MITACS-2022: Diverse Subset Selection in Machine Learning
Ayer's Lab, McMaster University Hamilton

AIM:

1. Create a dataset of Antiviral drugs that are blood brain permeable
2. Perform performance profiling of all Subset selectors in the DiverseSelector package for different selection proportions and sizes of datasets
3. Prediction of logP on the PhysProp dataset to demonstrate the effectiveness of DiverseSelector

INTRODUCTION:

Selecting structural diverse compounds from a given molecule dataset not only represents a significant problem in molecule dataset design of drug screening, but also has may potential applications in computational drug design (e.g. selecting representative molecules for expensive quantum mechanical calculations followed by machine learning
model constructions). Therefore, we have implemented a set of subset selection algorithms in DiverseSelector (https://github.com/theochem/DiverseSelector).
There are 4 basic forms that have been implemented:

1. Directed Sphere Exclusion
2. Grid Partitioning
    a. Equisized Dependent
    b. Equisized Independent
3. MaxMin
4. MaxSum
5. OptiSim

There are variants for each type. For example, we can perform classification algorithms and use selection algorithms based on that. Also, there are some isoforms for some algorithms, e.g. "GridPartitioning".

OBJECTIVES:

1. Generate antiviral and blood brain barrier datasets from B3DB, ViPR, FluDB, Enamine and DrugBank

2. Profile 6 selectors by selecting proportions [2%, 5%, 10%.....95%, 99%] 10 times and reporting mean and average

3. Generate external test data from PhysProp (5%) using the MaxMin subset selector

4. Generate training sets using different DiverseSelector algorithms by holding out 2%, 5%, 10%.....95%, 99%, 100% of remaining data *(6 selectors x 15 hold out proportions)*

5. Train and tune 3 models on each training dataset using cross-validation:

    a. k-Nearest Neighbours (kNN)
    b. Gaussian Process Regression (GPR)
    c. Graph Neural Network (GNN)
6. Compute and plot MAE, RMSE and R^2 for each case – different models, different selectors, different proportions of data held out *(3 models x 6 selectors x 15 hold out proportions)*

PROCEDURE:

Part 1: Dataset Creation

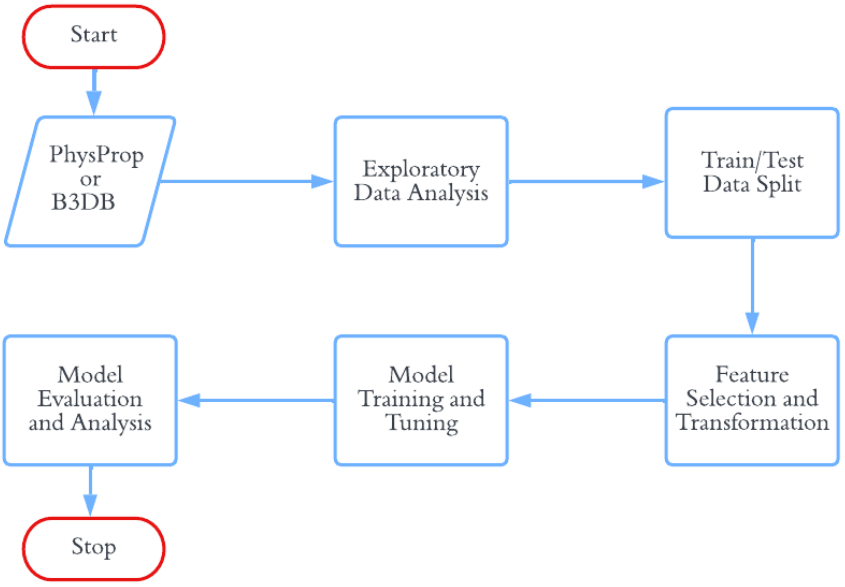
Creation of Antiviral Dataset:

1. Collect 4 datasets of antiviral drugs:
    a. Virus Pathogen Resource (ViPR)
    b. Enamine
    c. FluDB
    d. DrugBank
2. Using generic drug name, obtain SMILES formula and InchiKey from PubChem
3. Clean the molecules by removing salts and metal ions
4. Take the intersection of InchiKeys between an antiviral dataset and B3DB to obtain the final set of drugs

Part 2: Performance Profiling

1. Note the time to run 6 selectors on 14k molecule descriptors to select various proportions (6 selectors x 15 proportions)
2. Visualise the trends in time across algorithms and across proportions
3. Perform diversity calculations on the selected molecules using the following diversity metrics:
    a. "entropy",
    b. "explicit_diversity_index",
    c. "logdet",
    d. "shannon_entropy",
    e. "wdud",
    f. "total_diversity_volume",
    g. "gini_coefficient",

Part 3: Prediction of logP



Perform the following on the PhysProp dataset (14k x 507) and randomly generated matrices (507 x 10^k, where k = 1, 2, 3, 4, 5)

1. Exploratory Data Analysis:
    a. Check for missing values and impute if necessary
    b. Plot the distribution of each feature
    c. Check for correlations and relationships among features/output
    d. Look for outliers in the data
2. Train Test Split:
    a. Choose 5% of the data randomly for test
    b. Generate training sets using different DiverseSelector algorithms by holding out 2%, 5%, 10%.....95%, 99%,100% of remaining data *(6 selectors x 15 hold out proportions)*
    c. Choose validation set from the remaining data to choose between models
3. Feature Selection and Transformation
    a. Transform skewed features (according to 1(b)) and normalise all features

    (Note: Transform test data using the same parameters as training set)
    b. Feature extraction: Create new features from existing ones if they provide any new information (according to domain knowledge)

c. Drop highly correlated features (threshold to be decided based on number of features, or performance) (based on training set)

d. Perform feature selection (based on training set) (https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/)

4. Model Training/Tuning and Selection:

   a. kNN: Use normalised and transformed data to obtain predictions on test data; Try out various values of k.

   b. GPR:

      i. Use SMILES molecules and/or descriptor data as input

      ii. Choose kernel for Gaussian Process: Exponential, Matern, Custom Tanimoto or Jaccard kernel

      iii. Obtain Morgan fingerprints for molecules (Choose Number of bits and bond radius)

      iv. Build Gaussian Process model using GPFlow (like this post)

      v. Tune hyperparameters using GPFlowOpt

   c. GNN:

      i. Convert molecules to graphs (https://stackoverflow.com/questions/70459042/convert-a-smiles-dataset-to-graph)

      ii. Tune hyperparameters using Optuna (https://optuna.org/)

   d. (Selecting between models) Check learning curves – RMSE, MAE and R^2 (for Regression only) for different models on validation set. Choose the best performing model

5. Model Evaluation: Compute and plot MAE and RMSE for each case – different models, different selectors, different proportions of data held out.

RESULTS:

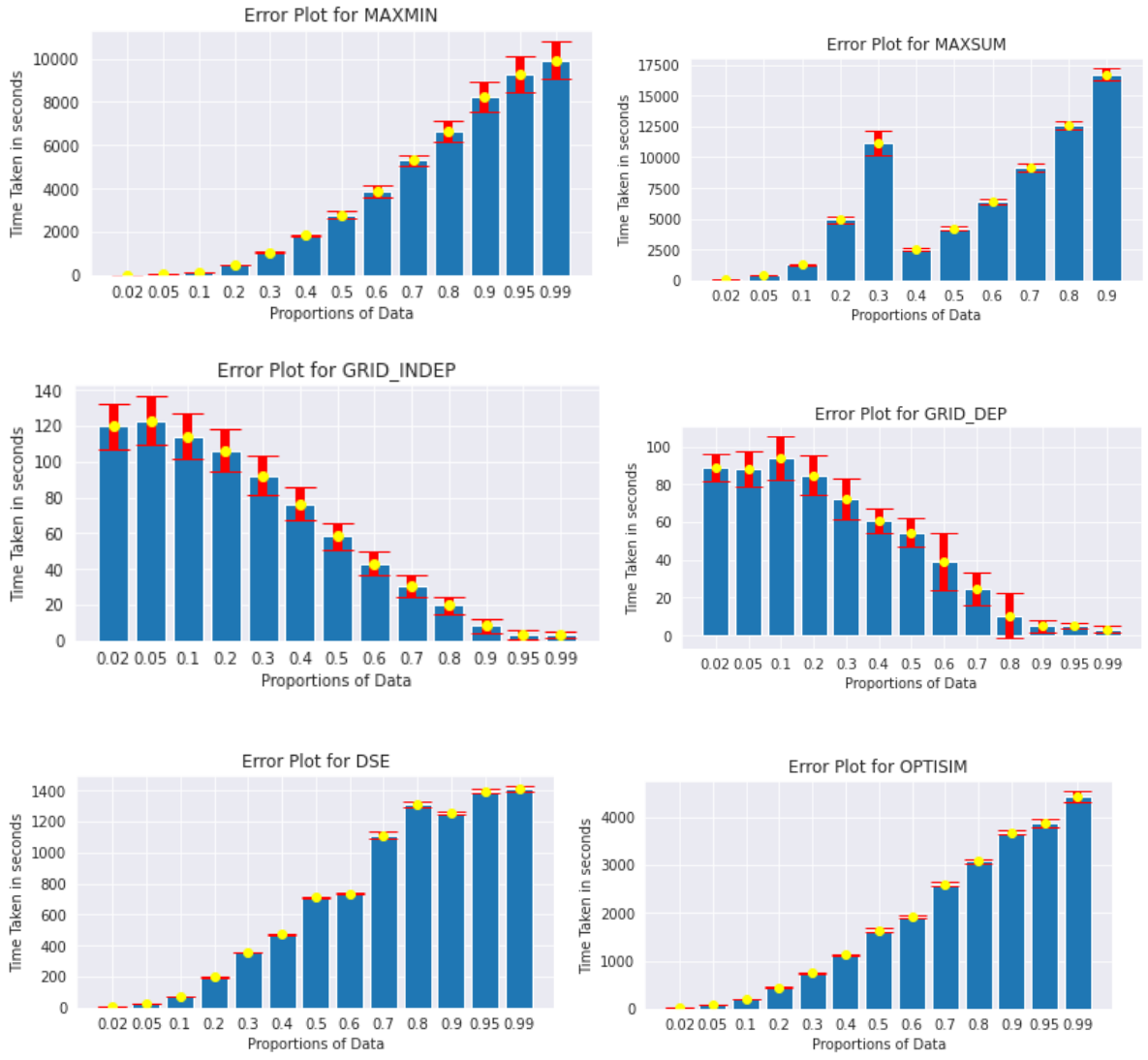
Part 1: Antiviral Blood Brain Permeable Dataset

Dataset	Total Number of Molecules	Number of common molecules
ViPR	72	36
Enamine	3200	0

FluDB	72	0
DrugBank	193	46

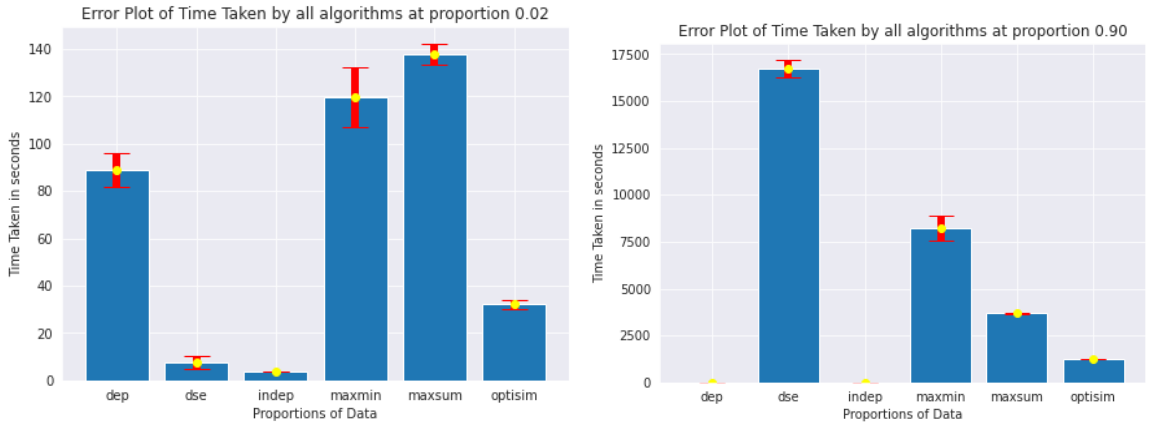
Part 2a: Profiling Diverse Selector on PhysProp

Plot of time taken for 10 iterations by each algorithm vs proportion of data selected



1. We observe that for MaxMin, MaxSum, Directed Sphere Exclusion and Optisim, as the proportion to be chosen increases, the time increases while for both variants of grid partitioning, as the proportion to be chosen increases, the time decreases
2. MaxMin, MaxSum and Optisim are seen to take the maximum amount of time for selection

Comparing time taken by different selectors for a small and large proportion



Selector	Ratio of Mean Running Times
dep	0.6462
dse	0.0555
indep	0.0267
maxmin	0.8684
maxsum	1.0000
optisim	0.2334

1. MaxMin, Maxsum and Optisim fare worse while picking small proportions than DSE as compared to large proportions
2. For large proportions, GridPartitioning is recommended

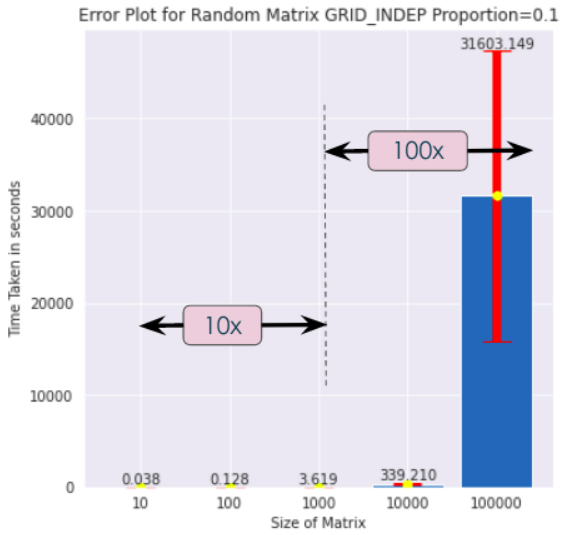
Proportion of Indices Common across Iterations in Selectors

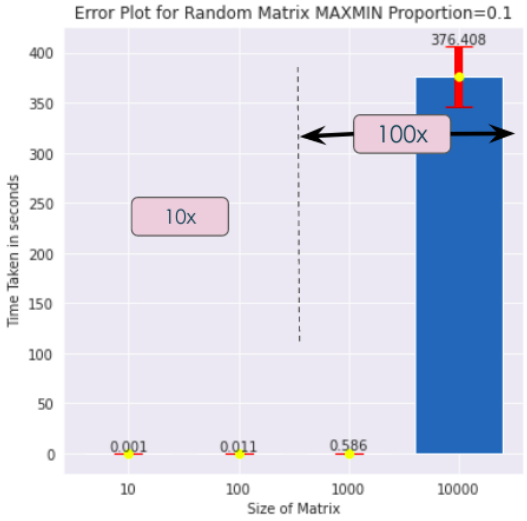
Selector	Proportion of Indices
dep	1.0000
dse	0.1246
indep	1.0000
maxmin	1.0000
maxsum	0.3801
optisim	0.1566

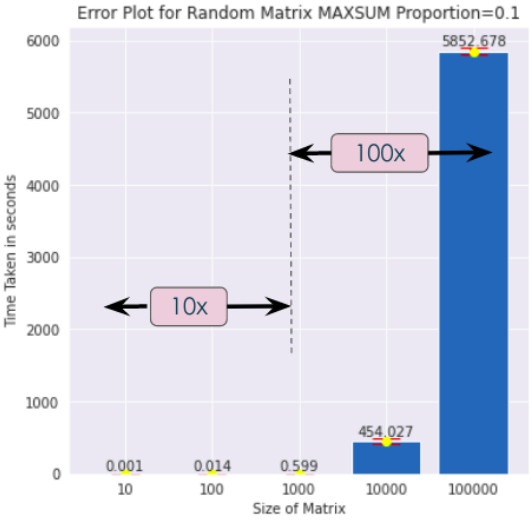
Across the 10 iterations, GridPartitioning, and Maxmin returned the same indices everytime.

Maxsum, Optimism and DSE returned between 12 and 40% common indices across the different iterations.

Part 2b: Profiling Diverse Selector on Large Random Matrices

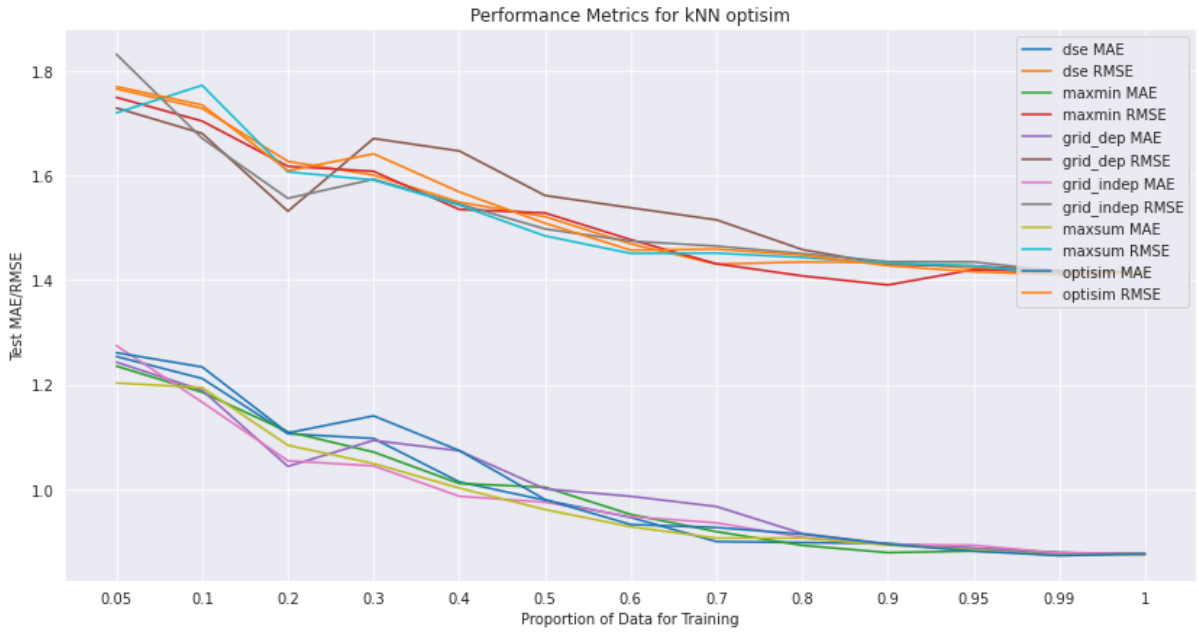




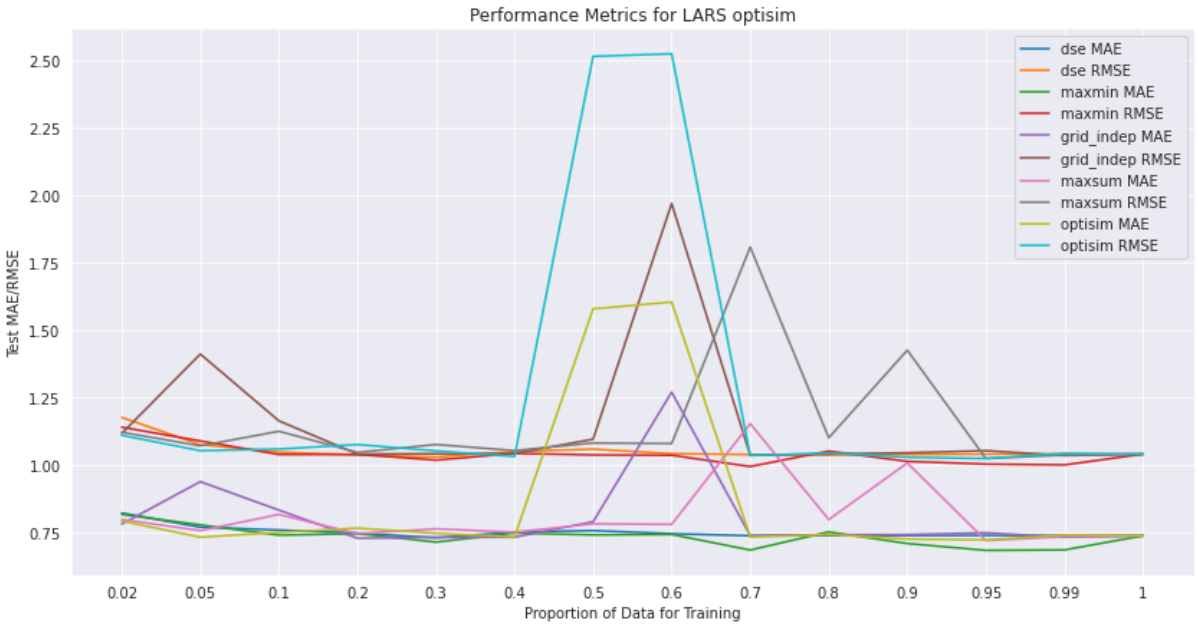


With smaller sizes of data, the time taken increases 10x as the size increases, however, at larger sizes it scales by a factor of 100. This may be due to caching effects that are at play for smaller data sizes.

Part 3: Prediction of logP


Performance Metrics for kNN optisim
Test MAE/RMSE
Proportion of Data for Training
dse MAE
dse RMSE
maxmin MAE
maxmin RMSE
grid_dep MAE
grid_dep RMSE
grid_indep MAE
grid_indep RMSE
maxsum MAE
maxsum RMSE
optisim MAE
optisim RMSE


Performance Metrics for SGD Regression optisim
Test MAE/RMSE
Proportion of Data for Training
dse MAE
dse RMSE
maxmin MAE
maxmin RMSE
grid_dep MAE
grid_dep RMSE
grid_indep MAE
grid_indep RMSE
maxsum MAE
maxsum RMSE
optisim MAE
optisim RMSE



1. The RMSE and MAE curves are seen to decrease as the proportion of indices selected increase, since the size of the dataset increase and the molecules become more representative of the entire dataset
2. For most algorithms, for a proportion of around 8.7-0.9 the RMSE/MAE drop below that of the entire dataset. This tells us that the remaining molecules may be noise.
