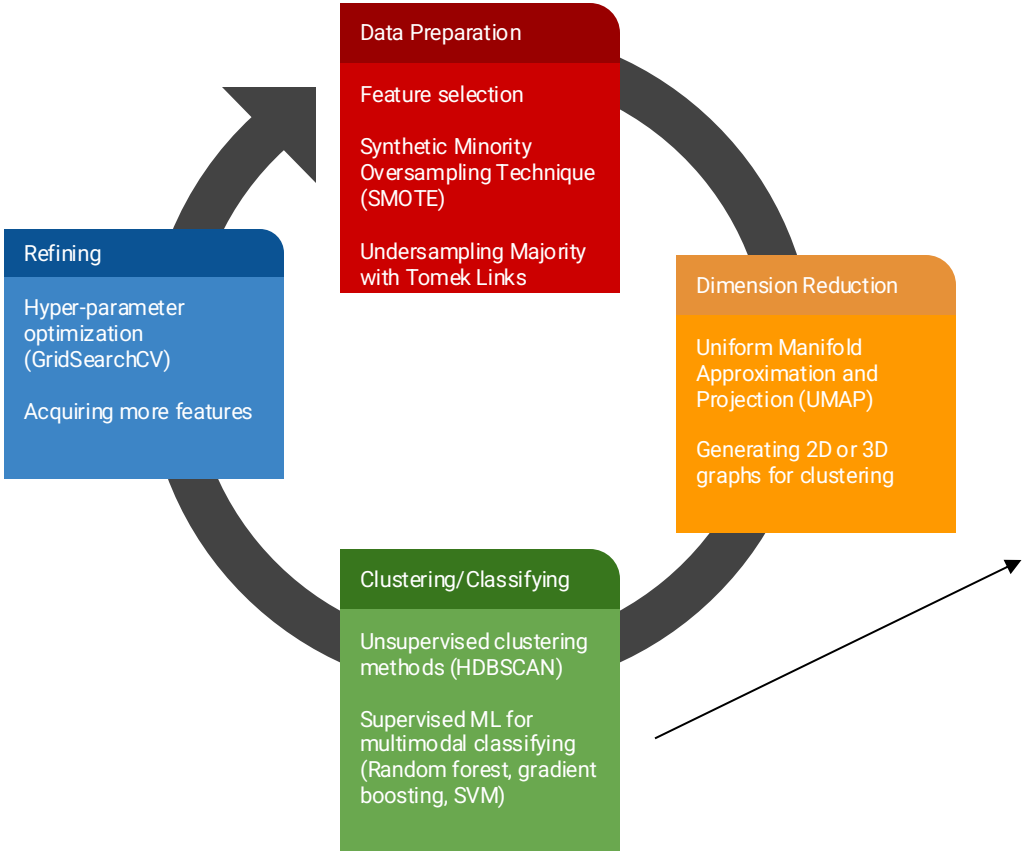
Final Summer Project

Diya

Analysis Process



Algorithms to Compare

Support Vector Machine: best for high-dimensional data, small to medium-sized datasets, using strategies like one-vs-one or one-vs-all, can handle noisy data with the use of the kernel trick and soft margin approach

Gradient Boosting: handles complex relationships between features, supports multiclass classification, can handle noisy data well by focusing on difficult-to-classify instances

Random Forest Classifier: handles datasets with a large number of features, inherently supports multiclass classification, robust to noise and can handle overlapping data well by averaging predictions from multiple trees

Raw Data

20 features presynaptic and postsynaptic

40 features total per synapse

16 normalized features: normalized to the GFP per week

Sample Sizes for Synaptic Analysis

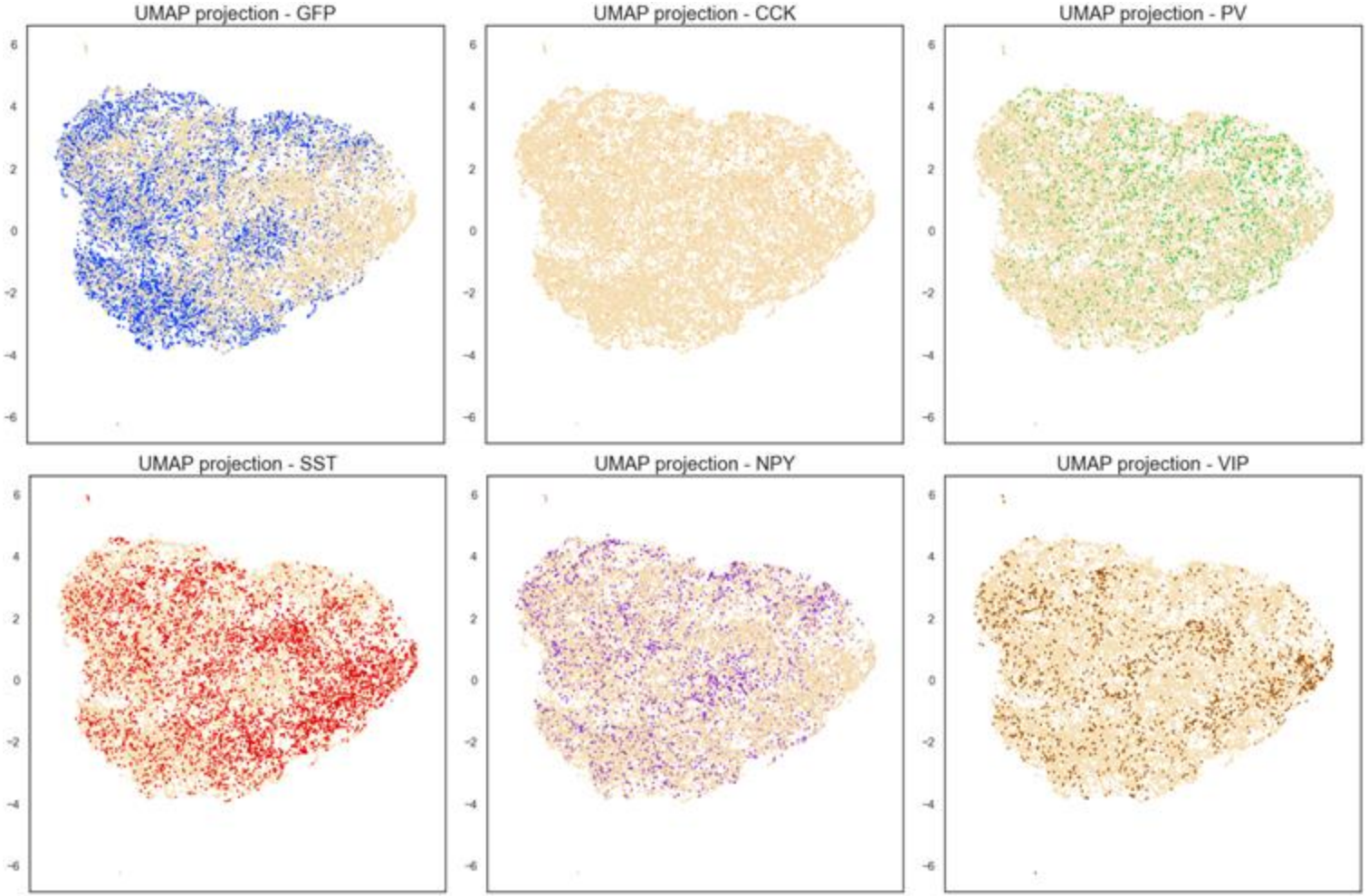
Subtype Marker	Total Number of Cells Analyzed	Total Number of Synapses Analyzed
GFP	30	11519
CCK	14	1112
PV	16	2769
SST	47	6251
NPY	26	4015
VIP	23	1619

Marker	Area_post	Perim__post	Major_post	Minor_post	Circ__post	Feret_post	Skew_post	Kurt_post	MinFeret_po	AR_post	Round_post	Solidity_post	normMeanIn	normRawIntI	normIntDen_	normStddev_	normMode_p	normMin_po	normMax_po	normMedian
GFP	0.192	1.941	0.819	0.298	0.64	0.859	0.708	0.253	0.301	2.743	0.365	0.864	0.76128469	0.70129288	0.70128554	0.55308762	0.74104748	0.82570521	0.75019478	0.76589401
CCK	0.202	1.598	0.57	0.451	0.994	0.644	0.386	-0.865	0.502	1.263	0.791	0.889	0.7409025	0.7184394	0.71843216	0.85293692	0.62919126	0.70107046	0.78037503	0.74780203
PV	0.263	1.882	0.631	0.53	0.932	0.725	0.349	-0.793	0.603	1.19	0.84	0.867	0.55722705	0.70243598	0.70244562	0.52189799	0.51733504	0.56085637	0.56480182	0.58497417
SST	0.222	1.799	0.633	0.447	0.863	0.785	0.662	0.119	0.502	1.416	0.706	0.898	0.7523137	0.80245737	0.80246758	0.54548732	0.74104748	0.79454652	0.74157186	0.75986335
NPY	0.202	1.622	0.587	0.438	0.965	0.644	0.527	-0.548	0.497	1.339	0.747	0.816	0.61358751	0.59498442	0.59498214	0.59077377	0.81095762	0.62317374	0.61222793	0.6271888
VIP	0.212	1.681	0.68	0.397	0.943	0.725	0.638	-0.839	0.449	1.711	0.585	0.857	0.73088234	0.74415919	0.74415209	0.72663313	0.71308343	0.75559816	0.75450625	0.68146475

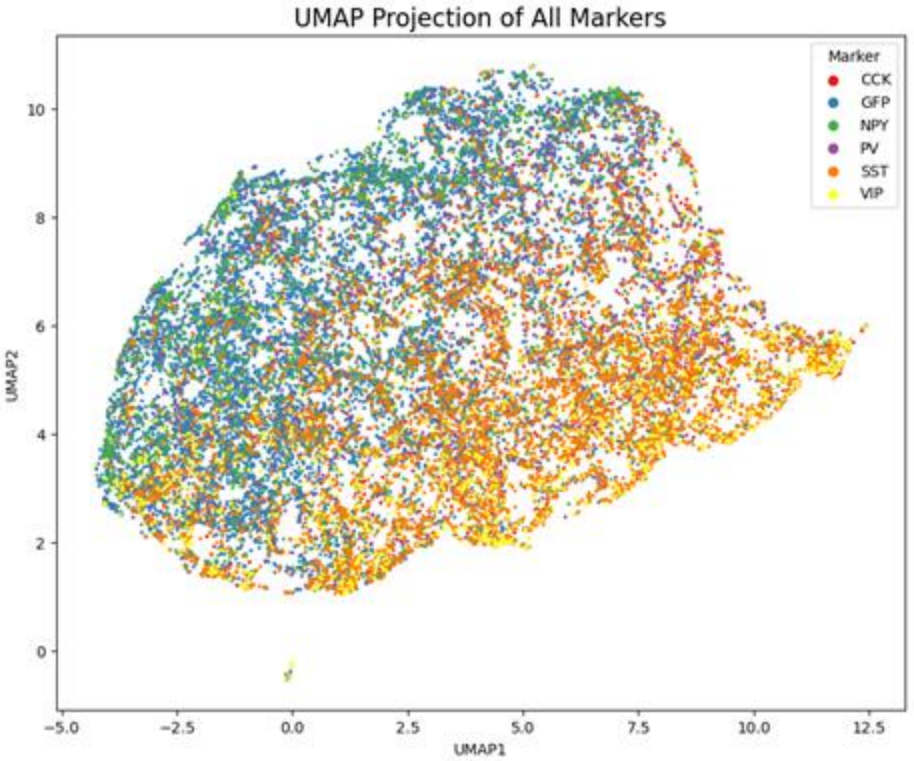
UMAP

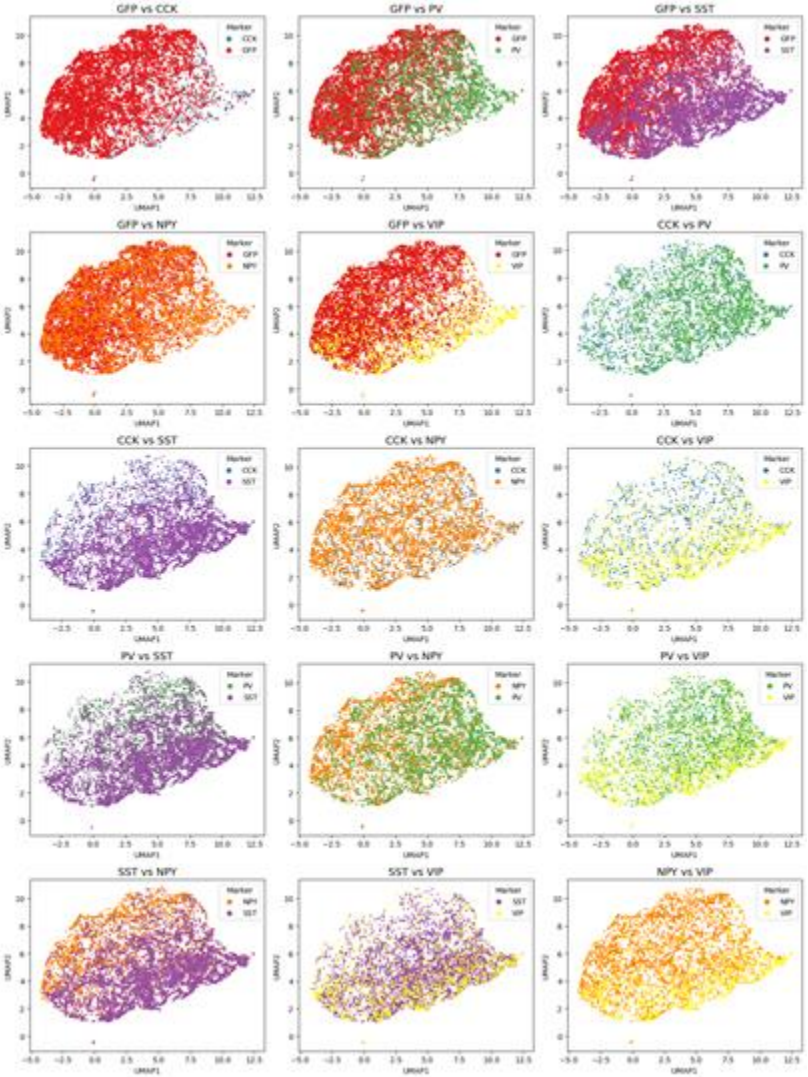
Dimensional reduction of 40 features to 2D

Color coded per subtype



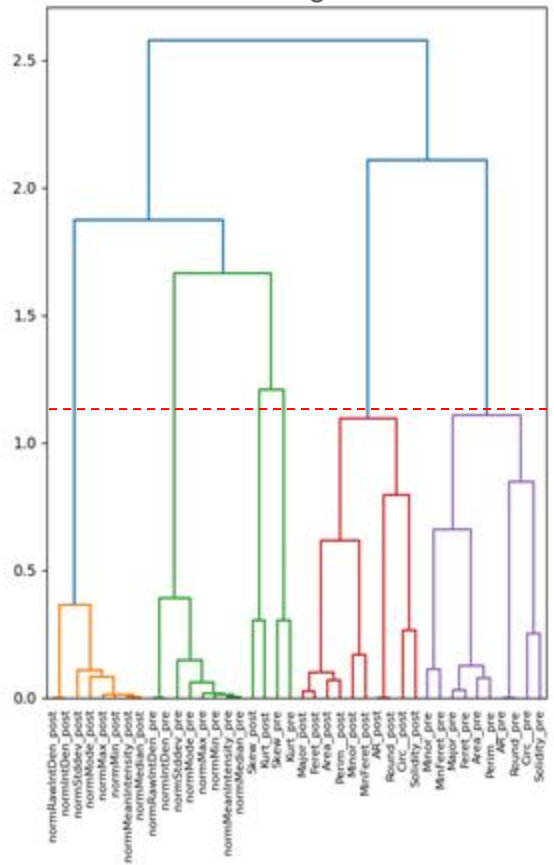
UMAP - pairwise comparisons



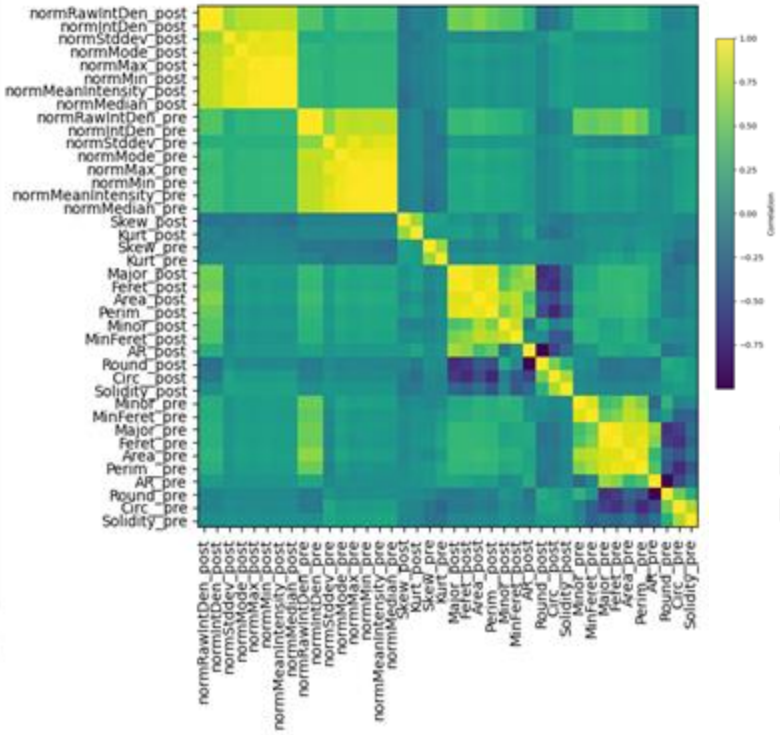


Data Preprocessing - Feature Selection

Dendrogram



Correlation Heatmap



Selects the 1st Feature from Each Cluster:

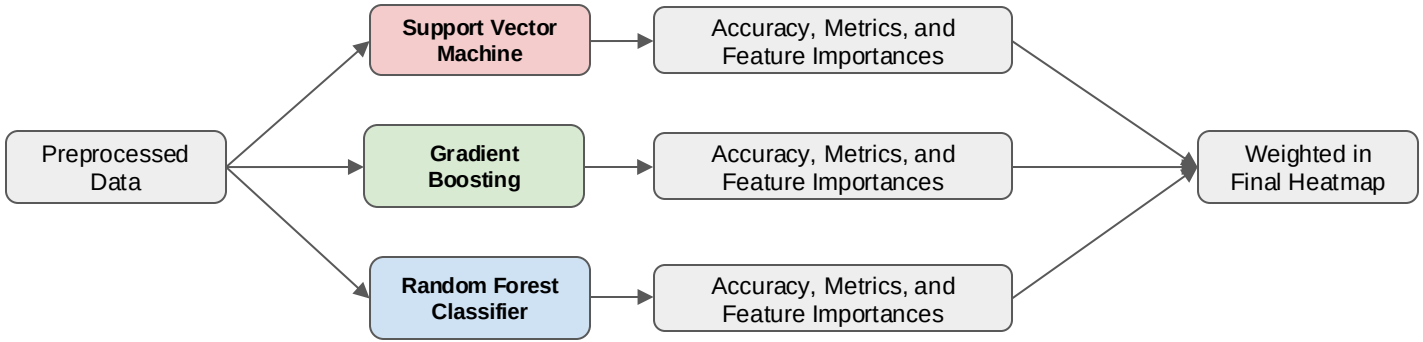
Normalized Mean Intensity pre
Area pre
Skew pre
Normalized Mean Intensity post
Area post
Skew post

Area_post	Skew_post	normMeanIntensity_post	Area_pre	Skew_pre	normMeanIntensity_pre
0.192	0.708	0.761285	0.151	0.560	0.843947
0.202	0.386	0.740903	0.192	0.568	1.290268
0.263	0.349	0.557227	0.222	0.202	1.043221
0.222	0.662	0.752314	0.333	0.126	1.200072
0.202	0.527	0.613588	0.242	0.603	0.709965

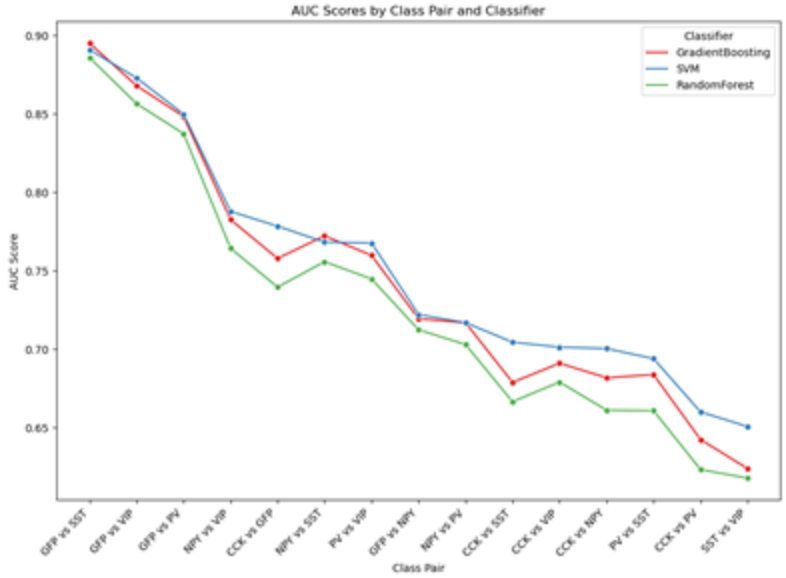
Pairwise Classification for Feature Importance

For each pair:

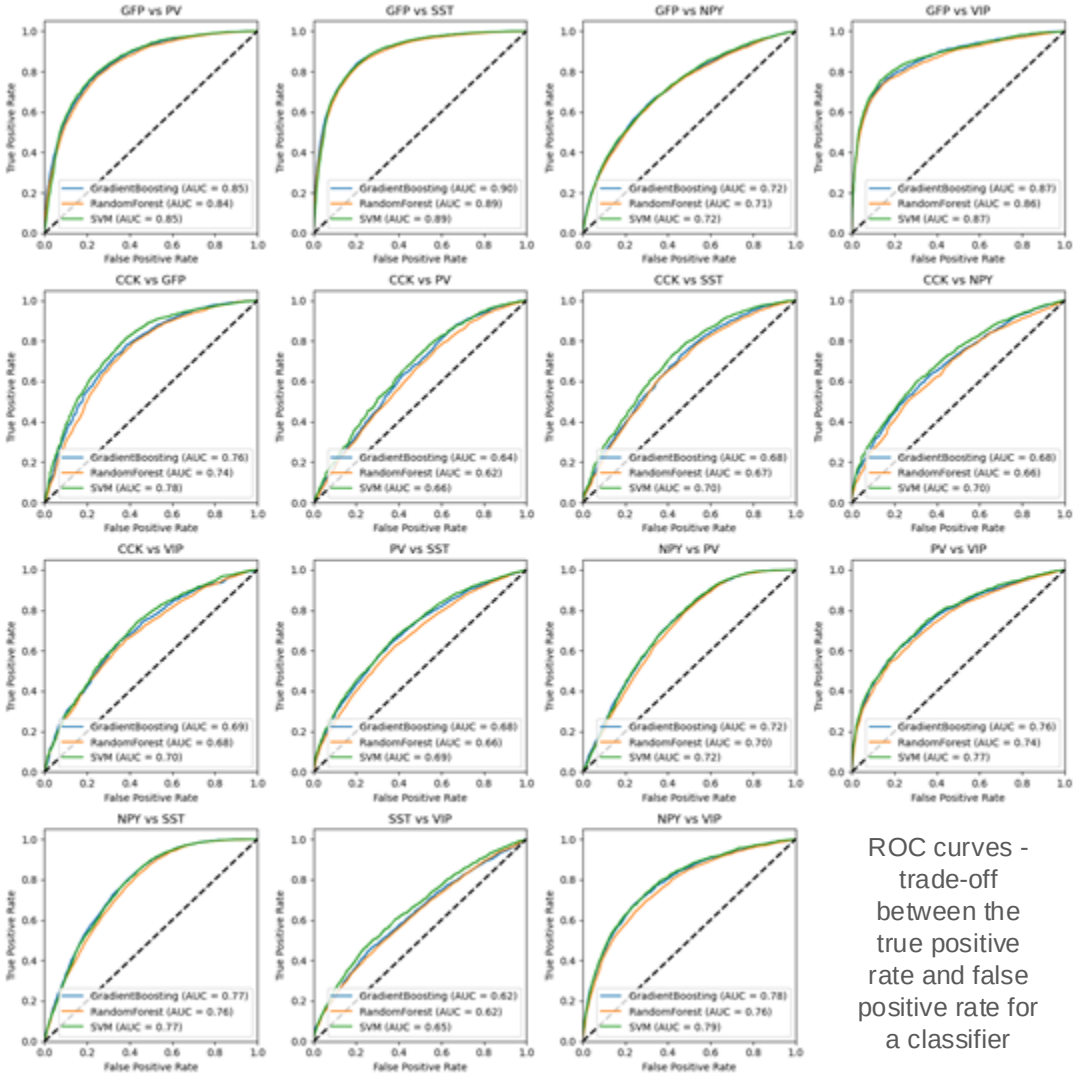
1. Uses Stratified K-Fold cross-validation
2. Runs 3 classifiers: SVM, Gradient Boosting, and Random Forest
3. For each fold in cross-validation (5 folds):
    a. Splits the data into training and testing sets (20% testing, 80% training).
    b. Applies SMOTETomek to balance the training set.
        i. Oversample minority class by using Synthetic Minority Oversampling Technique (SMOTE)
        ii. Undersample majority class by using Tomek Links
    c. Trains the classifier on the resampled training set.
    d. Makes predictions on the test set.
    e. Calculates accuracy and classification report for the predictions.
    f. Calculates permutation feature importances.
4. Averages accuracy, classification report metrics, and feature importances across all folds per



Evaluating Performance



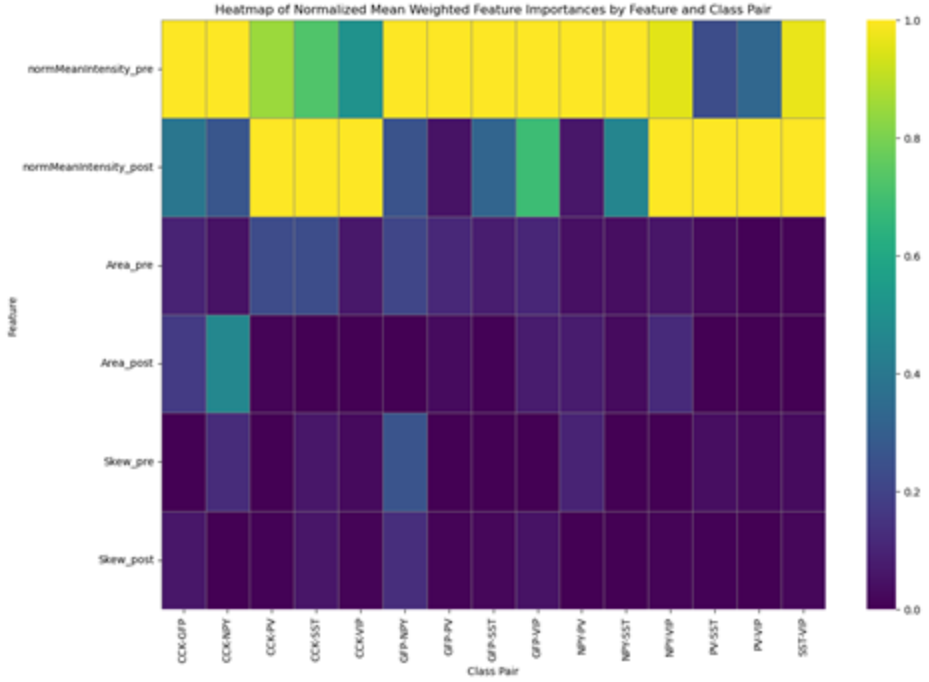
Area under the curve (AUC) - closer to 1 indicates better performance

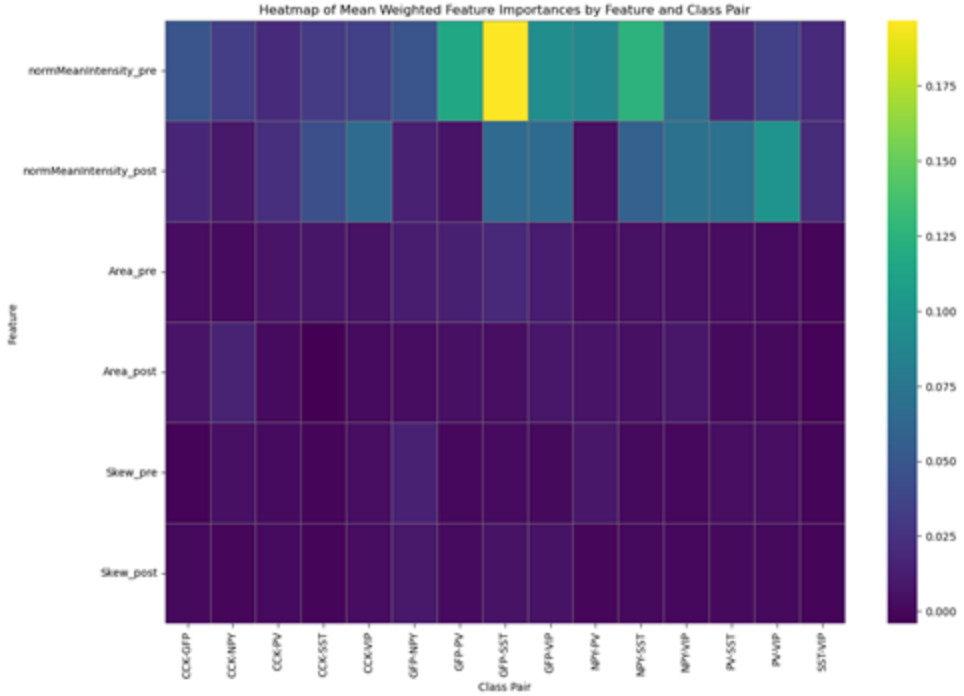


ROC curves - trade-off between the true positive rate and false positive rate for a classifier

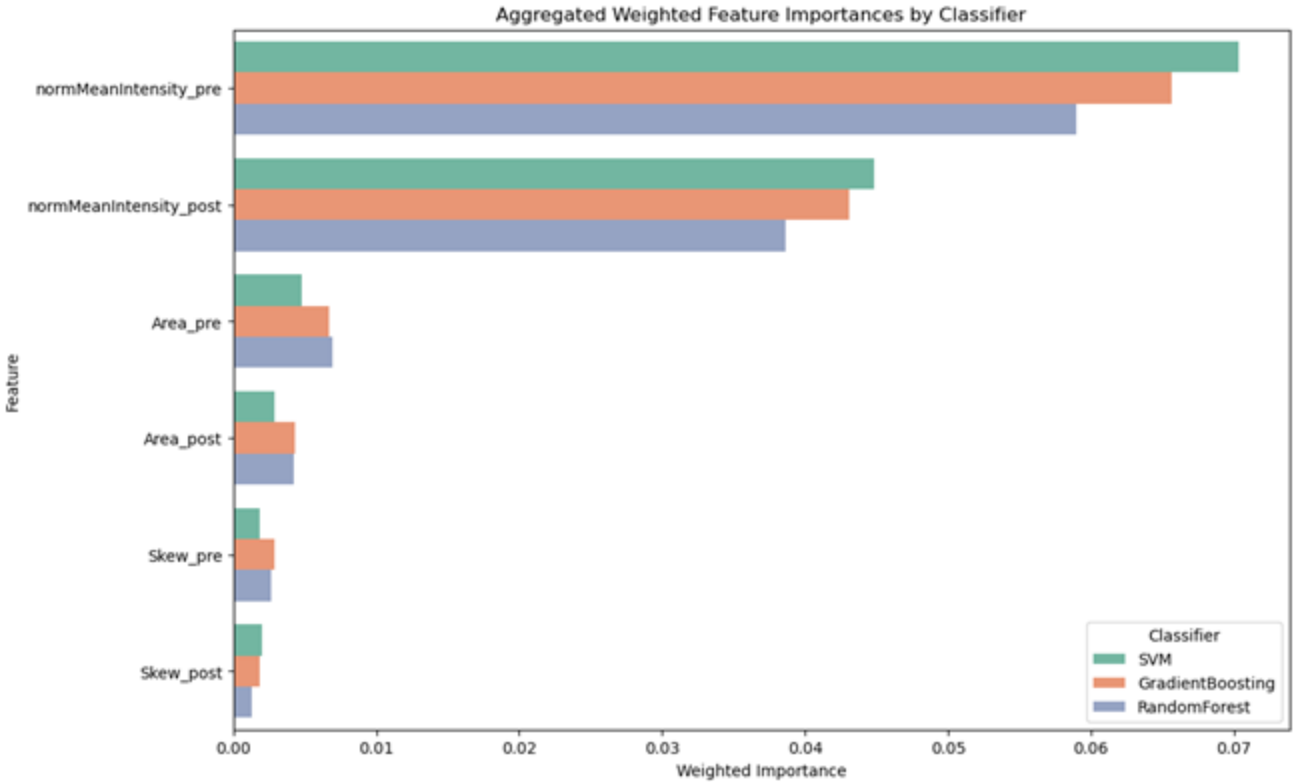
Weighted Feature Importances per Pair

Multiplying feature importances by classifier accuracy for each class pair and taking the mean for each weighted feature across the three classifiers.





Aggregated Weighted Feature Importances



Aggregates the weighted importances by calculating the mean for each feature across all class pairs for each classifier.