dataset	tipo	tecnica	densidad	riesgo	pureza	n_train	n_test	n_features	es_grande	cv_splits	n_iter	modelo	mejor_configuracion	cv_f1_macro	cv_balanced_accuracy	cv_mcc	cv_cohen_kappa	test_f1_macro	test_balanced_accuracy	test_mcc	test_cohen_kappa	search_time_sec	n_jobs_search
glass	aumentado	pcsmote	50	25	entropia	268	43	9	FALSO	5	10	LogisticRegression	{'classifier__C': np.float64(9.877700294007917), 'classifier__class_weight': 'balanced', 'classifier__dual': False, 'classifier__fit_intercept': False, 'classifier__penalty': 'l2', 'classifier__solver': 'saga', 'classifier__tol': np.float64(2.3270677083837777e-05)}	0,73034431	0,77019425	0,6740974	0,664656793	0,596264846	0,746031746	0,54163513	0,515573227	0,493	4
glass	aumentado	pcsmote	50	75	entropia	268	43	9	FALSO	5	10	LogisticRegression	{'classifier__C': np.float64(9.877700294007917), 'classifier__class_weight': 'balanced', 'classifier__dual': False, 'classifier__fit_intercept': False, 'classifier__penalty': 'l2', 'classifier__solver': 'saga', 'classifier__tol': np.float64(2.3270677083837777e-05)}	0,70374612	0,741736597	0,63441091	0,627837193	0,596264846	0,746031746	0,54163513	0,515573227	0,491	4
glass	base	base				149	65	9	FALSO	5	10	LogisticRegression	{'classifier__C': np.float64(0.4042872735027334), 'classifier__class_weight': 'balanced', 'classifier__dual': False, 'classifier__fit_intercept': True, 'classifier__penalty': 'l2', 'classifier__solver': 'liblinear', 'classifier__tol': np.float64(0.00087060208783048 54)}	0,54994696	0,603114478	0,46841298	0,445878493	0,603764081	0,694271912	0,54605056	0,512956603	0,641	4

dataset	tipo	tecnica	densidad	riesgo	pureza	n_train	n_test	n_features	es_grande	cv_splits	n_iter	modelo	mejor_configuracion	cv_f1_macro	cv_balanced_accuracy	cv_mcc	cv_cohen_kappa	test_f1_macro	test_balanced_accuracy	test_mcc	test_cohen_kappa	search_time_sec	n_jobs_search
wdbc	aumentado	pcsmote	25	25	entropia	513	114	30	FALSO	5	10	LogisticRegression	{'classifier__C': np.float64(0.4689400963537689), 'classifier__class_weight': 'balanced', 'classifier__dual': False, 'classifier__fit_intercept': False, 'classifier__penalty': 'l2', 'classifier__solver': 'saga', 'classifier__tol': np.float64(5.4041038546473305e-05)}	0,97206785	0,970114416	0,9458009	0,944222747	0,980955563	0,976190476	0,9626219	0,961923848	4,86	4
wdbc	aumentado	pcsmote	75	25	entropia	513	114	30	FALSO	5	10	LogisticRegression	{'classifier__C': np.float64(0.4689400963537689), 'classifier__class_weight': 'balanced', 'classifier__dual': False, 'classifier__fit_intercept': False, 'classifier__penalty': 'l2', 'classifier__solver': 'saga', 'classifier__tol': np.float64(5.4041038546473305e-05)}	0,97206785	0,970114416	0,9458009	0,944222747	0,980955563	0,976190476	0,9626219	0,961923848	1,98	4
wdbc	base	base				398	171	30	FALSO	5	10	LogisticRegression	{'classifier__C': np.float64(0.4042872735027334), 'classifier__class_weight': 'balanced', 'classifier__dual': False, 'classifier__fit_intercept': True, 'classifier__penalty': 'l2', 'classifier__solver': 'liblinear', 'classifier__tol': np.float64(0.00087060208783048 54)}	0,96791257	0,967655172	0,93661814	0,935868348	0,987514603	0,987514603	0,97502921	0,975029206	5,386	4

dataset	tipo	tecnica	densidad	riesgo	pureza	n_train	n_test	n_features	es_grande	cv_splits	n_iter	modelo	mejor_configuracion	cv_f1_macro	cv_balanced_accuracy	cv_mcc	cv_cohen_kappa	test_f1_macro	test_balanced_accuracy	test_mcc	test_cohen_kappa	search_time_sec	n_jobs_search
heart	aumentado	pcsmote	75	25	proporcion	444	60	13	FALSO	5	10	LogisticRegression	{'classifier__C': np.float64(0.35849855803404745), 'classifier__class_weight': None, 'classifier__dual': False, 'classifier__fit_intercept': True, 'classifier__penalty': 'l1', 'classifier__solver': 'liblinear', 'classifier__tol': np.float64(2.1930485556643678e-05)}	0,55160728	0,559953114	0,59573829	0,592746852	0,204761905	0,216071429	0,21416135	0,209809264	2,454	4
heart	aumentado	pcsmote	75	50	proporcion	444	60	13	FALSO	5	10	LogisticRegression	{'classifier__C': np.float64(0.35849855803404745), 'classifier__class_weight': None, 'classifier__dual': False, 'classifier__fit_intercept': True, 'classifier__penalty': 'l1', 'classifier__solver': 'liblinear', 'classifier__tol': np.float64(2.1930485556643678e-05)}	0,5483699	0,556177289	0,59055122	0,586627729	0,254019665	0,262824675	0,2672601	0,262966333	2,66	4
heart	base	base				207	90	13	FALSO	5	10	LogisticRegression	{'classifier__C': np.float64(0.4042872735027334), 'classifier__class_weight': 'balanced', 'classifier__dual': False, 'classifier__fit_intercept': True, 'classifier__penalty': 'l2', 'classifier__solver': 'liblinear', 'classifier__tol': np.float64(0.00087060208783048 54)}	0,31236192	0,326200452	0,30921155	0,301878153	0,374784041	0,407575758	0,37472798	0,370748299	1,339	4