purifAI [DRAFT]

Machine Learning for SPE Method Prediction

Created by Jen Amis, Luke Perrin, Teresa Tran, Yingying Cheung

Sections

1. Introduction & Context/Problem (2 mins) –Luke
2. Data, Data Source, Database (3 mins) –Luke
3. ML Models & Strategies (2 mins) –Jen/Yingying
4. ML Model Results (Visualizations) (3 mins) –Jen/Yingying/Teresa?
5. Conclusion (Dashboard and application function) (< 2 mins) –Teresa?

Overview

The team at an automated chemistry platform that works to automate the process of making small chemical compounds to be used in research and development for medicinal purposes is seeking a **machine learning model** that can be used to **select the best SPE method** to test for purification of each chemical compound in a large library of compounds.

Without a ML model that can effectively predict the optimal SPE method to use, the team must make a best guess of which method to test based on a subset of properties of each compound's structure. This process can be **time consuming** and **expensive**, especially if the wrong SPE method ends up being selected and the purification testing must be repeated using the other method.

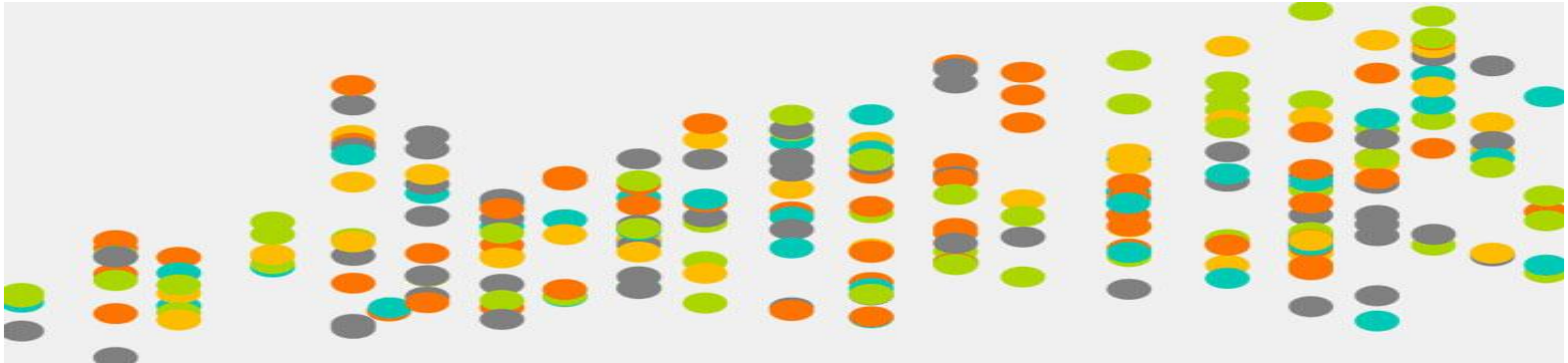
Development of a ML model has the potential to **save time & cost effective** in the automated chemistry platform's process.

Goals

1. Identify features and target for ML model
2. Identify a ML model with best prediction performance for SPE and LCMS method
3. Test the confidence level
4. Create an interactive User Interface for public

Data Source

This project utilizes datasets provided by the data team at the automated chemistry platform. The first dataset lists compounds tested by the platform over the past two years and includes compound properties such as molecular weight, topological polar surface area (TPSA), quantitative estimate of drug-likeness (QED), among many others that may be relevant to predicting the appropriate SPE method to use for compound purification. The second dataset includes the status of testing for each compound and the SPE method used for each compound that has completed the purification stage. Each compound is identified by a unique structure ID, and proprietary information about the actual structure of the compound has been excluded from the datasets.



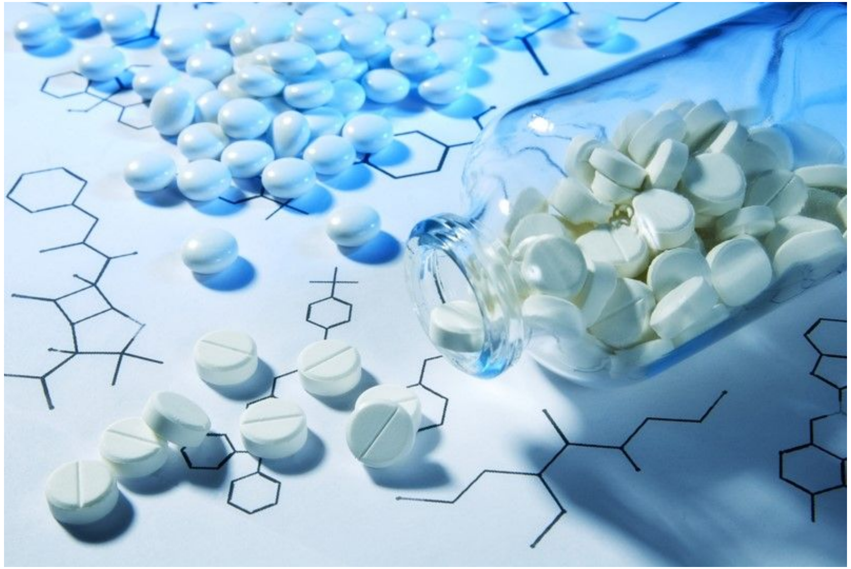
Database

A relational database (RDS) was created in Amazon Web Services (AWS), and connected to pgAdmin14. This Postgres database is hosted on the cloud, which can be accessed by anyone with credentials using pgAdmin14. Data was cleaned by Pandas, and stored in AWS S3 bucket. We call the data from RDS by using SQLAlchemy.

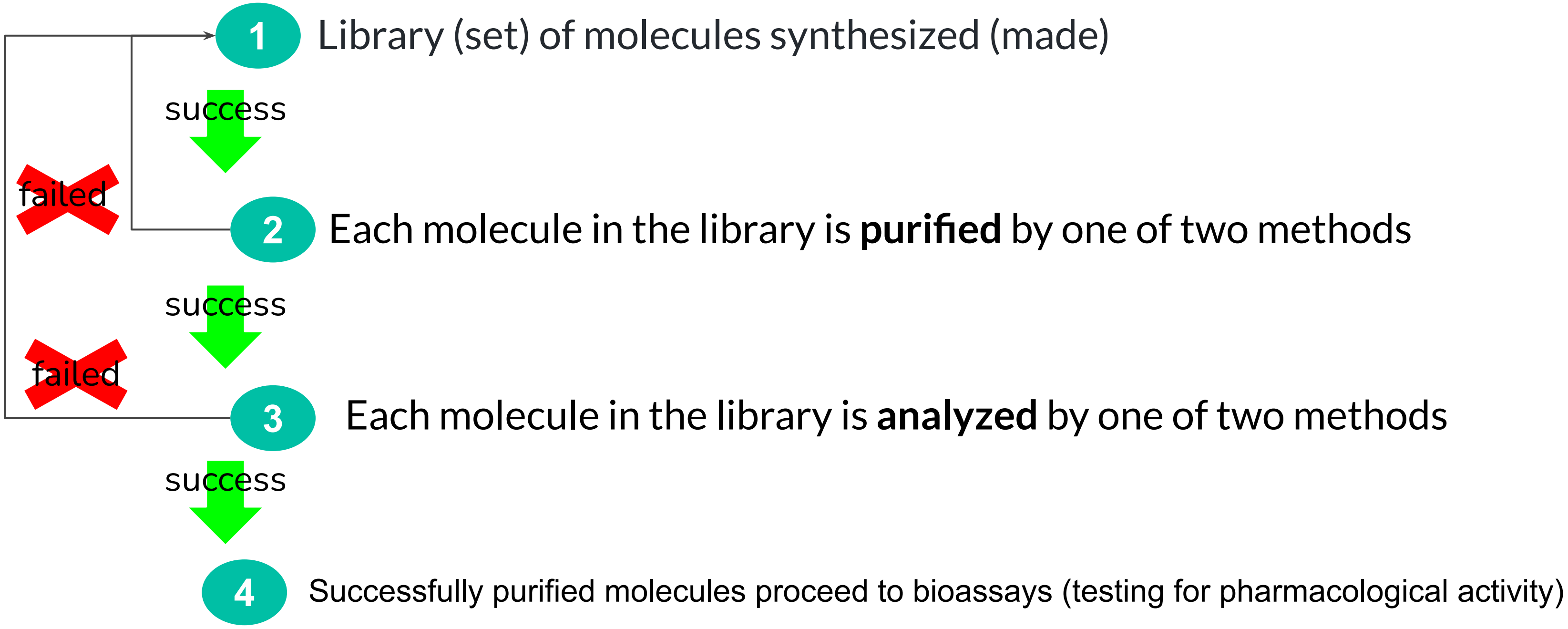
About the Data

Context & Problem

- The development of a completely automated chemistry platform is of high importance for the drug discovery process
    - By automating chemical synthesis and purification, many **varieties of novel drug candidates (or chemical structures)** can be tested for pharmacological activity
- Chemistry as a scientific field/practice is inherently **unpredictable and (seemingly) inconsistent,** and thus **very difficult to predict and automate**



Chemistry Purification Workflow Overview



1 Library (set) of molecules synthesized (made)

success

failed

2 Each molecule in the library is **purified** by one of two methods

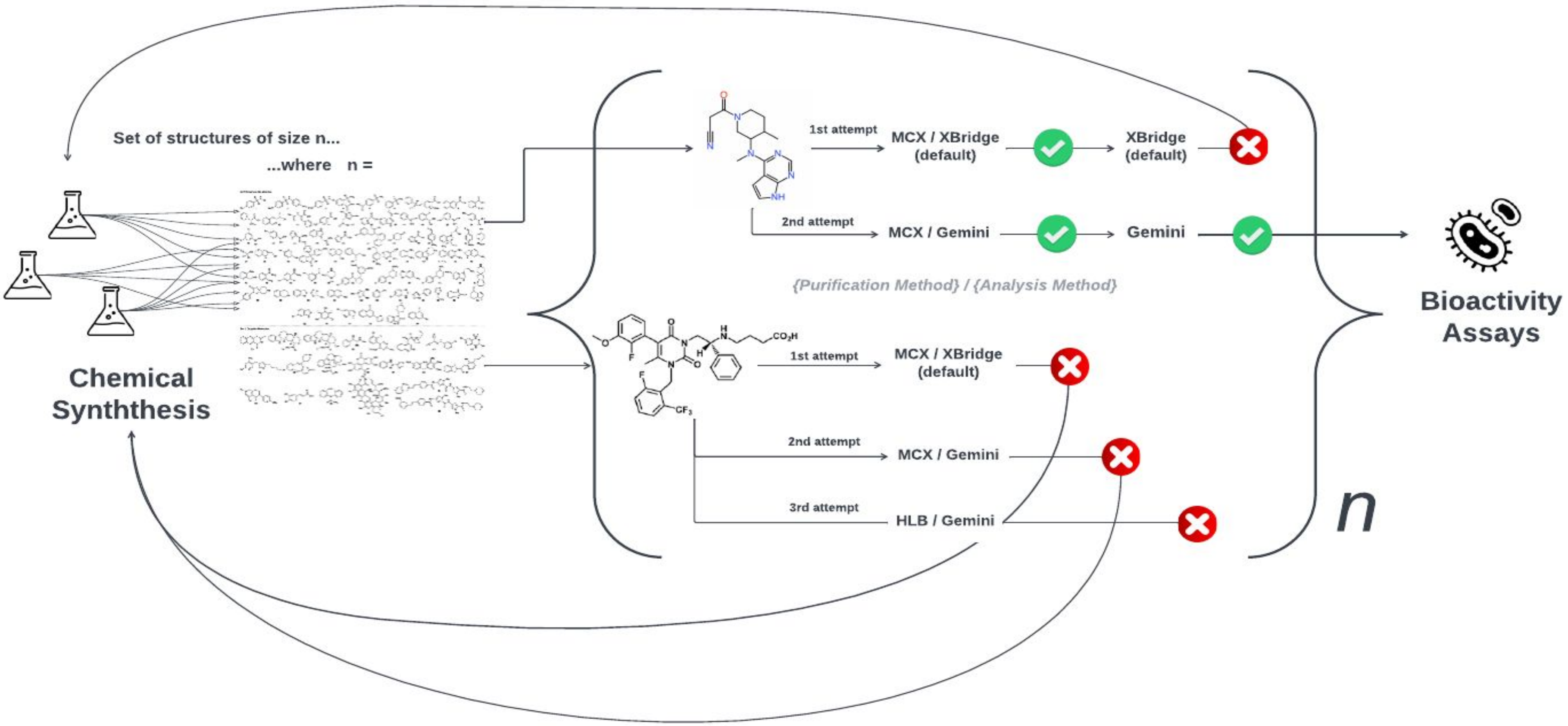
success

failed

3 Each molecule in the library is **analyzed** by one of two methods

success

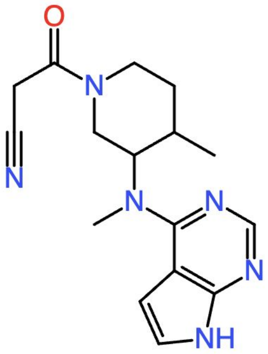
4 Successfully purified molecules proceed to bioassays (testing for pharmacological activity)


Set of structures of size n...
...where n =
Chemical Synththesis
1st attempt
MCX / XBridge (default)
XBridge (default)
2nd attempt
MCX / Gemini
Gemini
{Purification Method} / {Analysis Method}
1st attempt
MCX / XBridge (default)
2nd attempt
MCX / Gemini
3rd attempt
HLB / Gemini
n
Bioactivity Assays

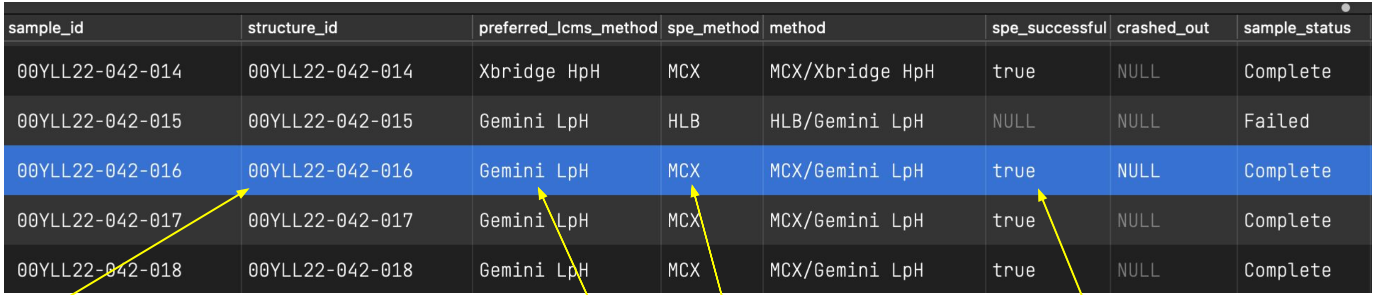
Data Sources

1. **Purification Outcomes**
   a. Successfully purified (binary/boolean) via either SPE method:
      i. MCX
      ii. HLB
   b. Successfully analyzed (binary/boolean) via either LCMS method:
      i. Xbridge
      ii. Gemini
2. **Chemical Structure Data**
   a. Data/metrics calculated from chemical structures (purified on the automated platform)
   b. These describe chemical properties of each molecule

Purification “Outcomes”



Example **successfully purified** molecule
(structure_id = 00YLL-042-016)



sample_id	structure_id	preferred_lcms_method	spe_method	method	spe_successful	crashed_out	sample_status
00YLL22-042-014	00YLL22-042-014	Xbridge HpH	MCX	MCX/Xbridge HpH	true	NULL	Complete
00YLL22-042-015	00YLL22-042-015	Gemini LpH	HLB	HLB/Gemini LpH	NULL	NULL	Failed
00YLL22-042-016	00YLL22-042-016	Gemini LpH	MCX	MCX/Gemini LpH	true	NULL	Complete
00YLL22-042-017	00YLL22-042-017	Gemini LpH	MCX	MCX/Gemini LpH	true	NULL	Complete
00YLL22-042-018	00YLL22-042-018	Gemini LpH	MCX	MCX/Gemini LpH	true	NULL	Complete

LCMS Method = **Gemini**

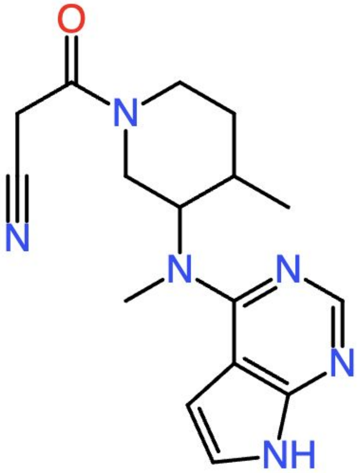
Purification Successful = **true**

SPE Method = **MCX**

Chemical Structure Data

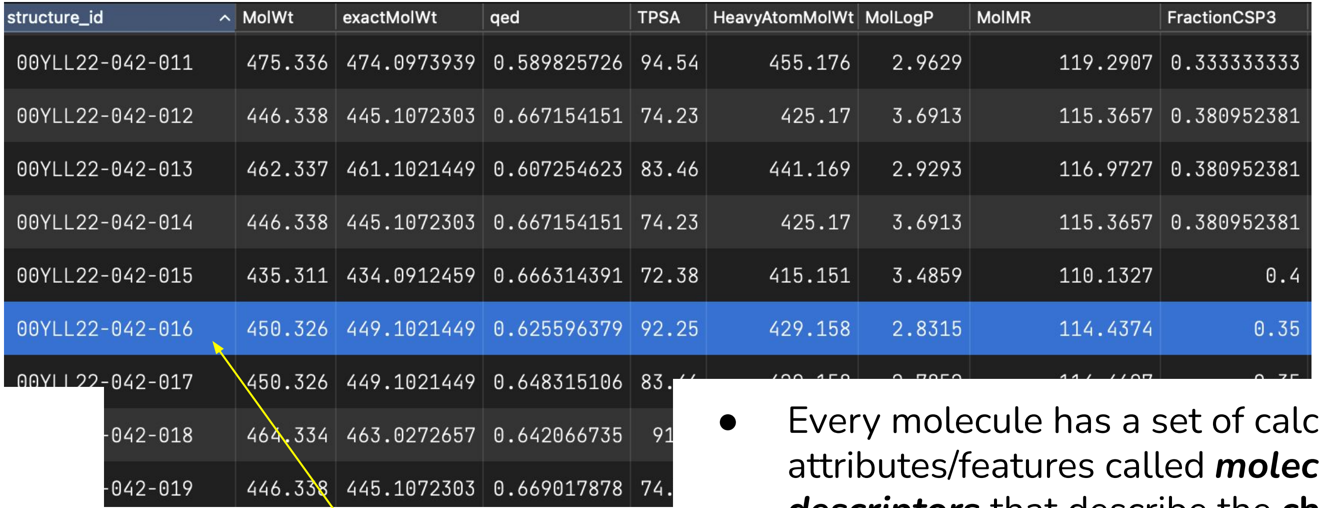
```
CC1CCN(CC1N(C)C2=NC=NC3=C2C=CN3)C(=O)CC#N
```

SMILES representation (text string representation of a 3D molecular structure)

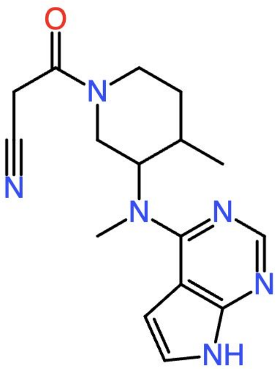


Molecular depiction

Generation of Chemical Structure Data



structure_id	MolWt	exactMolWt	qed	TPSA	HeavyAtomMolWt	MolLogP	MolMR	FractionCSP3
00YLL22-042-011	475.336	474.0973939	0.589825726	94.54	455.176	2.9629	119.2907	0.333333333
00YLL22-042-012	446.338	445.1072303	0.667154151	74.23	425.17	3.6913	115.3657	0.380952381
00YLL22-042-013	462.337	461.1021449	0.607254623	83.46	441.169	2.9293	116.9727	0.380952381
00YLL22-042-014	446.338	445.1072303	0.667154151	74.23	425.17	3.6913	115.3657	0.380952381
00YLL22-042-015	435.311	434.0912459	0.666314391	72.38	415.151	3.4859	110.1327	0.4
00YLL22-042-016	450.326	449.1021449	0.625596379	92.25	429.158	2.8315	114.4374	0.35
00YLL22-042-017	450.326	449.1021449	0.648315106	83.				
-042-018	464.334	463.0272657	0.642066735	91				
-042-019	446.338	445.1072303	0.669017878	74.				

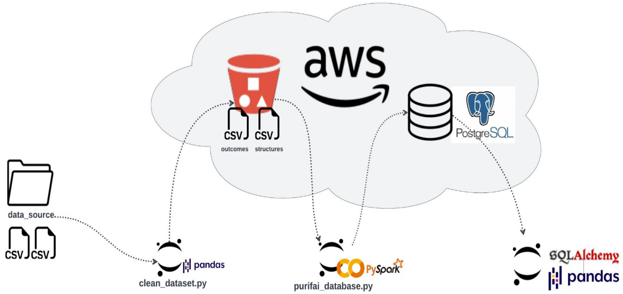


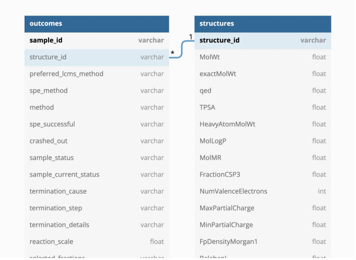
Example molecule
(structure_id = 00YLL-042-016)

- Every molecule has a set of calculated attributes/features called ***molecular descriptors*** that describe the **chemical properties** of a structure

- This was accomplished using the open-source python library RdKit

Data Pipeline, Storage, & Retrieval





outcomes	
sample_id	**varchar**
structure_id	varchar
preferred_lcms_method	varchar
spe_method	varchar
method	varchar
spe_successful	varchar
crashed_out	varchar
sample_status	varchar
sample_current_status	varchar
termination_cause	varchar
termination_step	varchar
termination_details	varchar
reaction_scale	float
selected_fractions	varchar

structures	
structure_id	**varchar**
MolWt	float
exactMolWt	float
qed	float
TPSA	float
HeavyAtomMolWt	float
MolLogP	float
MolMR	float
FractionCSP3	float
NumValenceElectrons	int
MaxPartialCharge	float
MinPartialCharge	float
FpDensityMorgan1	float
BalabanJ	float

Pipeline

1. Data is pulled from company database:
    a. RdKit used to calculate chemical descriptors from SMILES
    b. Pandas used to clean the purification outcomes dataset
2. Cleaned data is stored in AWS S3 buckets
3. Cleaned data is sourced from buckets and written to AWS RDS (postgres) using Pyspark (*see partial schema to the right*)

Storage

- Data is stored in AWS RDS instance

Retrieval

- Data is retrieved from AWS RDS using SQLAlchemy
    - ‘outcomes’ and ‘structures’ are merged into one table, joining on ‘structure_id’
- Data is further processed for ML modeling using Pandas

MACHINE LEARNING

DATA EXPLORATION

ML MODEL

Balanced Random Forest

XGBoost

SMOTE Oversampling

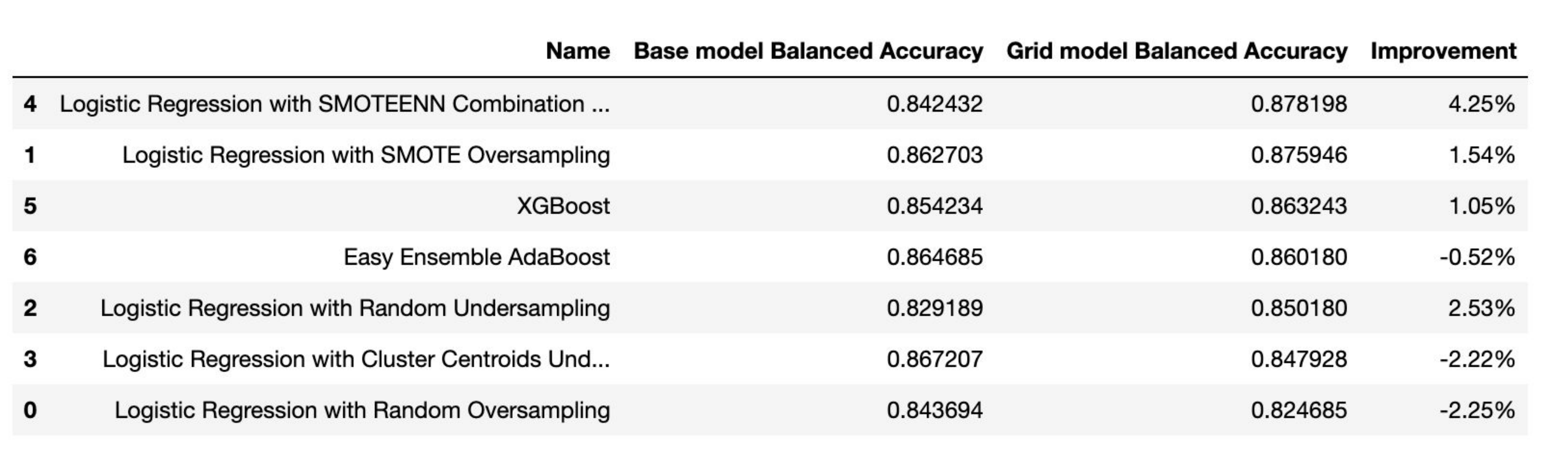
Easy Ensemble AdaBoost

Random Undersampling

Cluster Centroids Undersampling

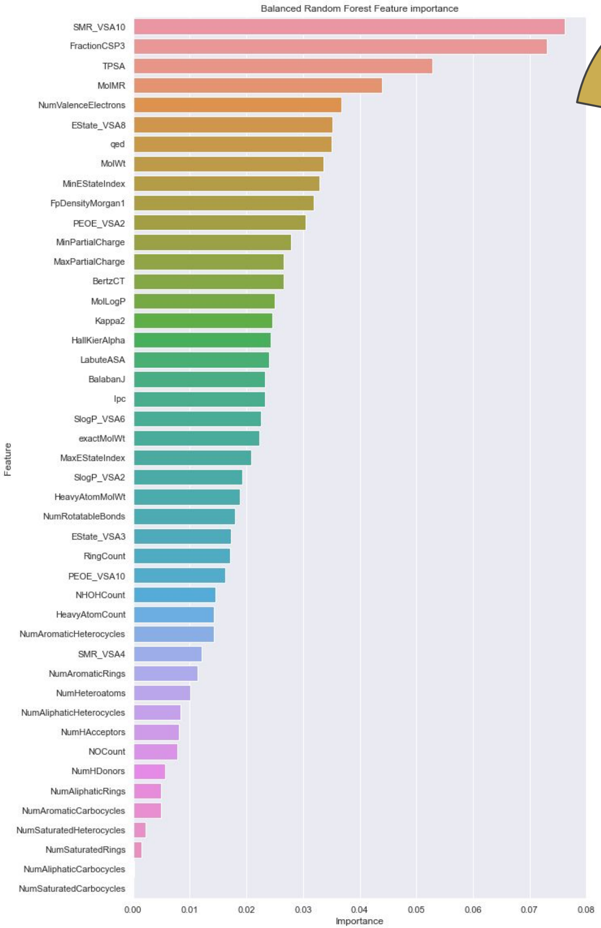
Random Oversampling

SMOTEENN over and undersampling



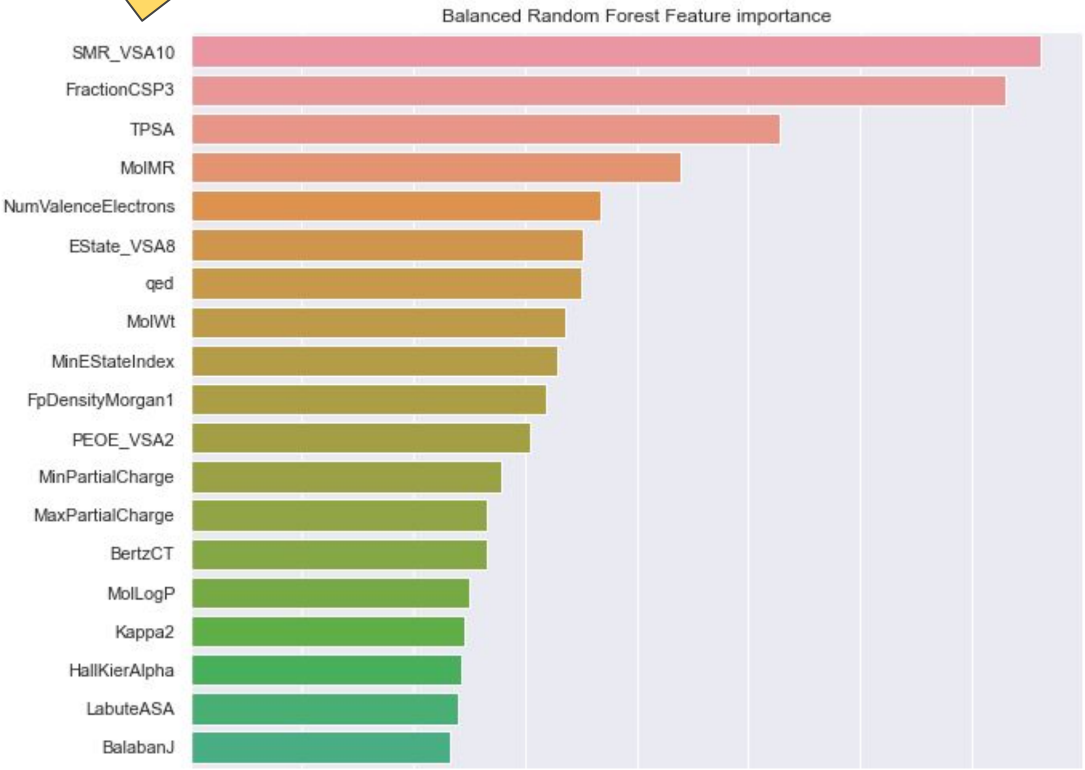
	Name	Base model Balanced Accuracy	Grid model Balanced Accuracy	Improvement
4	Logistic Regression with SMOTEENN Combination ...	0.842432	0.878198	4.25%
1	Logistic Regression with SMOTE Oversampling	0.862703	0.875946	1.54%
5	XGBoost	0.854234	0.863243	1.05%
6	Easy Ensemble AdaBoost	0.864685	0.860180	-0.52%
2	Logistic Regression with Random Undersampling	0.829189	0.850180	2.53%
3	Logistic Regression with Cluster Centroids Und...	0.867207	0.847928	-2.22%
0	Logistic Regression with Random Oversampling	0.843694	0.824685	-2.25%

DATA ANALYSIS



Balanced Random Forest Feature importance

Top 20



Balanced Random Forest Feature importance

Confidence Level

DASHBOARD

SMILE

Questions?