Classifier	KNeighborsClassifier			
Parameters	n_neighbors=1			
	result1	result2	result3	mean result
fit_time	10.11867356	10.63974762	9.55846453	10.10562857
score_time	390.4468646	391.8257494	457.1624386	413.1450175
test_f1	0.24	0.0952381	0.3	0.21174603
test_precision	0.42857143	0.25	1	0.55952381
test_recall	0.16666667	0.05882353	0.17647059	0.13398693

Classifier	KNeighborsClassifier			
Parameters	n_neighbors=3			
	result1	result2	result3	mean result
fit_time	10.88711643	11.21104455	10.48717833	10.86177977
score_time	476.5946572	447.5135305	458.3112814	460.8064897
test_f1	0	0	0	0
test_precision	0	0	0	0
test_recall	0	0	0	0

Classifier	KNeighborsClassifier			
Parameters	n_neighbors=5			
	result1	result2	result3	mean result
fit_time	12.9892602	11.68674159	8.6438818	11.10662786
score_time	400.2301948	416.3879404	391.7707279	402.7962877
test_f1	0	0	0	0
test_precision	0	0	0	0
test_recall	0	0	0	0

SVC classifier parameters that returned a mean zero f1

C	kernel	degree	gamma
1	poly	3	N/A
1.0	rbf	N/A	1/(X.shape[1]*X.std())
1.0	N/A	N/A	1/(X.shape[1]*X.std())
9.0	sigmoid	N/A	auto
1.0	sigmoid	N/A	auto
24.0	sigmoid	N/A	1/(X.shape[1]*X.std())
2.0	poly	2	auto_deprecated
2.0	rbf	N/A	auto
0.001	rbf	N/A	auto
0.001	sigmoid	N/A	auto
0.001	linear	N/A	auto

There are other combiations of parameters too for which I get 0 mean f1 score

Best (in terms of mean F1) SVC result I got

Parameters	C=1.0	kernel=linear		
	result1	result2	result3	mean result
fit_time	20.27377844	18.7149477	20.57497048	19.85456554
score_time	20.9429853	20.2867372	20.48222017	20.57064756
test_f1	0.48	0.5	0.58064516	0.520215053
test_precision	0.85714286	0.85714286	0.64285714	0.785714287

test_recall	0.33333333	0.35294118	0.52941176	0.405228757

Following are few of the parameter combinations for which I get mean f1 score that is pretty close or same to the best mean f1 score for SVC that I got

SVC result I got

Parameters	C=6.0	kernel=linear		
	result1	result2	result3	mean result
fit_time	20.2338798	18.76369762	19.33129835	19.44295859
score_time	20.16905379	20.935009	21.49760485	20.86722255
test_f1	0.48	0.5	0.58064516	0.520215053
test_precision	0.85714286	0.85714286	0.64285714	0.785714287
test_recall	0.33333333	0.35294118	0.52941176	0.405228757

SVC result I got

Parameters	C=24.0	kernel= linear		
	result1	result2	result3	mean result
fit_time	22.88679171	18.8987999	21.19630647	20.99396603
score_time	22.49932933	20.07730246	21.42170715	21.33277965
test_f1	0.48	0.5	0.58064516	0.520215053
test_precision	0.85714286	0.85714286	0.64285714	0.785714287
test_recall	0.33333333	0.35294118	0.52941176	0.405228757

SVC result I got

Parameters	C=80.0	kernel= linear		
	result1	result2	result3	mean result
fit_time	22.0729692	19.22258973	20.06168294	20.45241396
score_time	20.69364834	25.0480144	22.54669762	22.76278679
test_f1	0.48	0.5	0.58064516	0.520215053
test_precision	0.85714286	0.85714286	0.64285714	0.785714287
test_recall	0.33333333	0.35294118	0.52941176	0.405228757

SVC result I got

Parameters	C=1000000.0	kernel= linear		
	result1	result2	result3	mean result
fit_time	20.90291619	19.34227395	20.99981499	20.41500171
score_time	21.14402151	20.76527071	21.80366182	21.23765135
test_f1	0.48	0.5	0.58064516	0.520215053
test_precision	0.85714286	0.85714286	0.64285714	0.785714287

test_recall	0.33333333	0.35294118	0.52941176	0.405228757

SVC result I got

Parameters	C=4.0	kernel= poly	coef0= 5.0	degree = 10	gamma=1/(X.shape[1]*X.std())
	result1	result2	result3	mean result	
fit_time	28.3982749	32.2774148	28.69279456	29.78949475	
score_time	36.97526264	35.98357391	33.03843904	35.3324252	
test_f1	0.61538462	0.38095238	0.55172414	0.51602038	
test_precision	1.0	1.0	0.66666667	0.88888889	
test_recall	0.44444444	0.23529412	0.47058824	0.383442267	