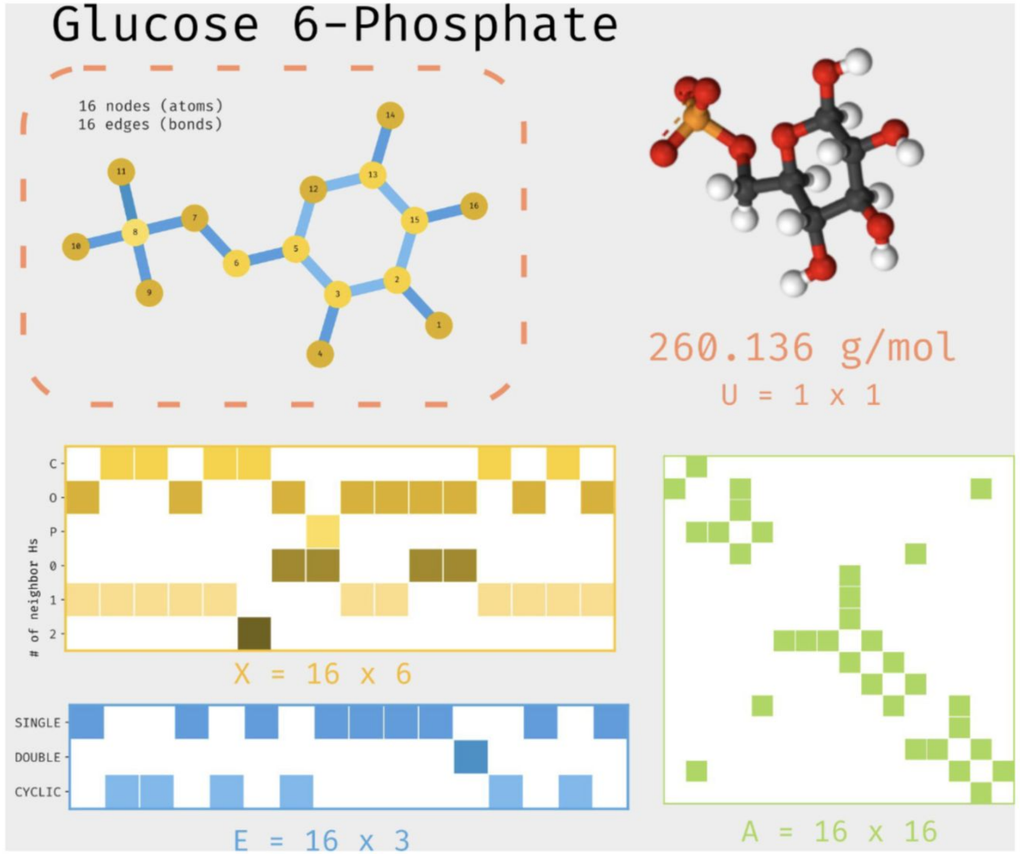


GNN Architecture



Initial Molecule
Tensor Representation

"Graph Tensors"

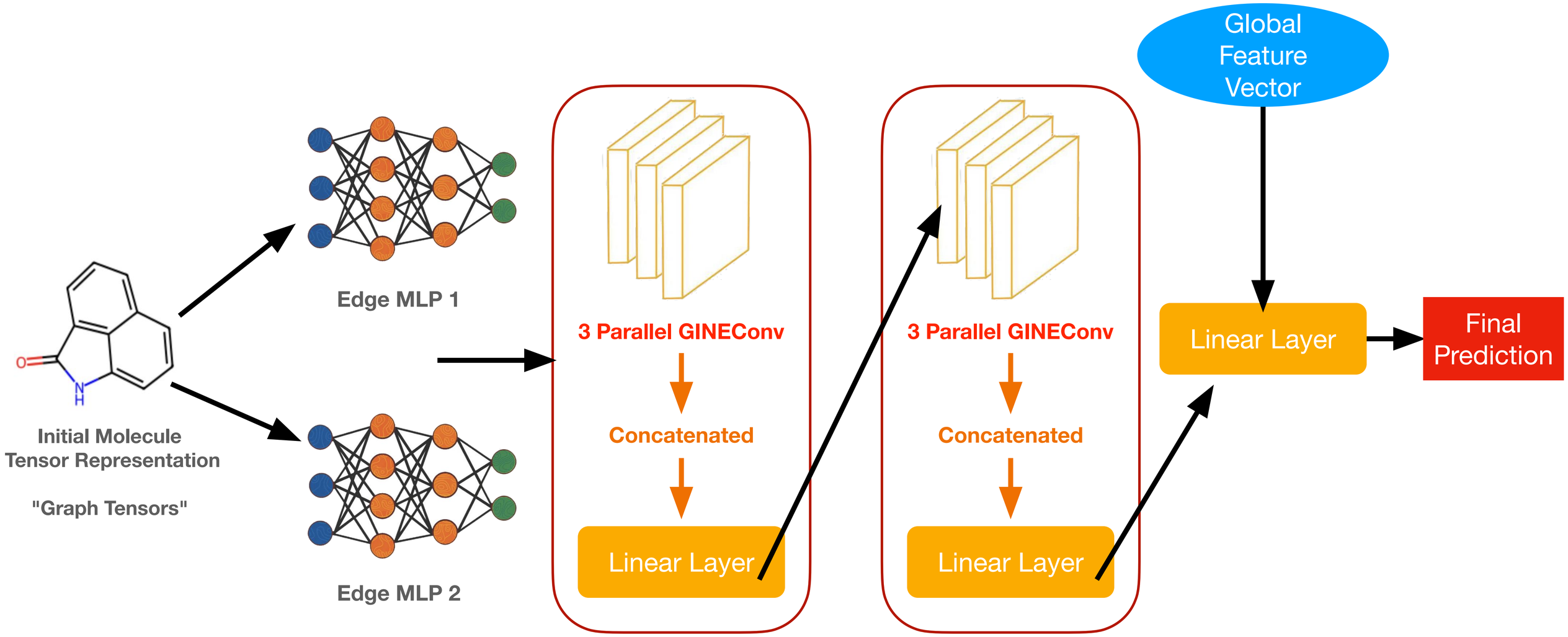



```python
# Node features (X)
  atom_features = []
  for atom in mol.GetAtoms():
    atom_features.append([
        atom.GetAtomicNum(),
        atom.GetIsAromatic(),
        atom.GetDegree(),
        atom.GetFormalCharge(),
        atom.GetTotalNumHs(),
        atom.IsInRing()
      ])
```

```python
# Edge features (E)
  bond_type_to_idx = {
    Chem.rdchem.BondType.SINGLE: 0,
    Chem.rdchem.BondType.DOUBLE: 1,
    Chem.rdchem.BondType.TRIPLE: 2,
    Chem.rdchem.BondType.AROMATIC: 3,
  }
```

```python
# Global features (U), these were the 6 most
  # important features obtained from XGBoost.
  mol_wt = Descriptors.MolWt(mol)
  mol_logP = Crippen.MolLogP(mol)
  tpsa = rdMolDescriptors.CalcTPSA(mol)
  balabanJ = float(BalabanJ(mol))
  mol_mr = Crippen.MolMR(mol)
  bertzCT = BertzCT(mol)
  global_features = [mol_wt, mol_logP,
                     tpsa, balabanJ,
                     mol_mr,bertzCT]
```

GNN Architecture



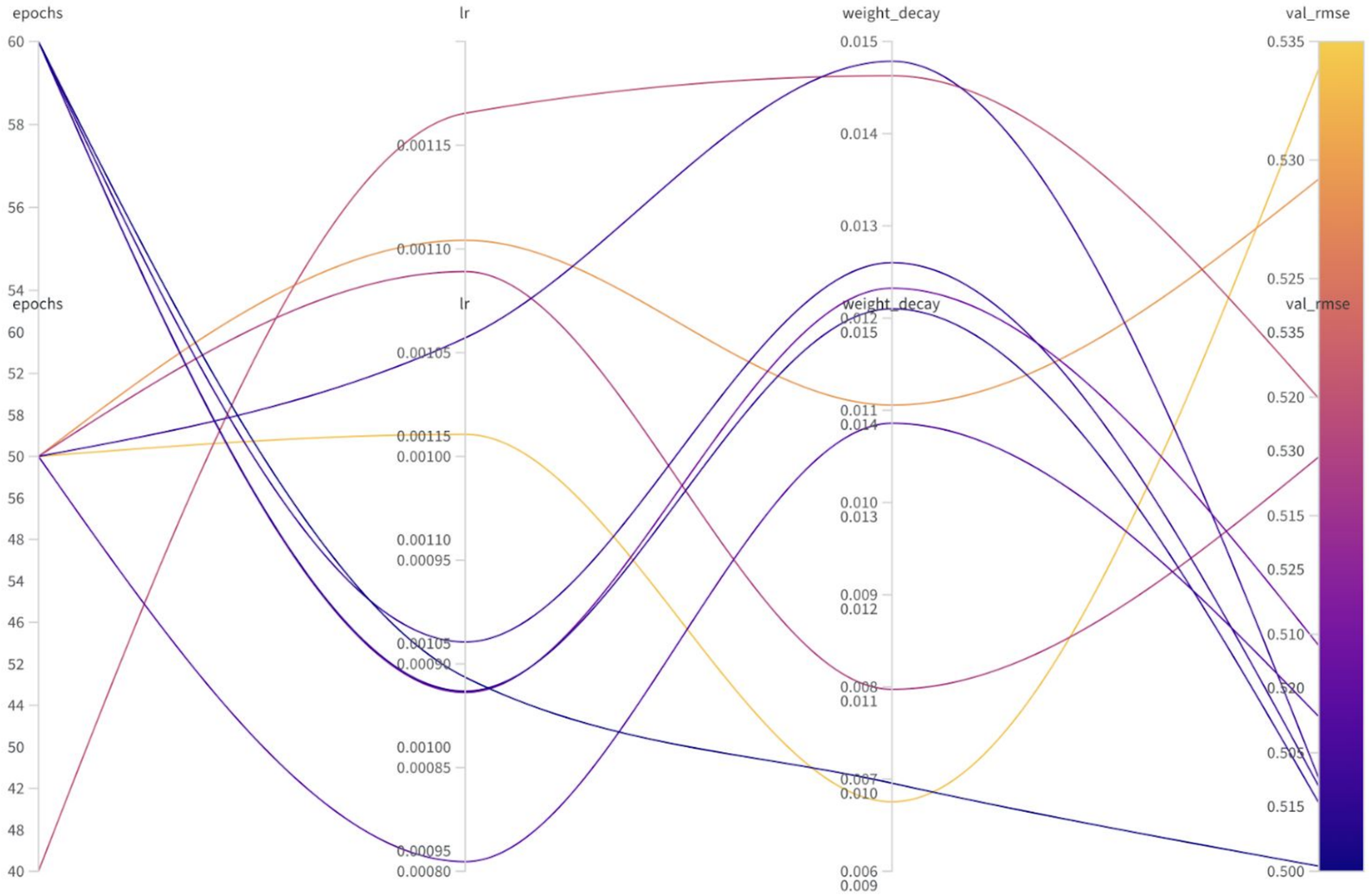
Hyperparameter Tuning

- LayerNorm
- Dropout
- Learning Rate Decay
- AdamW (Adam L2 Regularized)
- Xavier Weight Initialization

Final hyperparams

```
BATCH_SIZE = 32
EPOCHS = 60
LEARNING_RATE = 0.00089355
WEIGHT_DECAY = 0.006956
```

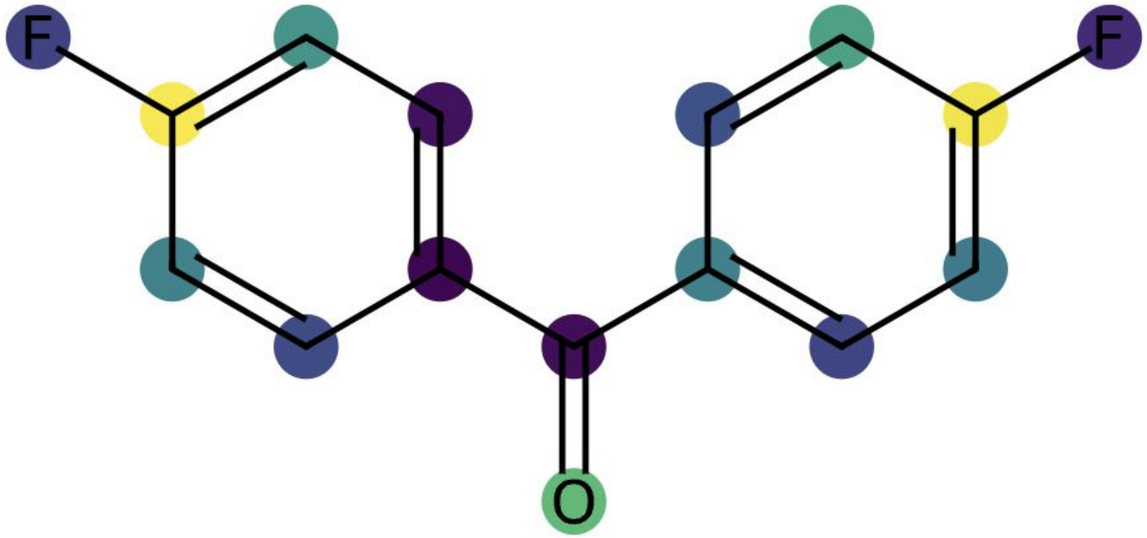
Bayesian Optimization



Evaluation

Model	RMSE	95% CI	R2	95% CI
XGBoost Regression	1.2430	[1.1644, 1.3171]	0.7297	[0.6927, 0.7641]
BioSolveAI GNN	0.4362	[0.4103, 0.4606]	0.7144	[0.6743, 0.7509]

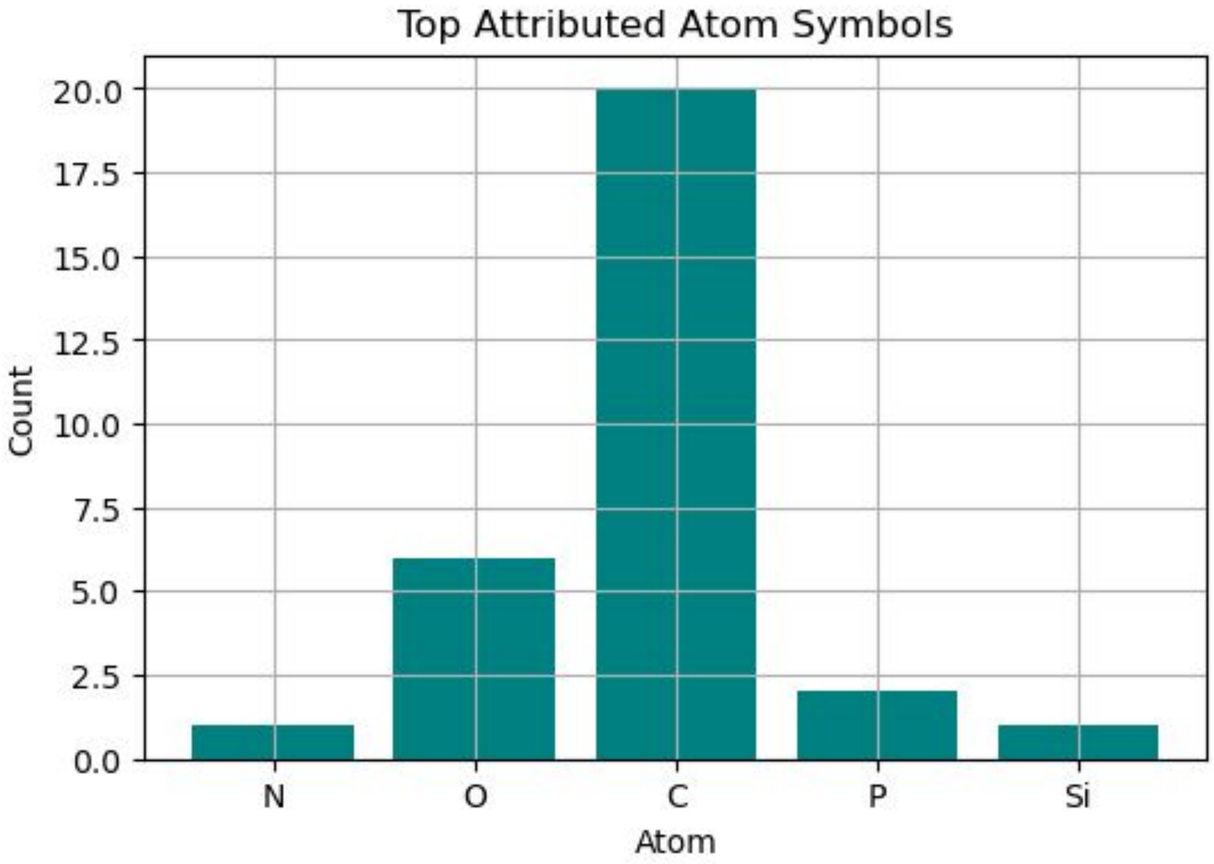
Interpretability - Single Sample

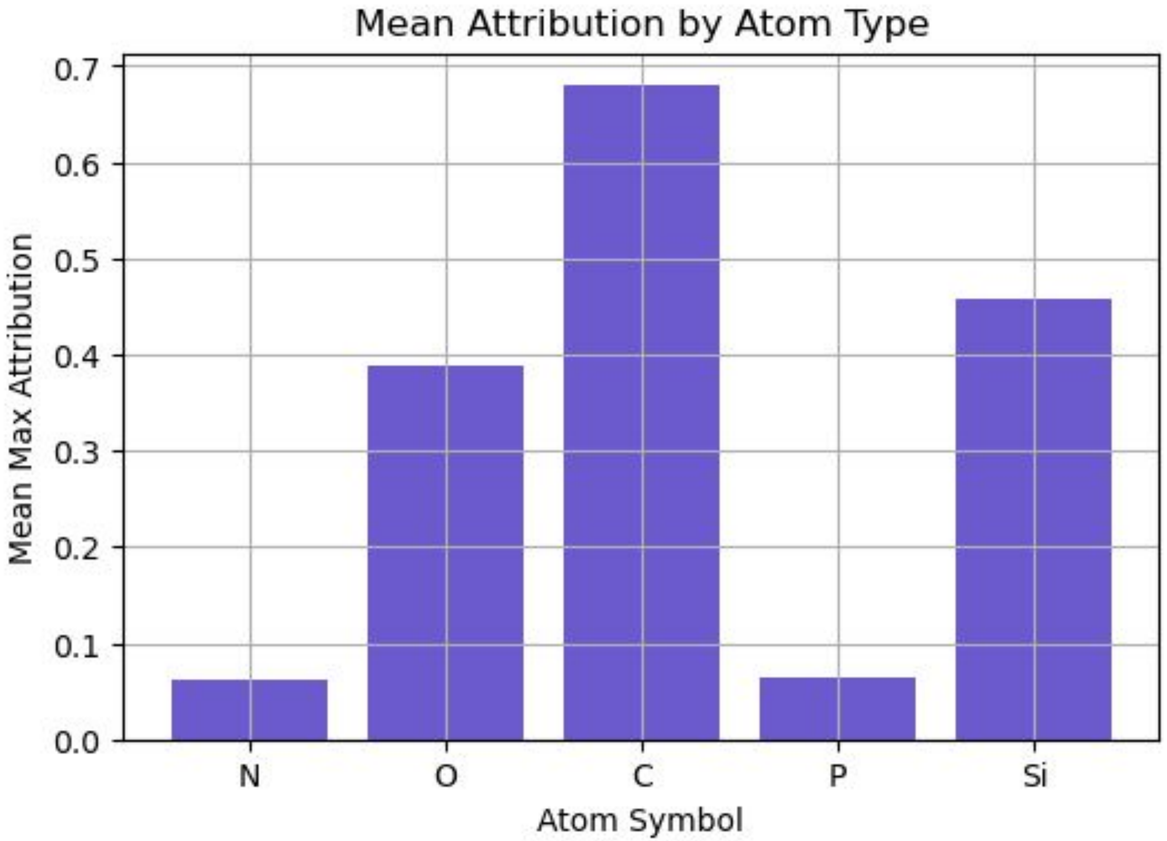


Sample Molecule

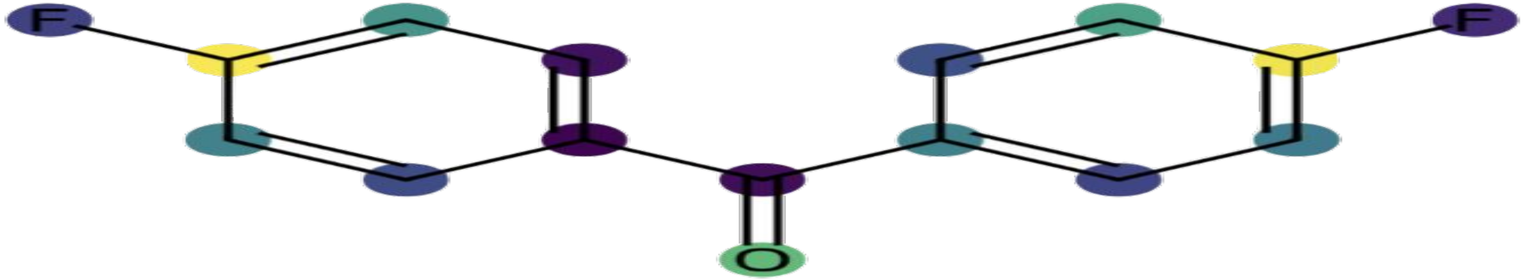
Prediction: -0.7181**, True Value:** -0.7026

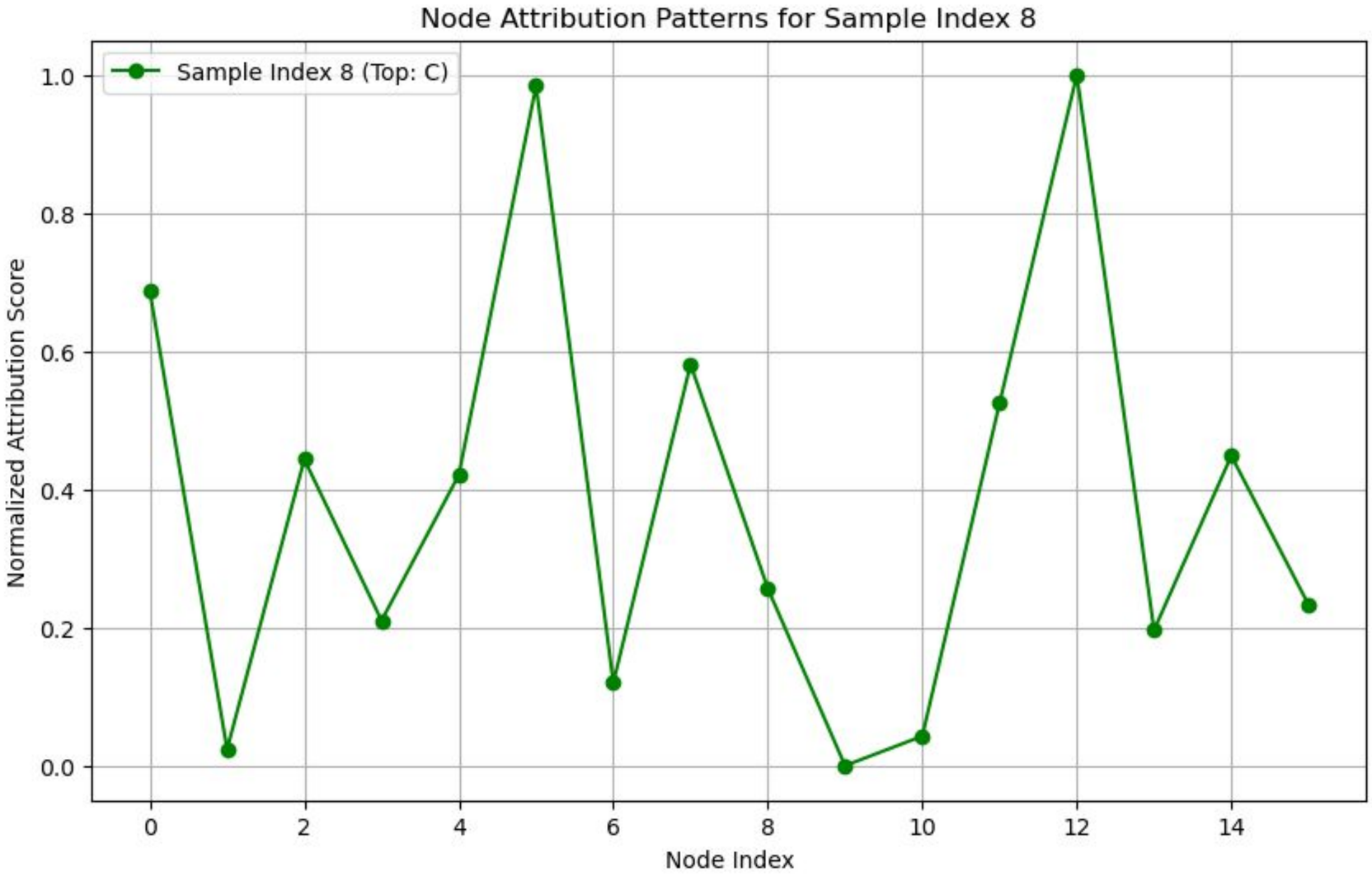
Interpretability - Multiple Samples





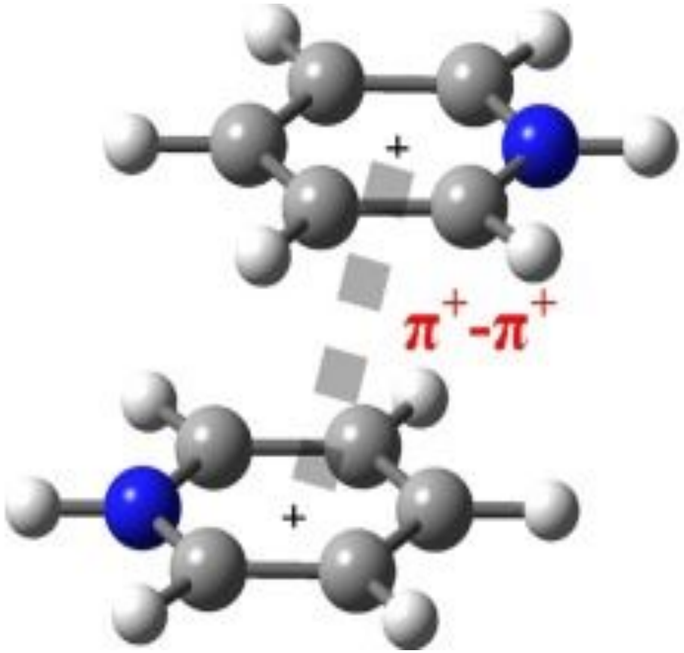
Interpretability - Node Attribution Pattern





Next Steps

Hypertune number of GINEConv layers



Account for intermolecular forces