ANNEX III

CLASSIFIERS HYPER PARAMETER TUNING SCORES

1. RANDOM FOREST

TABLE 1: RANDOM FOREST HYPER PARAMETER TUNING RESULTS

Test	Algorithm	Bayes_Best_Score	Accuracy_Score	Bayes_Best_Params
1	RF	0.861060329	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 10), ('max_features', 5), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 200)])
2	RF	0.861060329	0.84	OrderedDict([('criterion', 'entropy'), ('max_depth', 10), ('max_features', 4), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
3	RF	0.868372943	0.81	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
4	RF	0.859232176	0.82	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
5	RF	0.868372943	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 4), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
6	RF	0.859232176	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 10), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
7	RF	0.859232176	0.82	OrderedDict([('criterion', 'gini'), ('max_depth', 10), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 200)])
8	RF	0.864716636	0.82	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
9	RF	0.87202925	0.84	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 7), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
10	RF	0.86654479	0.83	OrderedDict([('criterion', 'gini'), ('max_depth', 10), ('max_features', 4), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
11	RF	0.859232176	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 10), ('max_features', 4), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
12	RF	0.859232176	0.82	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
13	RF	0.859232176	0.82	OrderedDict([('criterion', 'entropy'), ('max_depth', 10), ('max_features', 2), ('min_samples_leaf', 2), ('min_samples_split', 15), ('n_estimators', 100)])
14	RF	0.862888483	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 10), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
15	RF	0.870201097	0.82	OrderedDict([('criterion', 'gini'), ('max_depth', 10), ('max_features', 4), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
16	RF	0.86654479	0.81	OrderedDict([('criterion', 'gini'), ('max_depth', 10), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
17	RF	0.868372943	0.82	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
18	RF	0.864716636	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
19	RF	0.857404022	0.83	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
20	RF	0.862888483	0.82	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
21	RF	0.861060329	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 10), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
22	RF	0.861060329	0.83	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 7), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
23	RF	0.859232176	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 10), ('max_features', 7), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
24	RF	0.864716636	0.84	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 4), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 200)])
25	RF	0.864716636	0.83	OrderedDict([('criterion', 'gini'), ('max_depth', 10), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
26	RF	0.864716636	0.82	OrderedDict([('criterion', 'gini'), ('max_depth', 10), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
27	RF	0.868372943	0.83	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
28	RF	0.851919561	0.83	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 7), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
29	RF	0.864716636	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 4), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
30	RF	0.864716636	0.84	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 4), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
31	RF	0.868372943	0.83	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 5), ('min_samples_split', 10), ('n_estimators', 100)])
32	RF	0.862888483	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 4), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
33	RF	0.859232176	0.82	OrderedDict([('criterion', 'gini'), ('max_depth', 10), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 200)])
34	RF	0.870201097	0.82	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 200)])

35	RF	0.862888483	0.82	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
36	RF	0.862888483	0.82	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
37	RF	0.86654479	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 4), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 200)])
38	RF	0.861060329	0.82	OrderedDict([('criterion', 'entropy'), ('max_depth', 10), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
39	RF	0.851919561	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
40	RF	0.851919561	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
41	RF	0.855575868	0.83	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 4), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 200)])
42	RF	0.862888483	0.83	OrderedDict([('criterion', 'gini'), ('max_depth', 10), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
43	RF	0.851919561	0.82	OrderedDict([('criterion', 'gini'), ('max_depth', 10), ('max_features', 3), ('min_samples_leaf', 5), ('min_samples_split', 10), ('n_estimators', 150)])
44	RF	0.864716636	0.83	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
45	RF	0.86654479	0.84	OrderedDict([('criterion', 'entropy'), ('max_depth', 10), ('max_features', 7), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 200)])
46	RF	0.857404022	0.83	OrderedDict([('criterion', 'gini'), ('max_depth', 10), ('max_features', 4), ('min_samples_leaf', 5), ('min_samples_split', 10), ('n_estimators', 150)])
47	RF	0.862888483	0.84	OrderedDict([('criterion', 'entropy'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
48	RF	0.861060329	0.81	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 4), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])
49	RF	0.859232176	0.82	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 150)])
50	RF	0.862888483	0.83	OrderedDict([('criterion', 'gini'), ('max_depth', 20), ('max_features', 3), ('min_samples_leaf', 2), ('min_samples_split', 10), ('n_estimators', 100)])

2. GAUSSIAN NAIVE BAYES

TABLE 2: GAUSSIAN NAÏVE BAYES HYPER PARAMETER TUNING RESULTS

Test	Algorithm	Bayes_Best_Score	Accuracy_Score	Bayes_Best_Params
1	NB_G	0.722120658	0.72	OrderedDict([('var_smoothing', 1e-05)])
2	NB_G	0.736745887	0.72	OrderedDict([('var_smoothing', 1e-05)])
3	NB_G	0.744058501	0.72	OrderedDict([('var_smoothing', 1e-05)])
4	NB_G	0.729433272	0.72	OrderedDict([('var_smoothing', 1e-05)])
5	NB_G	0.736745887	0.72	OrderedDict([('var_smoothing', 1e-05)])
6	NB_G	0.734917733	0.72	OrderedDict([('var_smoothing', 1e-05)])
7	NB_G	0.73308958	0.72	OrderedDict([('var_smoothing', 1e-05)])
8	NB_G	0.745886654	0.72	OrderedDict([('var_smoothing', 1e-05)])
9	NB_G	0.725776965	0.72	OrderedDict([('var_smoothing', 1e-05)])
10	NB_G	0.740402194	0.72	OrderedDict([('var_smoothing', 1e-05)])
11	NB_G	0.729433272	0.72	OrderedDict([('var_smoothing', 1e-05)])
12	NB_G	0.734917733	0.72	OrderedDict([('var_smoothing', 1e-05)])
13	NB_G	0.736745887	0.72	OrderedDict([('var_smoothing', 1e-05)])
14	NB_G	0.729433272	0.72	OrderedDict([('var_smoothing', 1e-05)])
15	NB_G	0.731261426	0.72	OrderedDict([('var_smoothing', 1e-05)])
16	NB_G	0.73308958	0.72	OrderedDict([('var_smoothing', 1e-05)])
17	NB_G	0.727605119	0.72	OrderedDict([('var_smoothing', 1e-05)])
18	NB_G	0.742230347	0.72	OrderedDict([('var_smoothing', 1e-05)])
19	NB_G	0.734917733	0.72	OrderedDict([('var_smoothing', 1e-05)])
20	NB_G	0.727605119	0.72	OrderedDict([('var_smoothing', 1e-05)])
21	NB_G	0.723948812	0.72	OrderedDict([('var_smoothing', 1e-05)])
22	NB_G	0.727605119	0.72	OrderedDict([('var_smoothing', 1e-05)])
23	NB_G	0.731261426	0.72	OrderedDict([('var_smoothing', 1e-05)])
24	NB_G	0.731261426	0.72	OrderedDict([('var_smoothing', 1e-05)])
25	NB_G	0.740402194	0.72	OrderedDict([('var_smoothing', 1e-05)])
26	NB_G	0.742230347	0.72	OrderedDict([('var_smoothing', 1e-05)])
27	NB_G	0.73857404	0.72	OrderedDict([('var_smoothing', 1e-05)])
28	NB_G	0.734917733	0.72	OrderedDict([('var_smoothing', 1e-05)])
29	NB_G	0.725776965	0.72	OrderedDict([('var_smoothing', 1e-05)])
30	NB_G	0.742230347	0.72	OrderedDict([('var_smoothing', 1e-05)])
31	NB_G	0.734917733	0.72	OrderedDict([('var_smoothing', 1e-05)])
32	NB_G	0.736745887	0.72	OrderedDict([('var_smoothing', 1e-05)])
33	NB_G	0.731261426	0.72	OrderedDict([('var_smoothing', 1e-05)])
34	NB_G	0.73857404	0.72	OrderedDict([('var_smoothing', 1e-05)])
35	NB_G	0.73857404	0.72	OrderedDict([('var_smoothing', 1e-05)])
36	NB_G	0.73308958	0.72	OrderedDict([('var_smoothing', 1e-05)])
37	NB_G	0.734917733	0.72	OrderedDict([('var_smoothing', 1e-05)])
38	NB_G	0.742230347	0.72	OrderedDict([('var_smoothing', 1e-05)])
39	NB_G	0.734917733	0.72	OrderedDict([('var_smoothing', 1e-05)])
40	NB_G	0.73308958	0.72	OrderedDict([('var_smoothing', 1e-05)])
41	NB_G	0.736745887	0.72	OrderedDict([('var_smoothing', 1e-05)])
42	NB_G	0.744058501	0.72	OrderedDict([('var_smoothing', 1e-05)])
43	NB_G	0.73308958	0.72	OrderedDict([('var_smoothing', 1e-05)])
44	NB_G	0.740402194	0.72	OrderedDict([('var_smoothing', 1e-05)])
45	NB_G	0.731261426	0.72	OrderedDict([('var_smoothing', 1e-05)])
46	NB_G	0.727605119	0.72	OrderedDict([('var_smoothing', 1e-05)])
47	NB_G	0.740402194	0.72	OrderedDict([('var_smoothing', 1e-05)])
48	NB_G	0.736745887	0.72	OrderedDict([('var_smoothing', 1e-05)])
49	NB_G	0.731261426	0.72	OrderedDict([('var_smoothing', 1e-05)])
50	NB_G	0.73857404	0.72	OrderedDict([('var_smoothing', 1e-05)])

3. SUPPORT VECTOR MACHINES

TABLE 3: SUPPORT VECTOR MACHINES HYPER PARAMETER TUNING RESULTS

Test	Algorithm	Bayes_Best_Score	Accuracy_Score	Bayes_Best_Params
1	SVC	0.787934186	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 3.501), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', False), ('tol', 0.001)])
2	SVC	0.791590494	0.74	OrderedDict([('C', 51), ('cache_size', 400), ('coef0', 1.001), ('degree', 3), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
3	SVC	0.800731261	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 2.001), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', False), ('tol', 0.001)])
4	SVC	0.800731261	0.73	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 9.501), ('degree', 3), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
5	SVC	0.798903108	0.73	OrderedDict([('C', 41), ('cache_size', 200), ('coef0', 1.501), ('degree', 6), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
6	SVC	0.791590494	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 2.501), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.001)])
7	SVC	0.786106033	0.73	OrderedDict([('C', 41), ('cache_size', 200), ('coef0', 9.001), ('degree', 4), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
8	SVC	0.784277879	0.73	OrderedDict([('C', 31), ('cache_size', 300), ('coef0', 2.501), ('degree', 3), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', False), ('tol', 0.001)])
9	SVC	0.798903108	0.74	OrderedDict([('C', 51), ('cache_size', 400), ('coef0', 8.001), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
10	SVC	0.736745887	0.76	OrderedDict([('C', 11), ('cache_size', 400), ('coef0', 2.501), ('degree', 4), ('gamma', 'auto'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
11	SVC	0.800731261	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 6.501), ('degree', 6), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
12	SVC	0.786106033	0.74	OrderedDict([('C', 51), ('cache_size', 400), ('coef0', 4.001), ('degree', 6), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
13	SVC	0.802559415	0.74	OrderedDict([('C', 51), ('cache_size', 300), ('coef0', 1.001), ('degree', 5), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
14	SVC	0.780621572	0.74	OrderedDict([('C', 51), ('cache_size', 300), ('coef0', 5.001), ('degree', 6), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
15	SVC	0.793418647	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 2.001), ('degree', 5), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.001)])
16	SVC	0.793418647	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 7.501), ('degree', 3), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', False), ('tol', 1e-05)])
17	SVC	0.786106033	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 6.501), ('degree', 5), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.001)])
18	SVC	0.791590494	0.74	OrderedDict([('C', 51), ('cache_size', 300), ('coef0', 9.001), ('degree', 5), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
19	SVC	0.782449726	0.74	OrderedDict([('C', 51), ('cache_size', 300), ('coef0', 5.001), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.001)])
20	SVC	0.791590494	0.74	OrderedDict([('C', 51), ('cache_size', 300), ('coef0', 2.501), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.001)])
21	SVC	0.787934186	0.73	OrderedDict([('C', 51), ('cache_size', 400), ('coef0', 5.501), ('degree', 5), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
22	SVC	0.793418647	0.74	OrderedDict([('C', 51), ('cache_size', 400), ('coef0', 6.001), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
23	SVC	0.797074954	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 1.001), ('degree', 5), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
24	SVC	0.798903108	0.73	OrderedDict([('C', 41), ('cache_size', 400), ('coef0', 5.001), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.001)])
25	SVC	0.78976234	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 4.501), ('degree', 4), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
26	SVC	0.787934186	0.74	OrderedDict([('C', 51), ('cache_size', 300), ('coef0', 7.501), ('degree', 5), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
27	SVC	0.793418647	0.74	OrderedDict([('C', 51), ('cache_size', 400), ('coef0', 8.001), ('degree', 4), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', False), ('tol', 0.001)])
28	SVC	0.775137112	0.73	OrderedDict([('C', 41), ('cache_size', 300), ('coef0', 1.501), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.001)])
29	SVC	0.798903108	0.73	OrderedDict([('C', 41), ('cache_size', 200), ('coef0', 5.501), ('degree', 3), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.001)])
30	SVC	0.776965265	0.74	OrderedDict([('C', 51), ('cache_size', 400), ('coef0', 0.501), ('degree', 4), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.001)])
31	SVC	0.798903108	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 3.501), ('degree', 6), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
32	SVC	0.791590494	0.74	OrderedDict([('C', 51), ('cache_size', 400), ('coef0', 7.001), ('degree', 3), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', False), ('tol', 1e-05)])
33	SVC	0.786106033	0.73	OrderedDict([('C', 41), ('cache_size', 400), ('coef0', 8.501), ('degree', 6), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
34	SVC	0.802559415	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 9.001), ('degree', 6), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])

35	SVC	0.797074954	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 9.001), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
36	SVC	0.78976234	0.74	OrderedDict([('C', 51), ('cache_size', 300), ('coef0', 2.001), ('degree', 6), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
37	SVC	0.798903108	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 3.501), ('degree', 3), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
38	SVC	0.78976234	0.74	OrderedDict([('C', 51), ('cache_size', 400), ('coef0', 7.501), ('degree', 6), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
39	SVC	0.784277879	0.74	OrderedDict([('C', 51), ('cache_size', 400), ('coef0', 2.001), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
40	SVC	0.782449726	0.73	OrderedDict([('C', 21), ('cache_size', 400), ('coef0', 3.001), ('degree', 6), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', False), ('tol', 1e-05)])
41	SVC	0.793418647	0.74	OrderedDict([('C', 51), ('cache_size', 300), ('coef0', 4.501), ('degree', 6), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
42	SVC	0.787934186	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 1.001), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
43	SVC	0.793418647	0.74	OrderedDict([('C', 51), ('cache_size', 400), ('coef0', 7.501), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', False), ('tol', 1e-05)])
44	SVC	0.793418647	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 3.001), ('degree', 4), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
45	SVC	0.806215722	0.74	OrderedDict([('C', 51), ('cache_size', 300), ('coef0', 4.001), ('degree', 7), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.001)])
46	SVC	0.780621572	0.74	OrderedDict([('C', 51), ('cache_size', 400), ('coef0', 7.001), ('degree', 4), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
47	SVC	0.78976234	0.73	OrderedDict([('C', 41), ('cache_size', 200), ('coef0', 6.001), ('degree', 5), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.1)])
48	SVC	0.802559415	0.74	OrderedDict([('C', 51), ('cache_size', 300), ('coef0', 6.001), ('degree', 4), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 1e-05)])
49	SVC	0.787934186	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 9.001), ('degree', 6), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', False), ('tol', 0.1)])
50	SVC	0.798903108	0.74	OrderedDict([('C', 51), ('cache_size', 200), ('coef0', 8.001), ('degree', 5), ('gamma', 'scale'), ('kernel', 'rbf'), ('probability', True), ('tol', 0.001)])

4. K NEAREST NEIGHBORS

TABLE 4: K NEAREST NEIGHBORS HYPER PARAMETER TUNING RESULTS

Test	Algorithm	Bayes_Best_Score	Accuracy_Score	Bayes_Best_Params
1	KNB	0.806215722	0.8	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
2	KNB	0.81535649	0.8	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 2), ('weights', 'distance')])
3	KNB	0.81535649	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 20), ('metric', 'minkowski'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
4	KNB	0.809872029	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
5	KNB	0.81535649	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 3), ('weights', 'distance')])
6	KNB	0.813528336	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
7	KNB	0.817184644	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 4), ('weights', 'distance')])
8	KNB	0.811700183	0.8	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 3), ('weights', 'distance')])
9	KNB	0.811700183	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 2), ('weights', 'distance')])
10	KNB	0.81535649	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
11	KNB	0.817184644	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
12	KNB	0.819012797	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
13	KNB	0.81535649	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 3), ('weights', 'distance')])
14	KNB	0.819012797	0.8	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 4), ('weights', 'distance')])
15	KNB	0.809872029	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 4), ('weights', 'distance')])
16	KNB	0.820840951	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 4), ('weights', 'distance')])
17	KNB	0.809872029	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 4), ('weights', 'distance')])
18	KNB	0.809872029	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 2), ('weights', 'distance')])
19	KNB	0.806215722	0.81	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 7), ('p', 2), ('weights', 'distance')])
20	KNB	0.81535649	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 2), ('weights', 'distance')])
21	KNB	0.804387569	0.81	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 7), ('p', 2), ('weights', 'distance')])
22	KNB	0.808043876	0.8	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 2), ('weights', 'distance')])
23	KNB	0.81535649	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 3), ('weights', 'distance')])
24	KNB	0.820840951	0.8	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 2), ('weights', 'distance')])
25	KNB	0.802559415	0.8	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 20), ('metric', 'minkowski'), ('n_neighbors', 7), ('p', 1), ('weights', 'uniform')])
26	KNB	0.806215722	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
27	KNB	0.822669104	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 3), ('weights', 'distance')])
28	KNB	0.802559415	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
29	KNB	0.791590494	0.76	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 2), ('p', 4), ('weights', 'uniform')])
30	KNB	0.798903108	0.8	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 7), ('p', 2), ('weights', 'uniform')])
31	KNB	0.819012797	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
32	KNB	0.811700183	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
33	KNB	0.811700183	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 4), ('weights', 'distance')])
34	KNB	0.811700183	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 2), ('weights', 'distance')])

35	KNB	0.817184644	0.8	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 40), ('metric', 'minkowski'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
36	KNB	0.798903108	0.81	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 7), ('p', 3), ('weights', 'distance')])
37	KNB	0.813528336	0.8	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 4), ('weights', 'distance')])
38	KNB	0.81535649	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 20), ('metric', 'minkowski'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
39	KNB	0.802559415	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
40	KNB	0.811700183	0.8	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 30), ('metric', 'minkowski'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
41	KNB	0.813528336	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 4), ('weights', 'distance')])
42	KNB	0.806215722	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 4), ('weights', 'distance')])
43	KNB	0.81535649	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
44	KNB	0.806215722	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 3), ('weights', 'distance')])
45	KNB	0.813528336	0.8	OrderedDict([('algorithm', 'kd_tree'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
46	KNB	0.798903108	0.8	OrderedDict([('algorithm', 'brute'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
47	KNB	0.817184644	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 4), ('weights', 'distance')])
48	KNB	0.811700183	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 20), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 1), ('weights', 'distance')])
49	KNB	0.804387569	0.81	OrderedDict([('algorithm', 'brute'), ('leaf_size', 40), ('metric', 'manhattan'), ('n_neighbors', 7), ('p', 4), ('weights', 'distance')])
50	KNB	0.811700183	0.8	OrderedDict([('algorithm', 'ball_tree'), ('leaf_size', 30), ('metric', 'manhattan'), ('n_neighbors', 10), ('p', 3), ('weights', 'distance')])