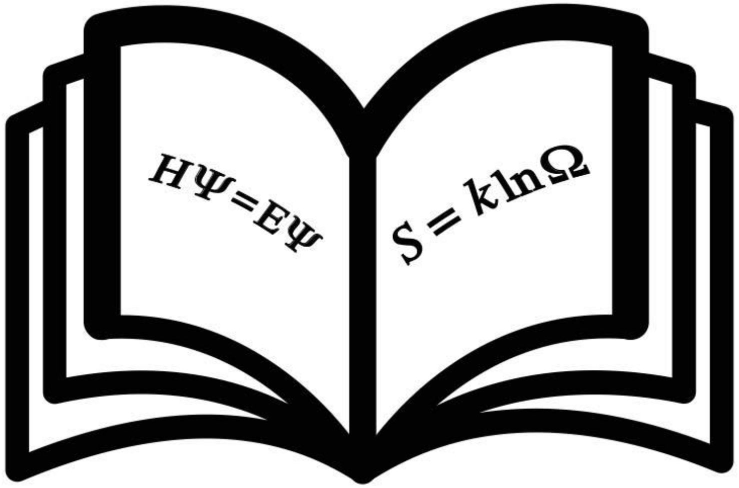
iCOMSE: Machine Learning in Molecular Science

Professor Camille Bilodeau

University of Virginia

May 1st 2025



How do we know whether our model is doing well?

- Answer: Test set performance
- Why do we care?
  - We want to know how closely our model approximates the ”ground truth function” that relates inputs and outputs
  - We want to know how well our model will perform when deployed for real world problems

Where does model error come from?

1. **The ground truth function can be represented within our neural network, but we can't find the weights because:**
    - Our optimization scheme has not been sufficient to arrive at the global loss optimum
    - We don't have enough training data to constrain the optimizer
    - Our training data has too much uncertainty and/or noise to constrain the optimizer
2. **The ground truth function is not represented within our neural network**
    - A larger and/or different architecture is required to represent the function
    - The ground truth function does not exist
3. **Our test set has too much uncertainty, preventing us from knowing whether or not we have found the ground truth function**

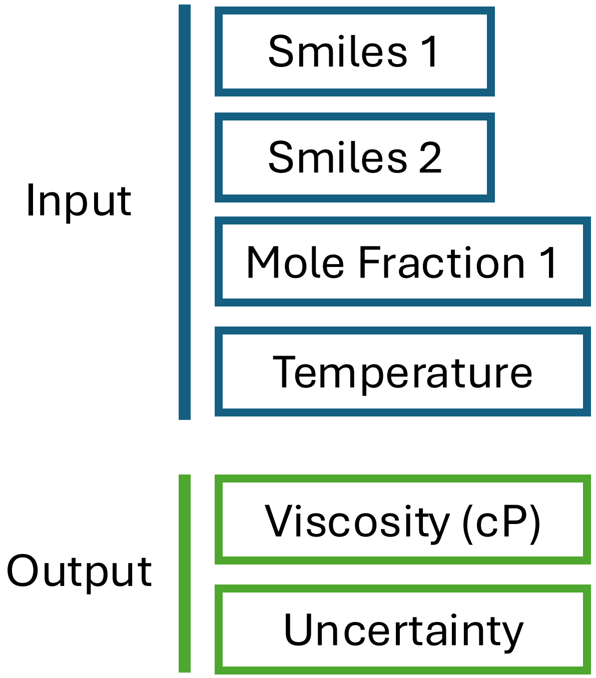
Splitting the dataset

- For most applications, the dataset will be split into training , validation, and testing sets often 80%, 10%, 10% respectively (though this may vary for some applications)
  - Training Set- model weights are updates on the basis of losses calculated using training set samples
  - Validation Set- hyperparameters choices are established by evaluating performance on a validation set
  - Test Set- final model performance is evaluated using the test set

- How do we decide which samples go in each set?
  - Random sampling
  - Scaffold (chemistry-based) sampling
  - Temporal sampling

Combinatorial Data Splitting

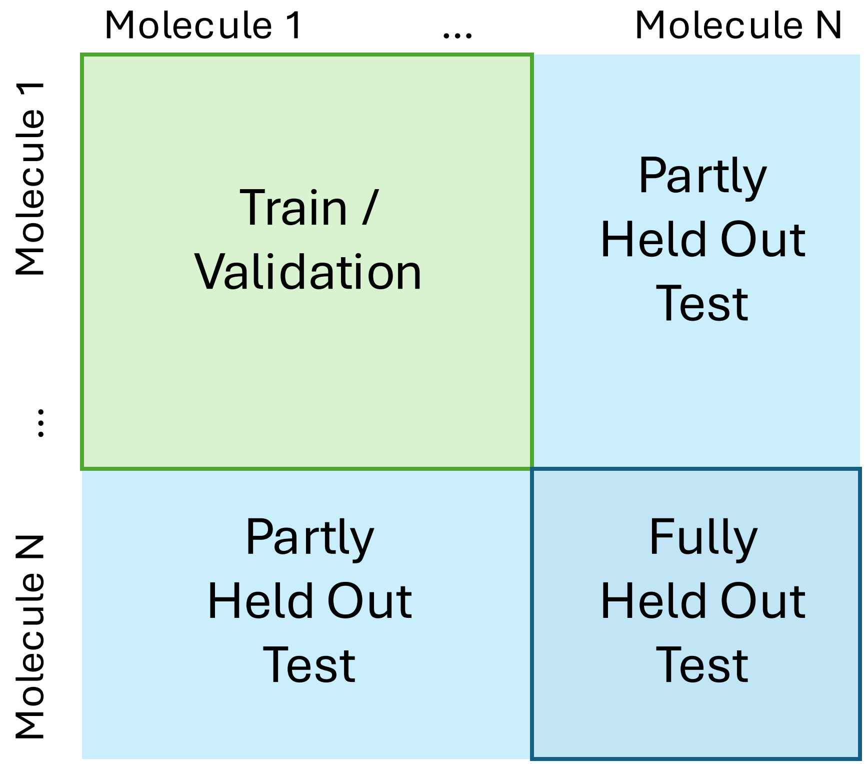
Typical Datapoint:



Use Case Scenarios:

- Both molecules are present in the training set
- One molecule is present in the training set and the other is not
- Neither molecule is present in the training set

Splitting Strategy:



Note: More complex data splitting strategies are needed ***any time your dataset contains non-independent data***.

Data Balancing

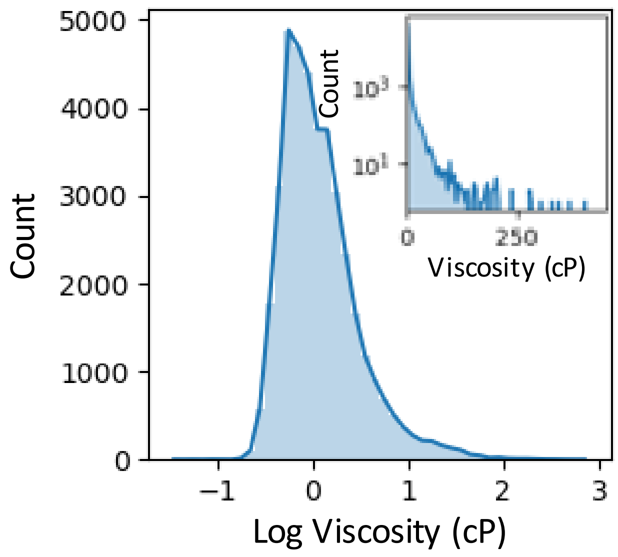
- Balanced datasets are required for learning functions in an unbiased way:
  - Learning cats and dogs
  - Bias in facial recognition
- Data balancing should be considered with respect to both the input and output representations of the data

Data Balancing: Categorical Data

- Scenario 1: Input data belongs to several discrete classes
- Scenario 2: Balancing positives and negatives in a binary prediction problem
- Strategies for dealing with an imbalanced dataset
    - Over-sampling
    - Under-sampling
    - Weighting

Data Balancing: Continuous Data

- Scenario 3: The variable you are predicting follows a skewed distribution (example from my recent viscosity paper)



Model Regularization

- Regularization is a class of techniques that involve modifying the learning algorithm to improve generalization and reduce overfitting
  - Early Stopping- use validation set performance to decide when to stop model training (in terms of epochs)
  - Dropout- randomly remove certain nodes during training with a specific probability
    - Outputs are typically scaled so that the magnitudes of each latent vector are not affected
  - L1 & L2 Regularization- update the cost function that penalizes nonzero weight values
  - More on regularization:

https://www.analyticsvidhya.com/blog/2018/04/fundamentals-deep-learning-regularization-techniques/