**Supplementary Table S1** The Best Combination of hyperparameter for Different Machine Learning Algorithms

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithms** | **Function** | **Parameter Range** | **Optimum Parameter** |
| **AdaBoost** | AdaBoostClassifier | **learning\_rate** = [0.1, 1, 0.01, 0.001], **n\_estimators** = [50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180] | **learning\_rate** = 0.01, **n\_estimators** = 60 |
| **Decision Tree** | DecisionTreeClassifier | **criterion** = ['gini', 'entropy'],**max\_depth** = range (1, 30), **max\_features** = [21, 22, 23, 24, 25, 26, 28, 29, 30, 'auto'] | **criterion** = 'gini', **max\_depth** = 21, **max\_features** = 24 |
| **Quadratic****Discriminant** | QuadraticDiscriminantAnalysis | **reg\_param =** np.linspace(0, 1, 10),**tol =** [0.0001, 0.001, 0.01, 0.1, 1] | **reg\_param =** 1.0,**tol =** 0.0001 |
| **Random Forest** | RandomForestClassifier | **Criterion** = ['gini', 'entropy'], **max\_depth** = range (1, 10),**n\_estimators** = [50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180] | **criterion** = 'entropy', **max\_depth** = 8, **n\_estimators** = 150 |
| **SVM** | SVC | **Cs =** [0.0001, 0.001, 0.01, 0.1, 1, 2, 3, 4, 5, 10],**gamma =** [0.0001, 0.001, 0.01, 0.1, 1],kernel = ['rbf', ' linear '] | **Cs =** 0.01,**gamma =** 0.01,**kernel =** 'linear' |
| **XGBoost** | XGBClassifier | **n\_estimators =** [50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180],**learning\_ rate =** [0.1, 1, 0.01, 0.001] | **Estimators** = 70, **learning\_rate** = 0.01 |