Supervised machine learning for Quantitative Structure-Activity Relationship modeling

Manpreet Kaur

DSCI-6003: Machine Learning & Data Analysis
Spring-21’

Contents

- Introduction & objectives
- Data Collection
- Data Preprocessing
- Data preparation - I
- Data set description
- Data Exploration
- Data preprocessing-II
- Feature selection/comparison
- Model complexity comparison
- Model evaluation and metrics
- Prediction
- Links and references

Introduction

- **Quantitative structure-activity relationship (QSAR)**

- machine learning - relationship between the chemical structure and the biological activity.

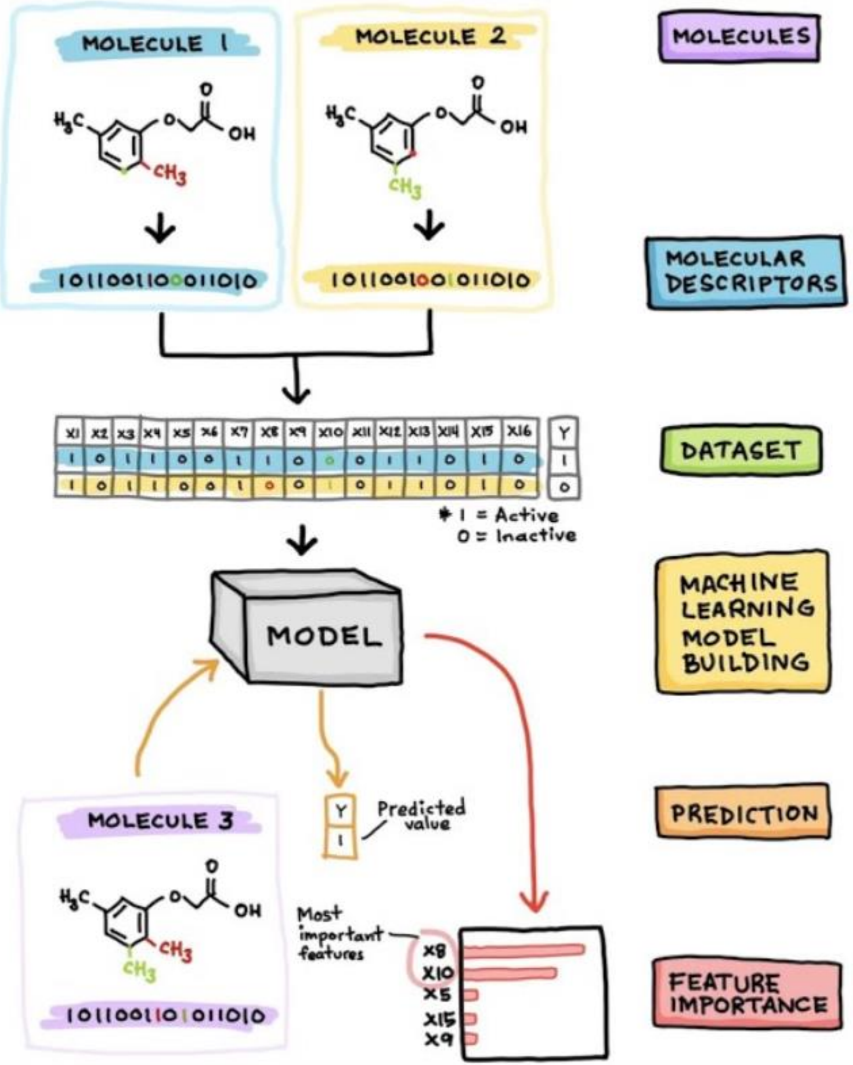
- The diagram here shows the workflow of the QSAR process
- Collection of molecules

- Calculation of molecular descriptors – physical and chemical properties

- prediction biological activity

- features important for biological activity

- biologists & chemists - design molecules- robust properties.



X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	Y
1	0	1	1	0	0	1	1	0	0	0	1	1	0	1	0	1
1	0	1	1	0	0	1	0	0	1	0	1	1	0	1	0	0

* 1 = Active
0 = Inactive

Objectives

- Creating different supervised machine learning models for classification of chemical compounds based on bioactivity data and molecular descriptors
- Comparison of the above models to find the best among them

Data Collection

Raw data source : https://www.ebi.ac.uk/chembl/target_report_card/CHEMBL1615382/

Target search: Breast cancer

Filters: organism type(Homo Sapiens), target_type(Single protein)

Chosen Target ID: CHEMBL5393

Sub-filters for the target: retrieve only bioactivity data that were reported as IC50 values in nM (nanomolar) unit.

Raw data collected above saved to bioactivity_data_raw.csv file

Data Preprocessing

- Preprocessing of raw data
  - handling missing values
  - Labeling compounds as either being active, inactive or intermediate – bioactivity_class

    The bioactivity data is in the IC50 unit. Compounds having values of less than 1000 nM will be considered to be active while those greater than 5,000 nM will be considered to be inactive. As for those values in between 1,000 and 5,000 nM will be referred to as intermediate.

  - Combine the 3 columns from raw data (molecule_chembl_id,canonical_smiles,standard_value) and bioactivity_class into a DataFrame

  - Saved preprocessed above data to a bioactivity_data_preprocessed.csv file

Data preparation–I

- Calculation of Lipinski Descriptors
  - Christopher Lipinski, a scientist at Pfizer, came up with a set of rule-of-thumb for evaluating the **druglikeness** of compounds. Such druglikeness is based on the Absorption, Distribution, Metabolism and Excretion (ADME) that is also known as the pharmacokinetic profile. Lipinski analyzed all orally active FDA-approved drugs in the formulation of what is to be known as the **Rule-of-Five** or **Lipinski's Rule**.

- The Lipinski's Rule stated the following
  - Molecular weight < 500 Dalton
  - Octanol-water partition coefficient (LogP) < 5
  - Hydrogen bond donors < 5
  - Hydrogen bond acceptors < 10

Data preparation–I

- Combine the data frame of the above 4 Lipinski descriptor with preprocessed bioactivity data into single data frame

- Convert IC50 values (standard_value column) to pIC50 – uniform distribution – negative logarithmic scale (the standard values needs to be normalized before converting to pIC50 values)

- Removing the 'intermediate' bioactivity class

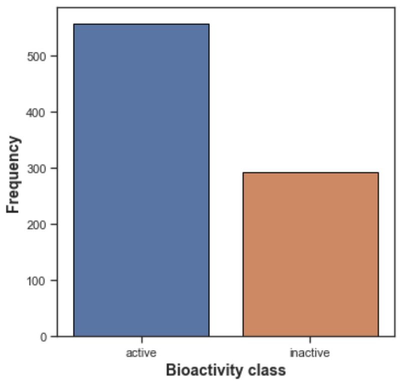
- Saved data to bioactivity_data_preprocessed_2-class.csv

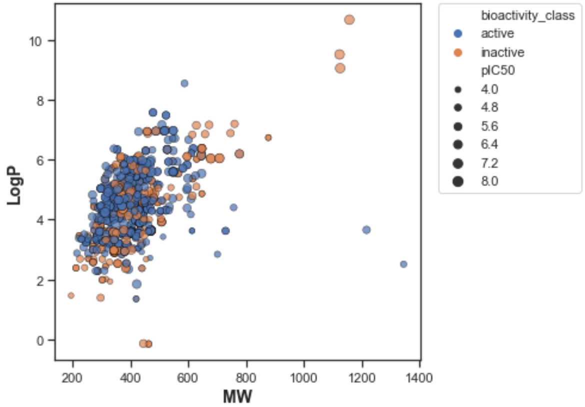
Data set description

Data features: 850 rows, 8 columns

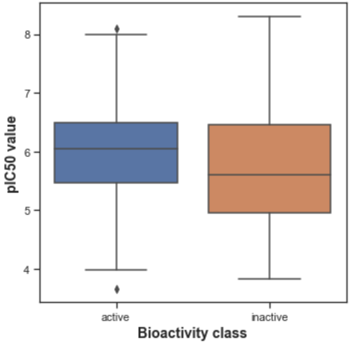
- molecule_chembl_id
- canonical_smiles
- MW
- LogP
- NumHDonors
- NumHAcceptors
- pIC50 active 558
- bioactivity_class (target) inactive 292

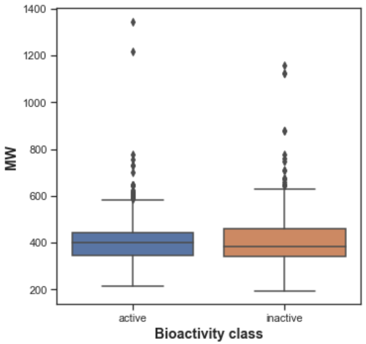
Data Exploration (Chemical Space Analysis)

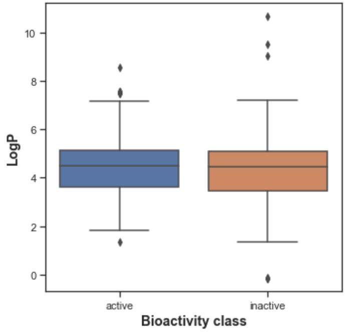


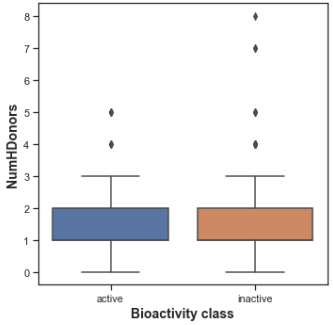


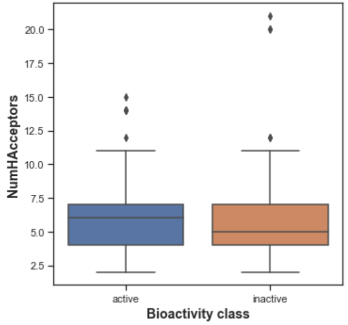
Descriptor	Statistics	p	alpha	Interpretation
LogP	74720.5	0.023579	0.05	Different distribution (reject H0)
MW	80292.5	0.364795	0.05	Same distribution (fail to reject H0)
NumH -Acceptors	77293	0.106799	0.05	Same distribution (fail to reject H0)
NumHDonors	78172	0.146315	0.05	Same distribution (fail to reject H0)
pIC50	66904.5	9.17E-06	0.05	Different distribution (reject H0)











Data preprocessing-II

- Removing non-essential / redundant features:
  - molecule_chembl_id only an identifier, index can be used
  - canonical_smiles structural information

- Data transformation:
  - Converting categorical data to numerical data

    Modify bioactivity_class Inactive- 0 ; Active – 1

  - Convert object data type to int

Feature selection/comparison

The correlation of various features with the target feature (Bioactivity_class) looks as below:

	Correlation with target
bioactivity_class	1.000000
pIC50	0.151300
NumHDonors	0.075819
LogP	0.071472
MW	0.037330
NumHAcceptors	0.003464

Decreasing correlation

Model complexity comparison

Models under consideration:

- Logistic Regression
- Random Forest
- Support Vector Machine
- KNeighbors
- Decision Tree
- XGBoost

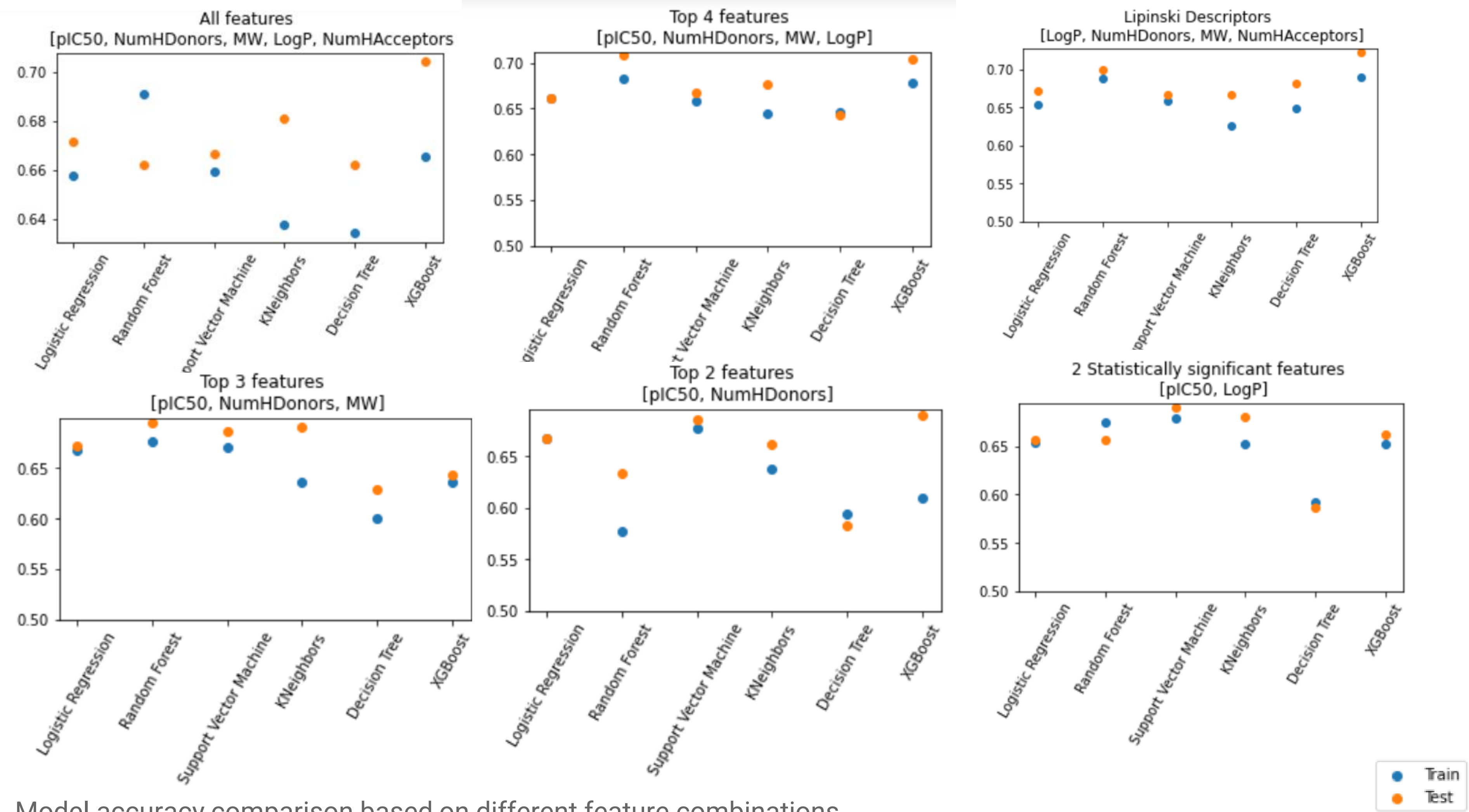
Number of features compared:

- All 5
- Top 4 highly correlated
- Top 3 highly correlated
- Top 2 highly correlated
- 2 statistically significant features (pIC50, LogP)
- Lipinski descriptor features

Order of correlation w.r.t target:

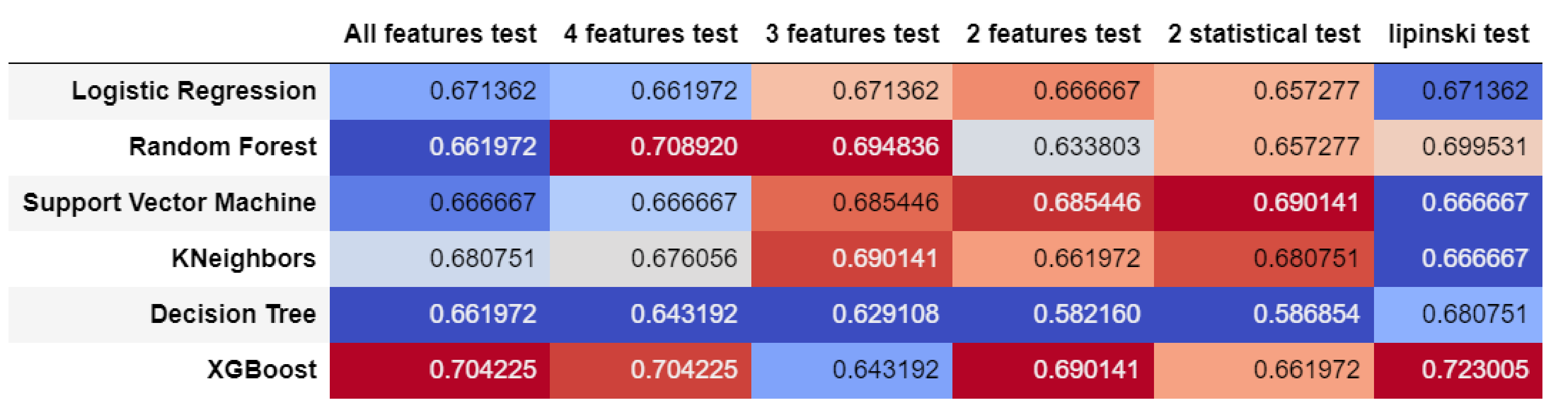
pIC50> NumHDonors > LogP > MW > NumHAcceptors

Model comparison using 5-fold cross validation - for a combination of 36 different models



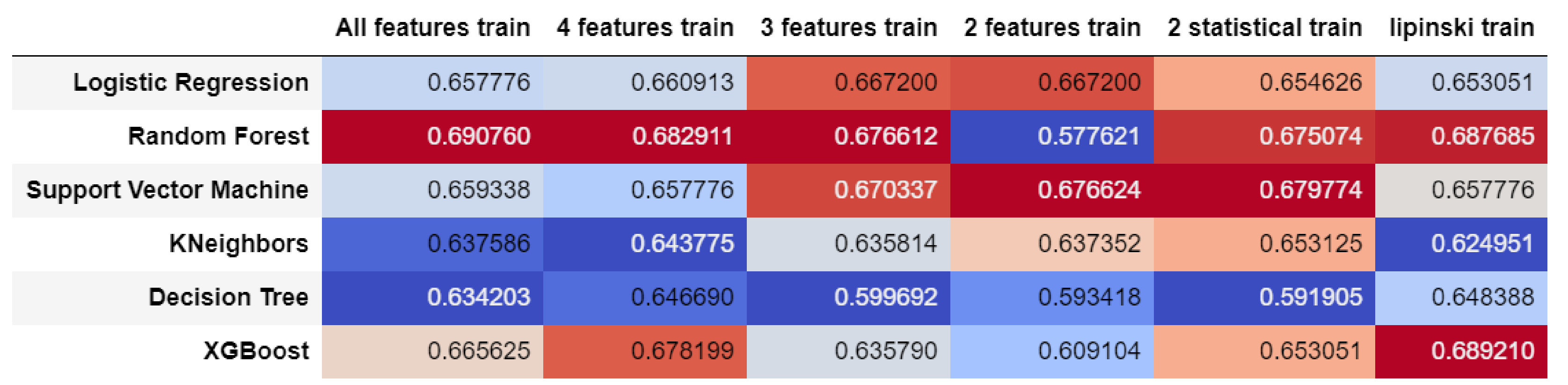
Model accuracy comparison based on different feature combinations

Test accuracy comparison

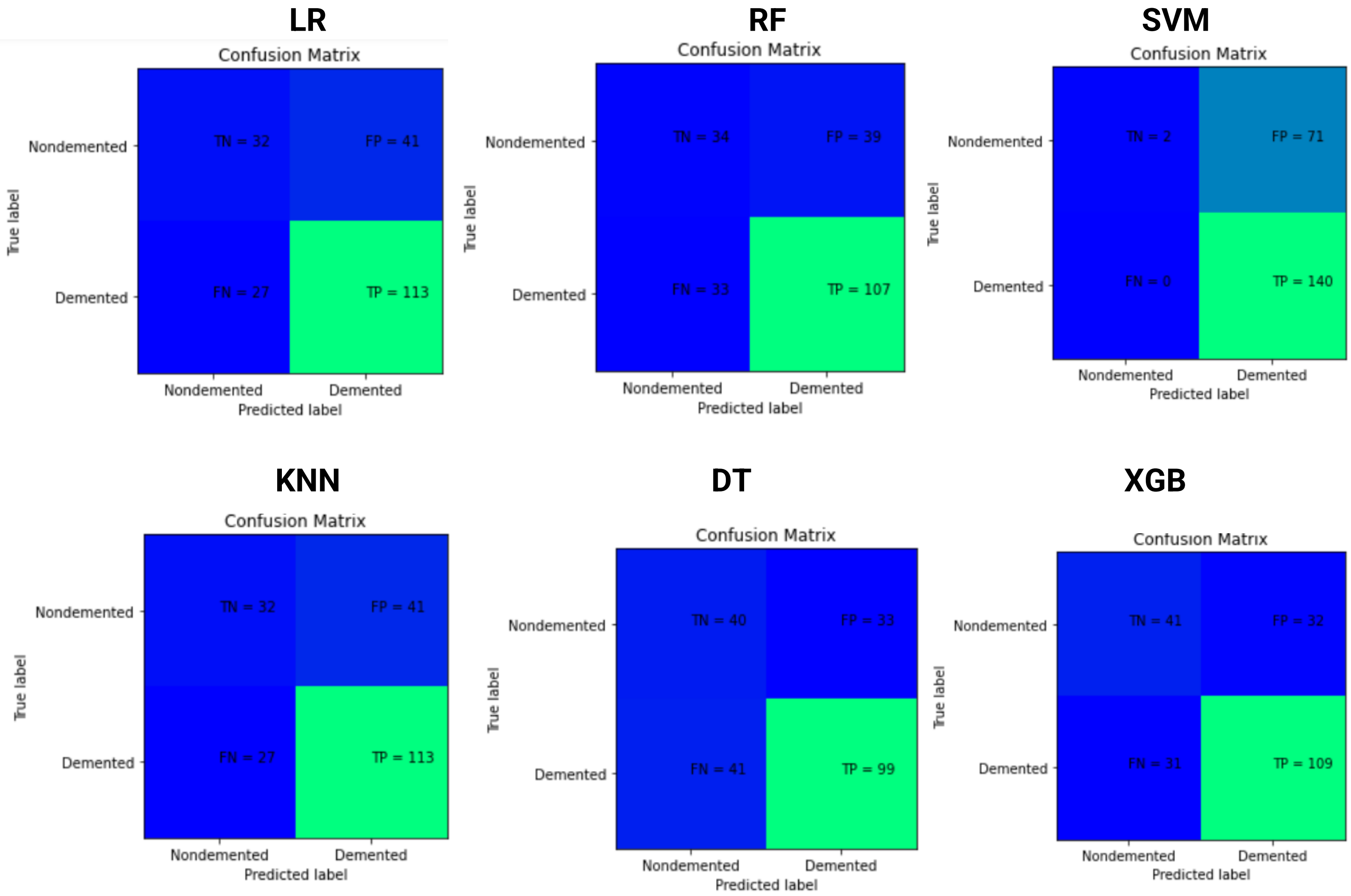


	All features test	4 features test	3 features test	2 features test	2 statistical test	lipinski test
Logistic Regression	0.671362	0.661972	0.671362	0.666667	0.657277	0.671362
Random Forest	0.661972	0.708920	0.694836	0.633803	0.657277	0.699531
Support Vector Machine	0.666667	0.666667	0.685446	0.685446	0.690141	0.666667
KNeighbors	0.680751	0.676056	0.690141	0.661972	0.680751	0.666667
Decision Tree	0.661972	0.643192	0.629108	0.582160	0.586854	0.680751
XGBoost	0.704225	0.704225	0.643192	0.690141	0.661972	0.723005

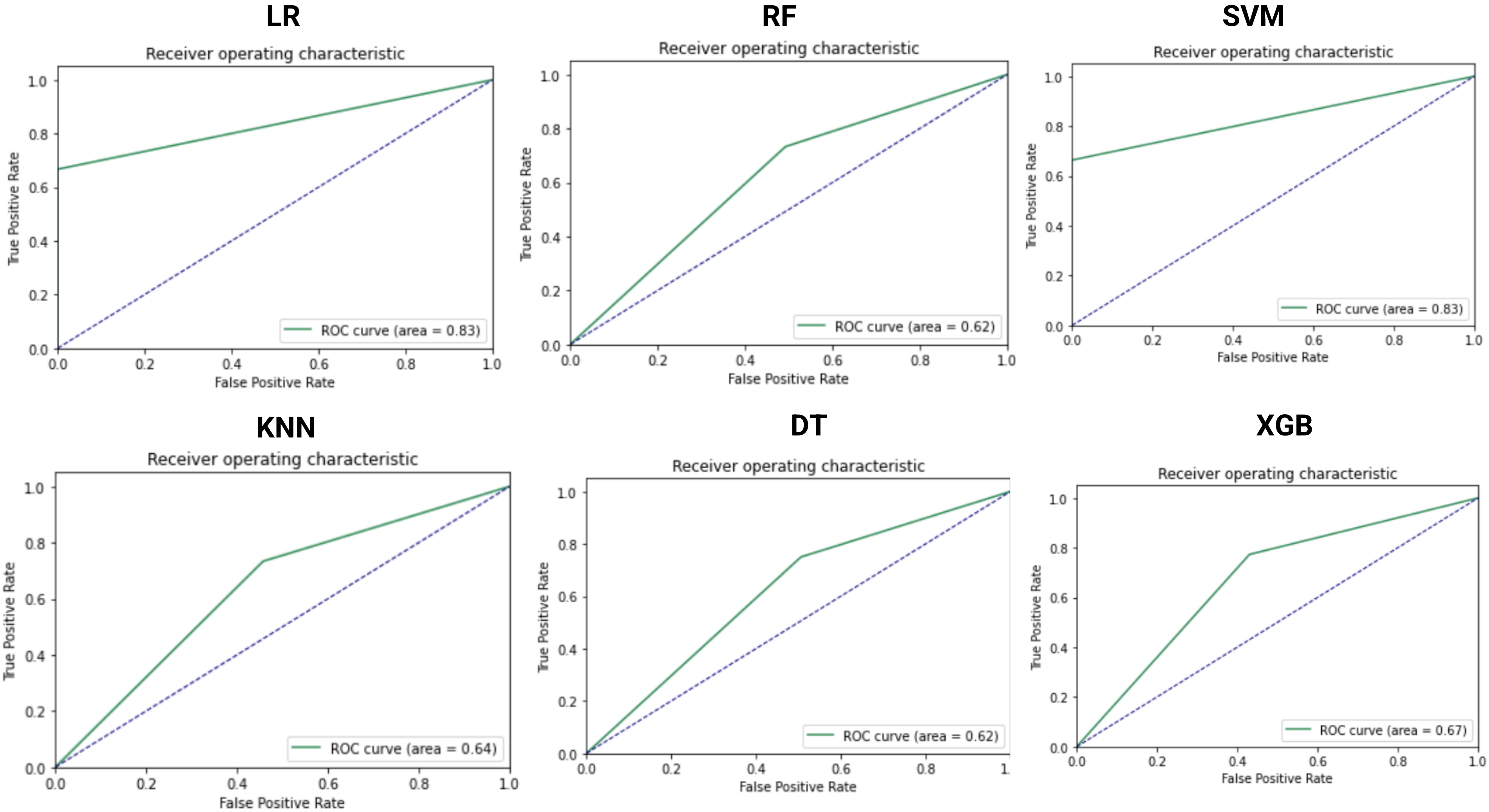
Train accuracy comparison



	All features train	4 features train	3 features train	2 features train	2 statistical train	lipinski train
Logistic Regression	0.657776	0.660913	0.667200	0.667200	0.654626	0.653051
Random Forest	0.690760	0.682911	0.676612	0.577621	0.675074	0.687685
Support Vector Machine	0.659338	0.657776	0.670337	0.676624	0.679774	0.657776
KNeighbors	0.637586	0.643775	0.635814	0.637352	0.653125	0.624951
Decision Tree	0.634203	0.646690	0.599692	0.593418	0.591905	0.648388
XGBoost	0.665625	0.678199	0.635790	0.609104	0.653051	0.689210



TN - True Negative
TP - True Positive
FN - False Negative
FP - False Positive


LR
Receiver operating characteristic
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (area = 0.83)
RF
Receiver operating characteristic
True Positive Rate
False Positive Rate
ROC curve (area = 0.62)
SVM
Receiver operating characteristic
True Positive Rate
False Positive Rate
ROC curve (area = 0.83)
KNN
Receiver operating characteristic
True Positive Rate
False Positive Rate
ROC curve (area = 0.64)
DT
Receiver operating characteristic
True Positive Rate
False Positive Rate
1.
ROC curve (area = 0.62)
XGB
Receiver operating characteristic
True Positive Rate
False Positive Rate
ROC curve (area = 0.67)

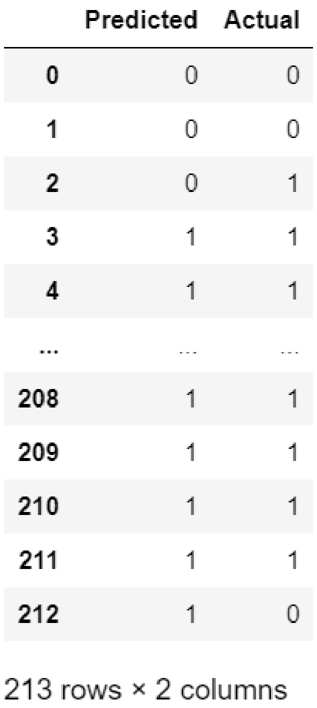
Model evaluation– metrics*

	LR	RFC	SVM	KNN	DT	XGB
Accuracy	0.6714	0.662	0.667	0.6808	0.6526	0.7042
Precision	0.6667	0.7329	0.6635	0.7338	0.75	0.773
Recall	1.0	0.7643	1.0	0.8071	0.7071	0.7786
F1-score	0.8	0.7483	0.7977	0.7687	0.7279	0.7758
AUC	0.83	0.62	0.83	0.64	0.62	0.67

Model with best metrics- Logistic Regression

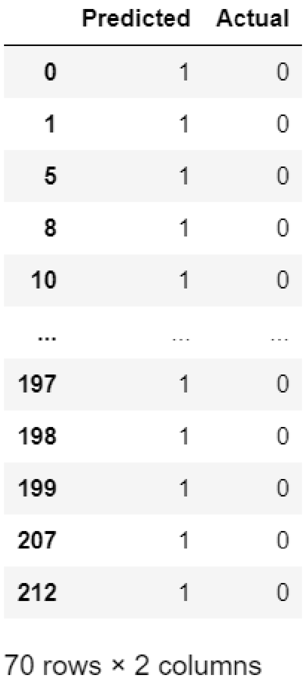
*metrics are calculated mathematically from the scores in the confusion matrix

Prediction



	Predicted	Actual
0	0	0
1	0	0
2	0	1
3	1	1
4	1	1
...	...	...
208	1	1
209	1	1
210	1	1
211	1	1
212	1	0

213 rows × 2 columns



	Predicted	Actual
0	1	0
1	1	0
5	1	0
8	1	0
10	1	0
...	...	...
197	1	0
198	1	0
199	1	0
207	1	0
212	1	0

70 rows × 2 columns

- XGBoost
  - Total number of wrongly predicted = 63 (out of 213~ 29.6%)
- LR
  - Total number of wrongly predicted = 70 (out of 213~ 32.86%)

LR Model equation

Y = -0.8327 - 0.0021(x1) + 0.1491(x2) - 0.0616(x3) + 0.07204(x4) + 0.2327(x5)

where,

Y = bioactivity_class
x1 = MW
x2 = LogP
x3 = NumHDonors
x4 = NumHAcceptors
x5 = pIC50

Thank you!

Special thanks – Prof. Travis Millburn

References

https://www.ebi.ac.uk/chembl/target_report_card/CHEMBL1615382/

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2728118/

https://www.semanticscholar.org/paper/A-practical-overview-of-quantitative-relationship-Nantasenamat-Isarankura-Na-Ayudhya/57832ea3bafae3f8e862c5da1ce5698b7627c47e

https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc

https://scikit-learn.org/0.18/auto_examples/model_selection/plot_confusion_matrix.html