



Aprendizaje de Máquina I

Trabajo Práctico Grupal

*Ribet, Juan Ignacio
Sarmiento, Fabián
Cáceres, Martín Nahuel
Raczkowski, Karen Elizabeth*

- 1. DATASET**
- 2. EVALUACIÓN DE LOS MODELOS**
- 3. SPLIT DATASET**
- 4. MODELOS**
- 5. COMPARACIÓN DE LOS MODELOS**
- 6. AUTO ML**
- 7. CONCLUSIONES**

- 1. DATASET**
- 2. EVALUACIÓN DE LOS MODELOS**
- 3. SPLIT DATASET**
- 4. MODELOS**
- 5. COMPARACIÓN DE LOS MODELOS**
- 6. AUTO ML**
- 7. CONCLUSIONES**

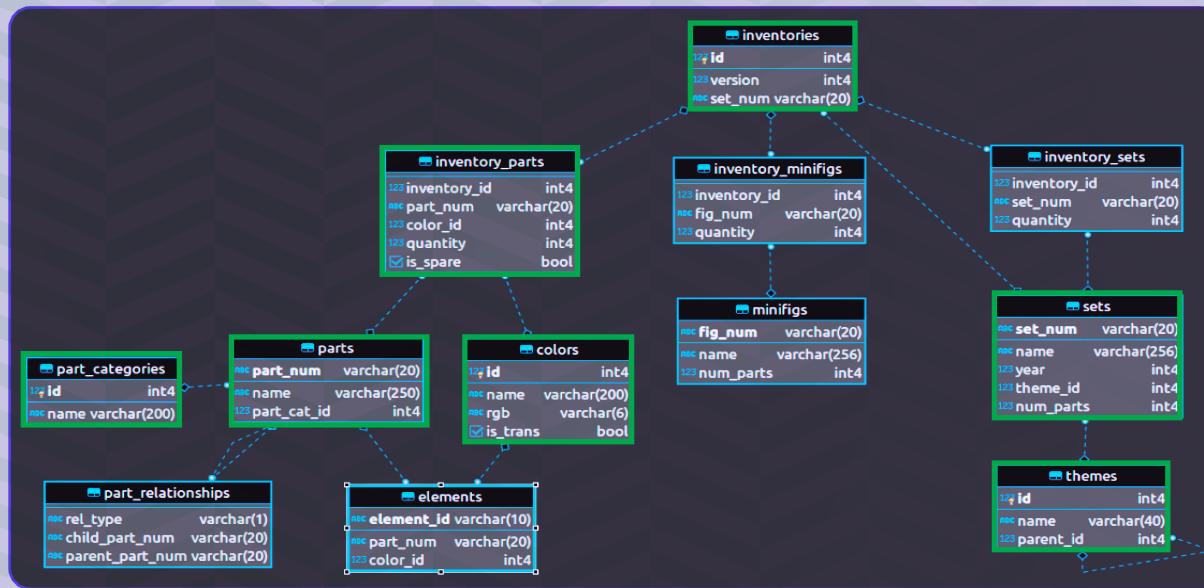


● DATASET

1.0.

LEGO DATABASE

<https://www.kaggle.com/datasets/rtatman/lego-database?resource=download>



Después de combinar los datasets mencionados, se analizaron las temáticas más frecuentes en los sets de Lego para trabajar sólo con ellas y reducir la cantidad de categorías. Además, se procesaron los datos para obtener la proporción de piezas de cada material, de piezas de cada categoría y de piezas de cada uno de los 10 colores más frecuentes presentes en cada set de Lego.

● DATASET



Se seleccionaron **12 features** basandonos en top 20 por MI y luego eliminando features correlacionadas entre si (Spearman)

'Plates Special'

'Bricks'

'Duplo, Quattro and Primo'

'Minifig Accessories'

'gris'

'Bricks Sloped'

'negro'

'Transportation - Land'

'Technic Pins'

'rojo'

'amarillo'

'Bars, Ladders and Fences'

● DATASET



1.1.

PREGUNTA A RESPONDER

- ¿Podría predecir a qué temática pertenece un set basado en el contenido de este?

Se trabajará en un problema de clasificación multiclas.

Para responder la pregunta planteada se utilizarán **los siguientes datasets** los cuales darán la información necesaria para trabajar:

• themes

• sets

• inventories

• inventory_parts

• colors

• parts

• part_categories

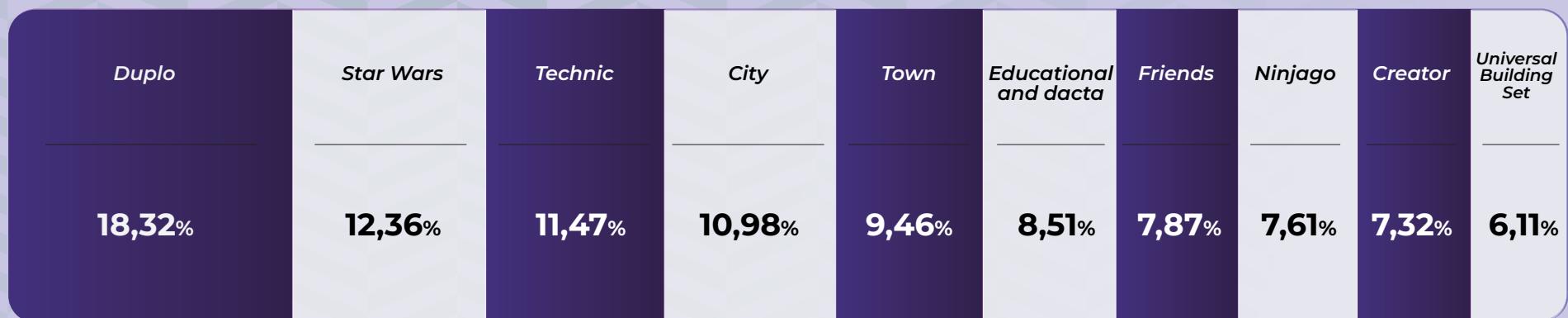
● DATASET



1.2.

BALANCE DE CLASES

Se verifica la participación de cada clase en el dataset.



- 
1. DATASET
 2. EVALUACIÓN DE LOS MODELOS
 3. SPLIT DATASET
 4. MODELOS
 5. COMPARACIÓN DE LOS MODELOS
 6. AUTO ML
 7. CONCLUSIONES

● EVALUACIÓN DE LOS MODELOS



Para evaluar la performance de los diferentes modelos con los que se trabajará, se crea una función llamada “**evaluar**”. El propósito de la misma está destinado a calcular y mostrar las diferentes métricas de evaluación, así como las matrices de confusión para el modelo de clasificación en los datos de entrenamiento y prueba.

Las métricas de evaluación con las que se trabajará son:

- a • Accuracy
- b • Precision
- c • Recall
- d • F1
- e • Matriz de confusión | Train Data
- f • Matriz de confusión | Test Data

- 
1. DATASET
 2. EVALUACIÓN DE LOS MODELOS
 3. SPLIT DATASET
 4. MODELOS
 5. COMPARACIÓN DE LOS MODELOS
 6. AUTO ML
 7. CONCLUSIONES

- SPLIT DATASET



Se separa el dataset en **entrenamiento** y **prueba**.

70 %

TRAINING

30 %

TESTING

- 
1. DATASET
 2. EVALUACIÓN DE LOS MODELOS
 3. SPLIT DATASET
 4. MODELOS
 5. COMPARACIÓN DE LOS MODELOS
 6. AUTO ML
 7. CONCLUSIONES

● MODELOS



MODELOS SELECCIONADOS

4.1. Logistic Regression CV (Balanced - ROS - SMOTE)

4.2. XGB Classifier

4.3. Cat Boost Classifier

4.4. Random Forest Classifier

4.5. Bagging - Tree Classifier

4.6. SVM

4.7. Decision Tree Classifier

Se seleccionaron diferentes modelos en los cuales se evaluará los mismos parámetros en cada caso:

- a • Accuracy**
- b • Precision**
- c • Recall**
- d • F1**
- e • Matriz de confusión | Train Data**
- f • Matriz de confusión | Test Data**



4.1.1.

LOGISTIC REGRESSION CV BALANCED

Para cada modelo se implementaron diferentes métodos de balanceo. Aquí se presentan los resultados del modelo de **LG** utilizando la técnica de **validación cruzada** (cross-validation) y con la opción de **balanceo de clases**.

Accuracy

0,7004

Precision

0,6728

Recall

0,7004

F1

0,6801

	precision	recall	f1-score	support
City	0.82	0.66	0.73	238
Creator	0.53	0.58	0.56	173
Duplo	0.84	0.96	0.90	392
Educational and Dacta	0.12	0.04	0.06	160
Friends	0.66	0.76	0.70	156
Ninjago	0.61	0.57	0.59	147
Star Wars	0.68	0.56	0.61	230
Technic	0.70	0.90	0.79	222
Town	0.79	0.79	0.79	185
Universal Building Set	0.61	0.83	0.71	117
accuracy			0.70	2020
macro avg	0.64	0.67	0.64	2020
weighted avg	0.67	0.70	0.68	2020

● MODELOS



MATRIZ DE CONFUSIÓN | TRAIN DATA

		Predicted										
		City	Creator	Duplo	Educational and Dacta	Friends	Ninjago	Star Wars	Technic	Town	Universal Building Set	
Actual	City	310	52	0	13	19	32	31	10	34	0	800
	Creator	15	188	0	2	30	15	19	3	7	41	700
	Duplo	0	0	802	38	0	0	0	1	0	0	600
	Educational and Dacta	7	24	163	31	5	4	3	112	21	43	500
	Friends	19	15	0	1	295	18	14	1	9	2	400
	Ninjago	15	12	0	8	18	222	77	1	9	3	300
	Star Wars	21	68	0	4	47	70	332	43	14	3	200
	Technic	0	4	0	22	3	2	3	501	3	12	100
	Town	16	2	0	6	7	18	0	20	360	23	50
	Universal Building Set	0	4	0	1	0	0	0	27	11	251	0

MATRIZ DE CONFUSIÓN | TEST DATA

		Predicted										
		City	Creator	Duplo	Educational and Dacta	Friends	Ninjago	Star Wars	Technic	Town	Universal Building Set	
Actual	City	158	25	0	3	7	9	15	3	17	1	350
	Creator	7	101	0	2	19	7	13	1	1	22	300
	Duplo	0	0	375	16	0	0	0	1	0	0	250
	Educational and Dacta	1	12	70	7	4	1	0	40	5	20	200
	Friends	11	4	0	1	118	8	9	1	4	0	150
	Ninjago	5	11	0	4	11	84	23	2	7	0	100
	Star Wars	4	32	0	5	18	22	129	14	5	1	50
	Technic	0	0	0	16	0	0	1	199	1	5	0
	Town	6	2	0	3	2	6	0	7	147	12	50
	Universal Building Set	0	3	0	0	1	0	0	16	0	97	0

4.1.2.

LOGISTIC REGRESSION CV WITH ROS

Aquí se utiliza otro se implementa el método de **sobremuestreo aleatorio** para abordar el desequilibrio de clases en los datos de entrenamiento y se aplica el sobremuestreo solo a la clase minoritaria.

Accuracy

0,6193

Precision

0,5943

Recall

0,6193

F1

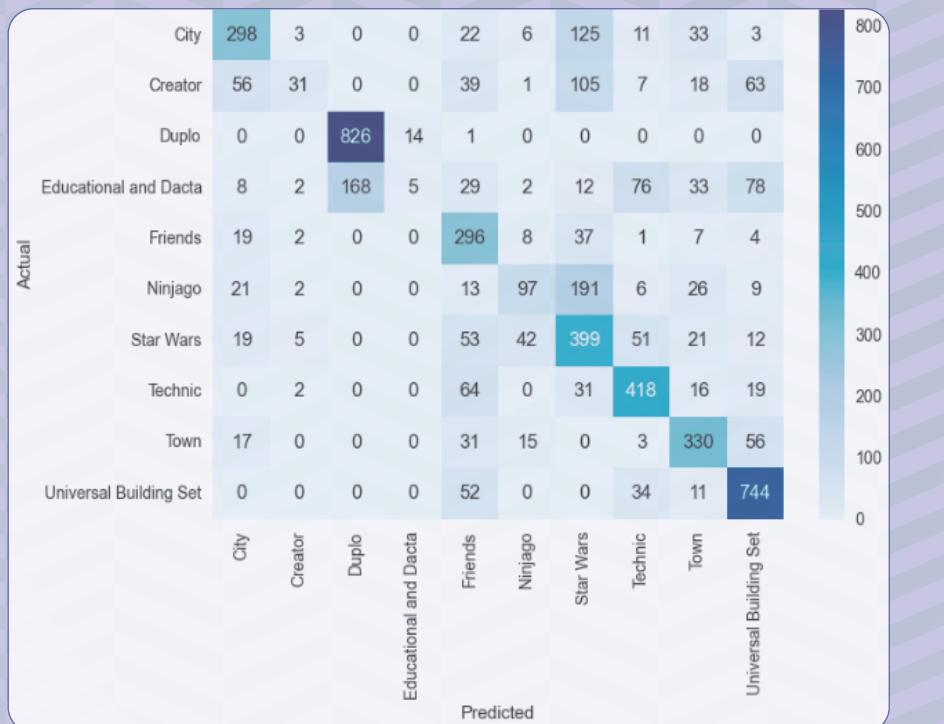
0,5774

	precision	recall	f1-score	support
City	0.71	0.58	0.64	238
Creator	0.59	0.11	0.19	173
Duplo	0.84	0.97	0.90	392
Educational and Dacta	0.00	0.00	0.00	160
Friends	0.45	0.76	0.57	156
Ninjago	0.64	0.31	0.42	147
Star Wars	0.42	0.68	0.52	230
Technic	0.70	0.69	0.70	222
Town	0.66	0.74	0.70	185
Universal Building Set	0.50	0.85	0.63	117
accuracy			0.62	2020
macro avg	0.55	0.57	0.53	2020
weighted avg	0.59	0.62	0.58	2020

● MODELOS



MATRIZ DE CONFUSIÓN | TRAIN DATA



MATRIZ DE CONFUSIÓN | TEST DATA



4.1.3.

LOGISTIC REGRESSION CV WITH SMOTE

Aquí se utiliza **LR** con la técnica de Sobremuestreo **SMOTE** (Synthetic Minority Over-sampling Technique) para abordar el desequilibrio de clases.

Accuracy

0,7009

Precision

0,6758

Recall

0,7009

F1

0,6822

	precision	recall	f1-score	support
City	0.82	0.66	0.73	238
Creator	0.52	0.58	0.55	173
Duplo	0.84	0.95	0.89	392
Educational and Dacta	0.15	0.06	0.08	160
Friends	0.68	0.73	0.71	156
Ninjago	0.60	0.60	0.60	147
Star Wars	0.68	0.57	0.62	230
Technic	0.70	0.89	0.78	222
Town	0.80	0.80	0.80	185
Universal Building Set	0.62	0.83	0.71	117
accuracy			0.70	2020
macro avg	0.64	0.67	0.65	2020
weighted avg	0.68	0.70	0.68	2020

● MODELOS



MATRIZ DE CONFUSIÓN | TRAIN DATA

		Predicted										Actual	Count
		City	Creator	Duplo	Educational and Dacta	Friends	Ninjago	Star Wars	Technic	Town	Universal Building Set		
Actual	City	572	75	0	19	30	44	44	12	44	1	800	800
	Creator	30	550	0	4	64	33	46	3	14	97	700	700
	Duplo	0	0	806	34	0	0	0	1	0	0	600	600
	Educational and Dacta	12	58	335	78	8	12	1	226	35	76	500	500
	Friends	31	18	0	8	691	49	20	3	19	2	400	400
	Ninjago	27	22	0	14	29	546	171	1	25	6	300	300
	Star Wars	28	93	0	7	70	97	455	66	22	3	200	200
	Technic	0	7	0	35	4	4	5	764	3	19	100	100
	Town	27	7	0	9	7	35	0	31	688	37	50	50
	Universal Building Set	0	13	0	2	2	0	0	90	30	704	0	0

MATRIZ DE CONFUSIÓN | TEST DATA

		Predicted										Actual	Count
		City	Creator	Duplo	Educational and Dacta	Friends	Ninjago	Star Wars	Technic	Town	Universal Building Set		
Actual	City	158	24	0	5	7	8	16	3	17	0	350	350
	Creator	9	100	0	2	17	7	13	2	1	22	300	300
	Duplo	0	0	374	17	0	0	0	1	0	0	250	250
	Educational and Dacta	1	11	70	9	4	2	0	38	5	20	200	200
	Friends	11	6	0	1	114	11	8	1	4	0	150	150
	Ninjago	5	11	0	4	10	88	22	2	5	0	100	100
	Star Wars	3	34	0	5	14	24	130	14	5	1	50	50
	Technic	0	0	0	16	0	0	2	198	1	5	0	0
	Town	6	2	0	3	1	6	0	7	148	12	50	50
	Universal Building Set	0	3	0	0	0	0	0	17	0	97	0	0



4.2. | XGB CLASSIFIER

Accuracy

0,7643

Precision

0,7481

Recall

0,7643

F1

0,7492

	precision	recall	f1-score	support
--	-----------	--------	----------	---------

City	0.76	0.75	0.76	238
Creator	0.71	0.62	0.66	173
Duplo	0.86	0.96	0.91	392
Educational and Dacta	0.47	0.20	0.28	160
Friends	0.74	0.72	0.73	156
Ninjago	0.70	0.66	0.68	147
Star Wars	0.70	0.74	0.72	230
Technic	0.76	0.96	0.85	222
Town	0.85	0.88	0.86	185
Universal Building Set	0.76	0.82	0.79	117
accuracy			0.76	2020
macro avg	0.73	0.73	0.72	2020
weighted avg	0.75	0.76	0.75	2020

● MODELOS



MATRIZ DE CONFUSIÓN | TRAIN DATA



MATRIZ DE CONFUSIÓN | TEST DATA



4.3. CATBOOST CLASSIFIER

Accuracy

0,7702

Precision

0,7587

Recall

0,7702

F1

0,7608

	precision	recall	f1-score	support
--	-----------	--------	----------	---------

City	0.82	0.76	0.79	238
Creator	0.71	0.66	0.68	173
Duplo	0.87	0.91	0.89	392
Educational and Dacta	0.45	0.26	0.33	160
Friends	0.75	0.80	0.78	156
Ninjago	0.69	0.71	0.70	147
Star Wars	0.74	0.71	0.73	230
Technic	0.77	0.95	0.85	222
Town	0.85	0.88	0.86	185
Universal Building Set	0.73	0.82	0.77	117
accuracy			0.77	2020
macro avg	0.74	0.75	0.74	2020
weighted avg	0.76	0.77	0.76	2020

● MODELOS



MATRIZ DE CONFUSIÓN | TRAIN DATA



MATRIZ DE CONFUSIÓN | TEST DATA



● MODELOS

4.4. RANDOM FOREST CLASSIFIER

Accuracy

0,7707

Precision

0,7598

Recall

0,7707

F1

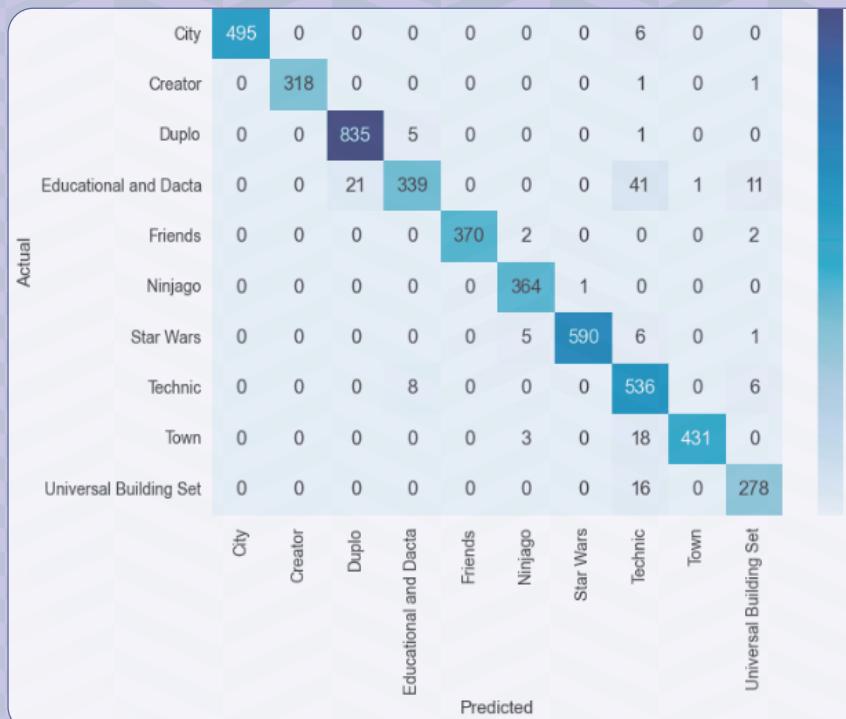
0,7547

	precision	recall	f1-score	support
City	0.77	0.76	0.77	238
Creator	0.80	0.57	0.67	173
Duplo	0.86	0.97	0.91	392
Educational and Dacta	0.57	0.21	0.30	160
Friends	0.72	0.72	0.72	156
Ninjago	0.69	0.67	0.68	147
Star Wars	0.69	0.77	0.73	230
Technic	0.77	0.95	0.85	222
Town	0.83	0.89	0.86	185
Universal Building Set	0.75	0.85	0.80	117
accuracy			0.77	2020
macro avg	0.74	0.74	0.73	2020
weighted avg	0.76	0.77	0.75	2020

● MODELOS



MATRIZ DE CONFUSIÓN | TRAIN DATA



MATRIZ DE CONFUSIÓN | TEST DATA



4.5. DECISION TREE BAGGING

Accuracy

0,7569

Precision

0,7441

Recall

0,7569

F1

0,7435

	precision	recall	f1-score	support
Educational and Dacta	City	0.77	0.73	0.75
	Creator	0.73	0.58	0.65
	Duplo	0.87	0.97	0.92
	Friends	0.51	0.23	0.32
	Ninjago	0.74	0.71	0.73
	Star Wars	0.63	0.64	0.64
	Technic	0.65	0.71	0.68
	Town	0.76	0.94	0.84
	Universal Building Set	0.82	0.89	0.85
	accuracy			117
		macro avg	0.76	2020
		weighted avg	0.73	2020
			0.72	2020
			0.74	2020

● MODELOS



MATRIZ DE CONFUSIÓN | TRAIN DATA

		Predicted										Actual	Count
		City	Creator	Duplo	Educational and Dacta	Friends	Ninjago	Star Wars	Technic	Town	Universal Building Set		
Actual	City	495	0	0	0	0	0	0	6	0	0	800	
	Creator	0	318	0	0	0	0	0	1	0	1	700	
	Duplo	0	0	837	3	0	0	0	1	0	0	600	
	Educational and Dacta	0	0	23	337	0	0	0	43	1	9	500	
	Friends	0	0	0	0	370	2	0	0	0	2	400	
	Ninjago	0	0	0	0	0	360	5	0	0	0	300	
	Star Wars	0	0	0	0	0	0	595	6	0	1	200	
	Technic	0	0	0	8	0	0	0	542	0	0	100	
	Town	0	0	0	0	0	3	0	18	431	0	50	
	Universal Building Set	0	0	0	0	0	0	21	0	273		0	

MATRIZ DE CONFUSIÓN | TEST DATA

		Predicted										Actual	Count
		City	Creator	Duplo	Educational and Dacta	Friends	Ninjago	Star Wars	Technic	Town	Universal Building Set		
Actual	City	174	10	0	2	4	17	17	3	11	0	350	
	Creator	17	101	0	1	12	7	25	2	4	4	300	
	Duplo	0	0	381	11	0	0	0	0	0	0	250	
	Educational and Dacta	3	4	59	37	2	1	3	33	5	13	200	
	Friends	16	4	0	3	111	10	9	1	2	0	150	
	Ninjago	9	3	0	1	6	94	28	2	4	0	100	
	Star Wars	5	11	0	4	14	17	164	8	6	1	50	
	Technic	1	0	0	3	0	2	4	208	1	3	0	
	Town	1	2	0	4	1	1	3	6	164	3	0	
	Universal Building Set	0	3	0	6	0	0	11	2	95		0	

4.6. SUPPORT VECTOR MACHINE

Accuracy

0,7128

Precision

0,6841

Recall

0,7128

F1

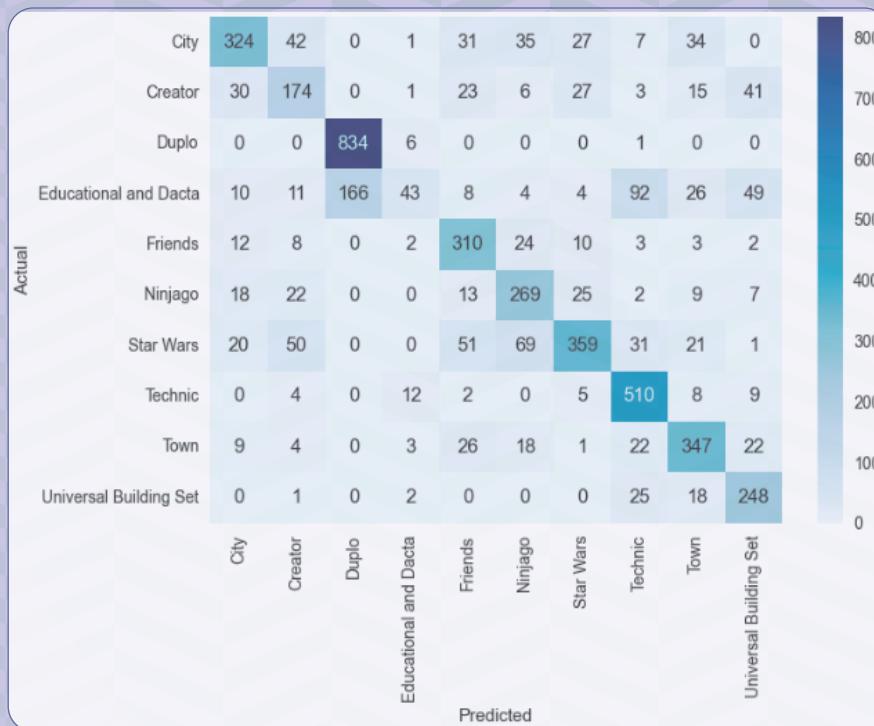
0,6865

	precision	recall	f1-score	support
City	0.78	0.63	0.70	238
Creator	0.54	0.52	0.53	173
Duplo	0.85	1.00	0.92	392
Educational and Dacta	0.26	0.04	0.07	160
Friends	0.62	0.79	0.69	156
Ninjago	0.64	0.69	0.67	147
Star Wars	0.74	0.60	0.67	230
Technic	0.73	0.92	0.82	222
Town	0.73	0.76	0.74	185
Universal Building Set	0.59	0.79	0.68	117
accuracy			0.71	2020
macro avg	0.65	0.68	0.65	2020
weighted avg	0.68	0.71	0.69	2020

● MODELOS



MATRIZ DE CONFUSIÓN | TRAIN DATA



MATRIZ DE CONFUSIÓN | TEST DATA



4.7. DECISION TREE CLASSIFIER

Accuracy

0,6762

Precision

0,6668

Recall

0,6762

F1

0,6694

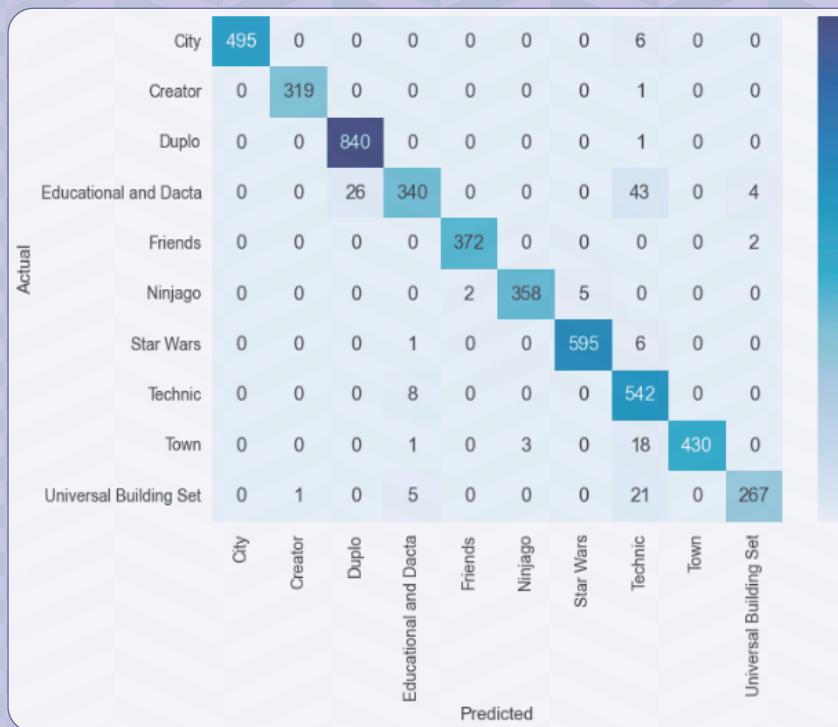
	precision	recall	f1-score	support
--	-----------	--------	----------	---------

City	0.65	0.61	0.63	238
Creator	0.56	0.49	0.52	173
Duplo	0.87	0.89	0.88	392
Educational and Dacta	0.32	0.25	0.28	160
Friends	0.61	0.60	0.60	156
Ninjago	0.59	0.54	0.57	147
Star Wars	0.56	0.61	0.58	230
Technic	0.73	0.92	0.82	222
Town	0.78	0.82	0.80	185
Universal Building Set	0.75	0.68	0.71	117
accuracy			0.68	2020
macro avg	0.64	0.64	0.64	2020
weighted avg	0.67	0.68	0.67	2020

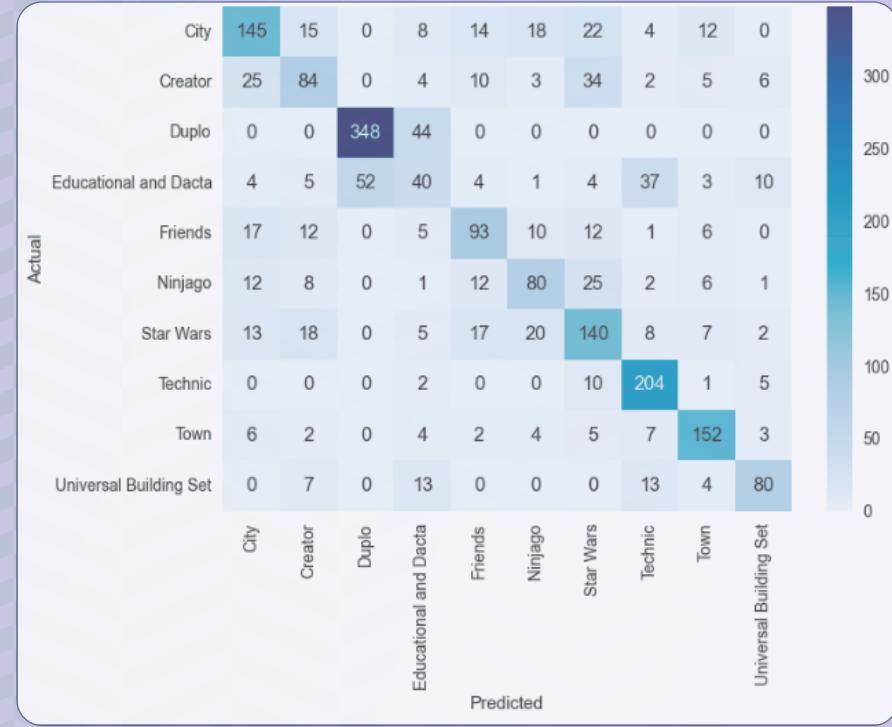
● MODELOS



MATRIZ DE CONFUSIÓN | TRAIN DATA



MATRIZ DE CONFUSIÓN | TEST DATA



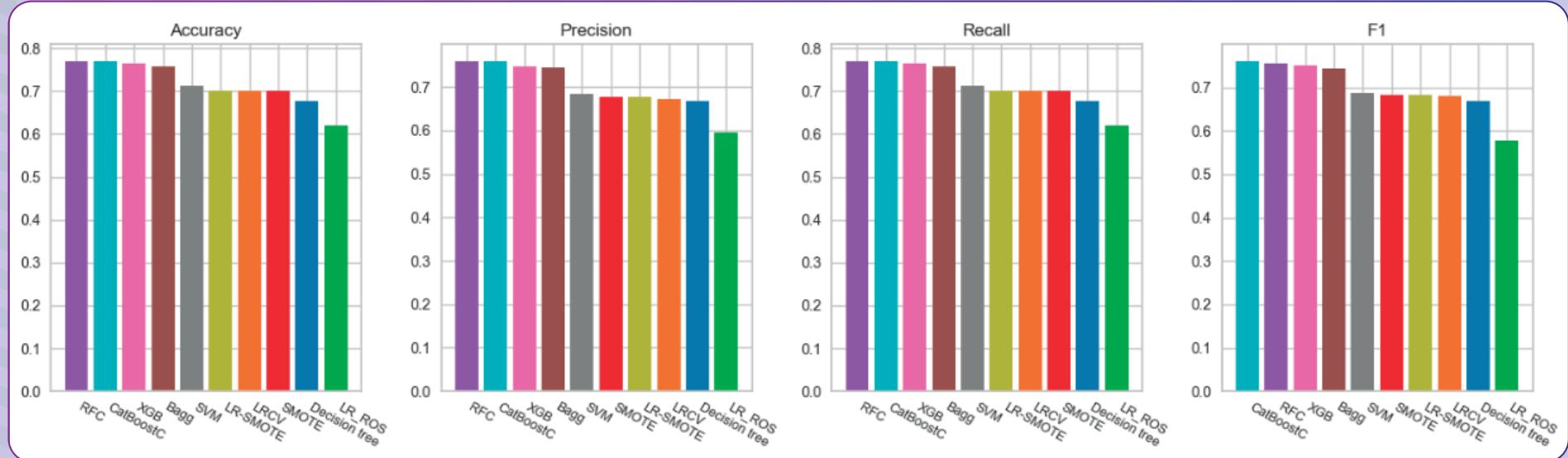
- 
1. DATASET
 2. EVALUACIÓN DE LOS MODELOS
 3. SPLIT DATASET
 4. MODELOS
 5. COMPARACIÓN DE LOS MODELOS
 6. AUTO ML
 7. CONCLUSIONES

● COMPARACIÓN DE LOS MODELOS



5.0.

COMPARACIÓN DE MÉTRICAS ENTRE LOS MODELOS



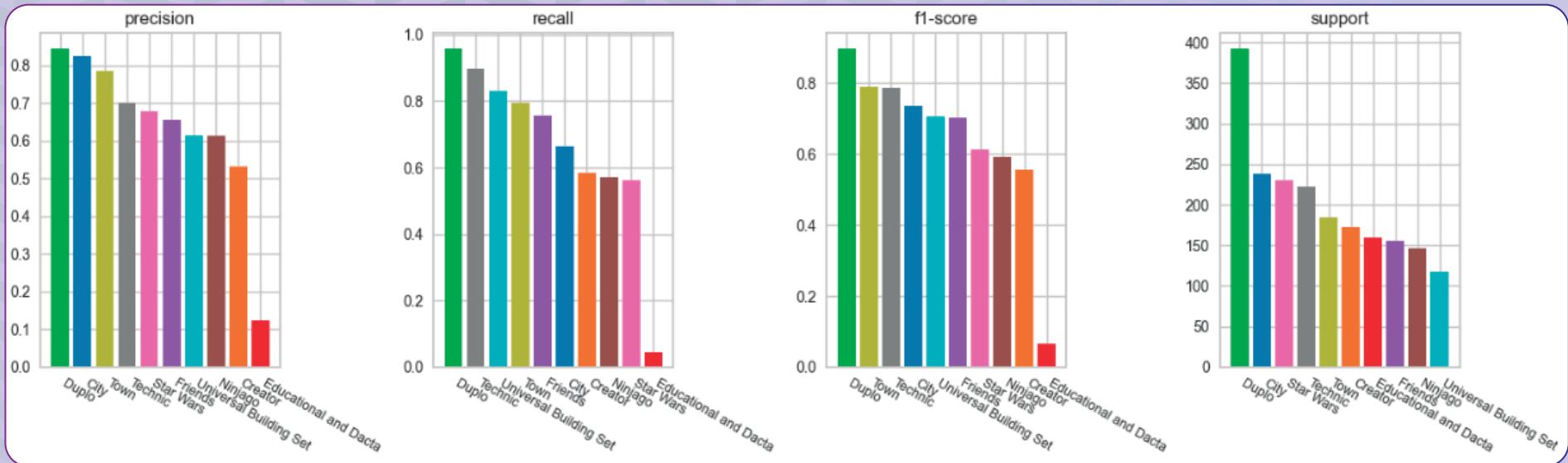
Aquí se comparan las métricas de Accuracy, Precision, Recall y F1 entre los diferentes modelos que se probaron, presentándose de forma descendente en función de su performance. Con esta forma de visualizar, se observa una buena performance entre los modelos Randon Forest Classifier y XGB classifier.

● COMPARACIÓN DE LOS MODELOS



5.1.1.

MÉTODO: LRCV BALANCED COMPARACIÓN DE MÉTRICAS ENTRE CLASES

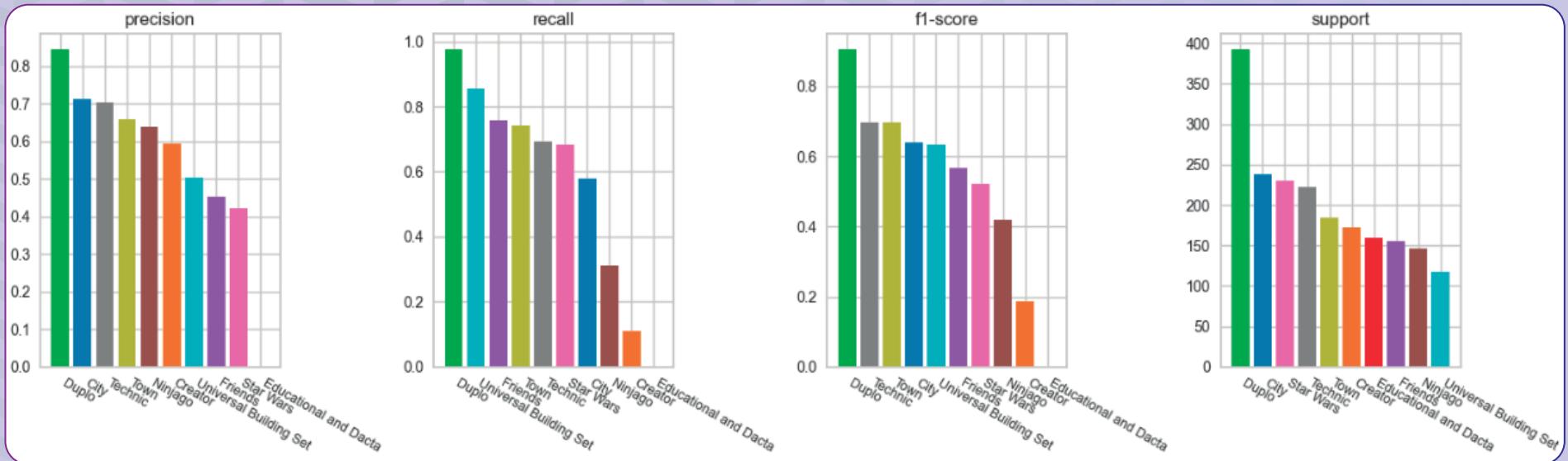


● COMPARACIÓN DE LOS MODELOS



5.1.2.

MÉTODO: LRCV WITH ROS COMPARACIÓN DE MÉTRICAS ENTRE CLASES

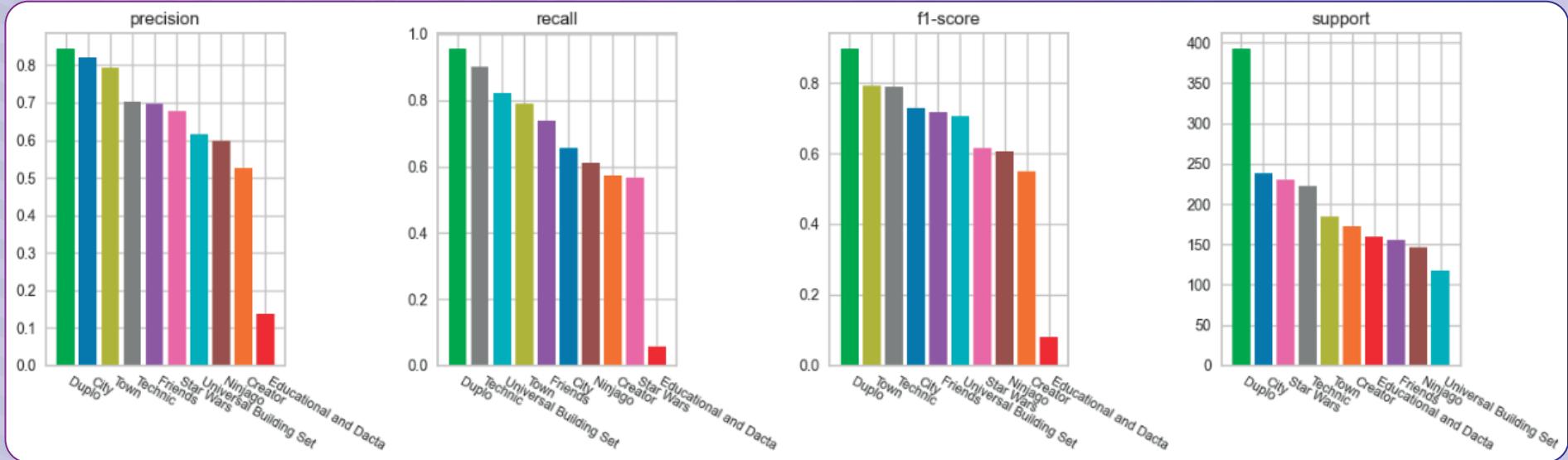


● COMPARACIÓN DE LOS MODELOS



5.1.3.

MÉTODO: LRCV WITH SMOTE COMPARACIÓN DE MÉTRICAS ENTRE CLASES

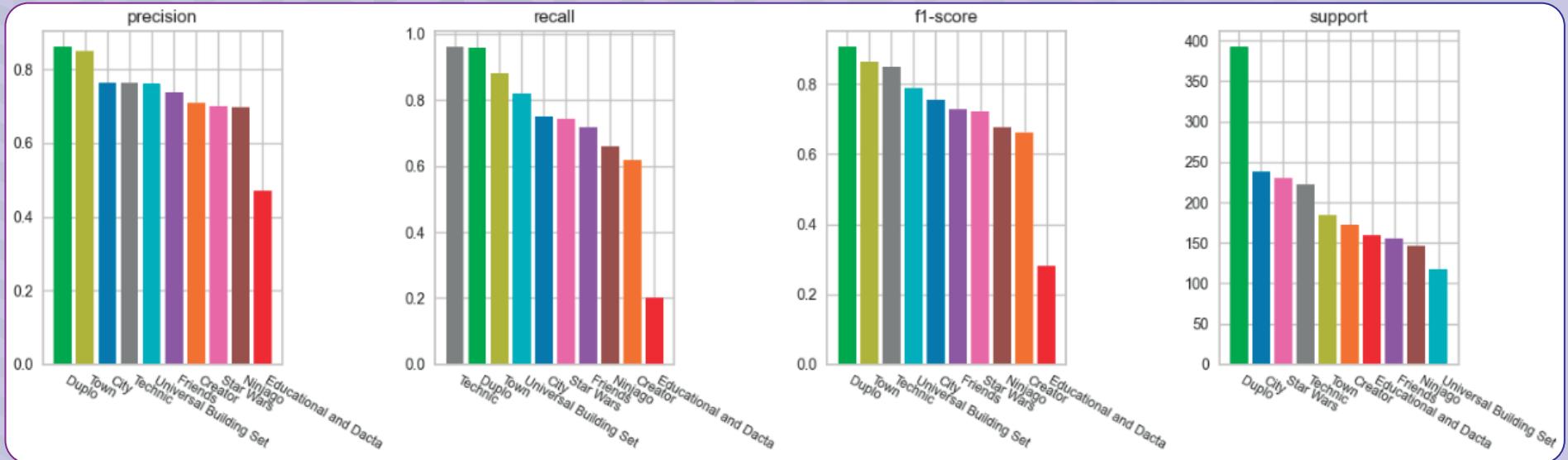


● COMPARACIÓN DE LOS MODELOS



5.2.

MÉTODO: XGB CLASSIFIER COMPARACIÓN DE MÉTRICAS ENTRE CLASES

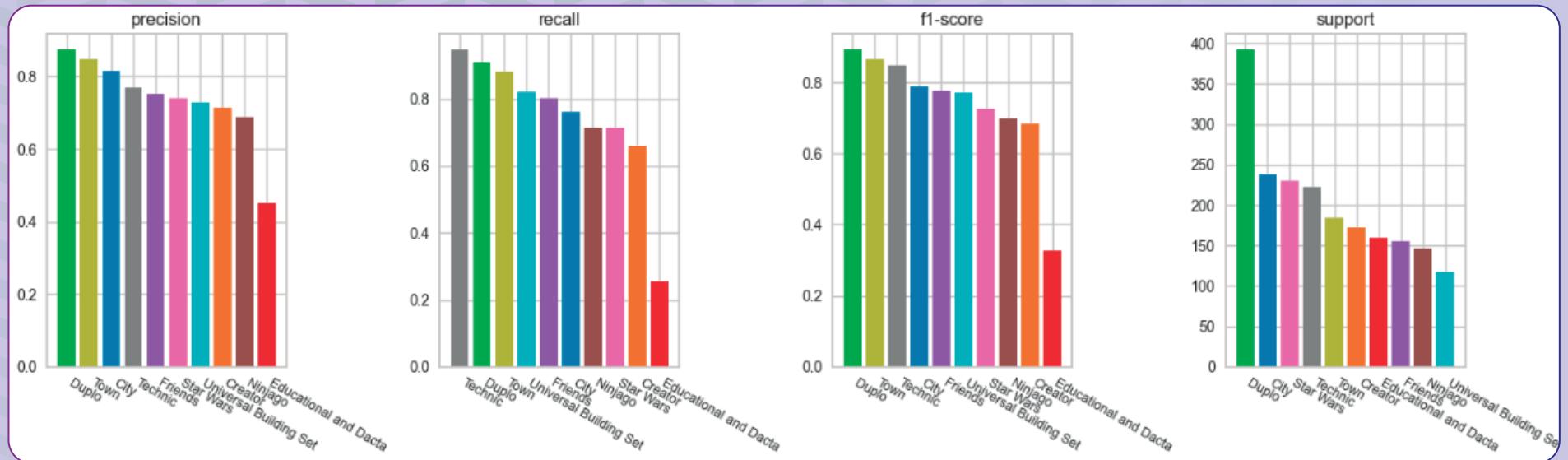


● COMPARACIÓN DE LOS MODELOS



5.3.

MÉTODO: CAT BOOST CLASSIFIER COMPARACIÓN DE MÉTRICAS ENTRE CLASES

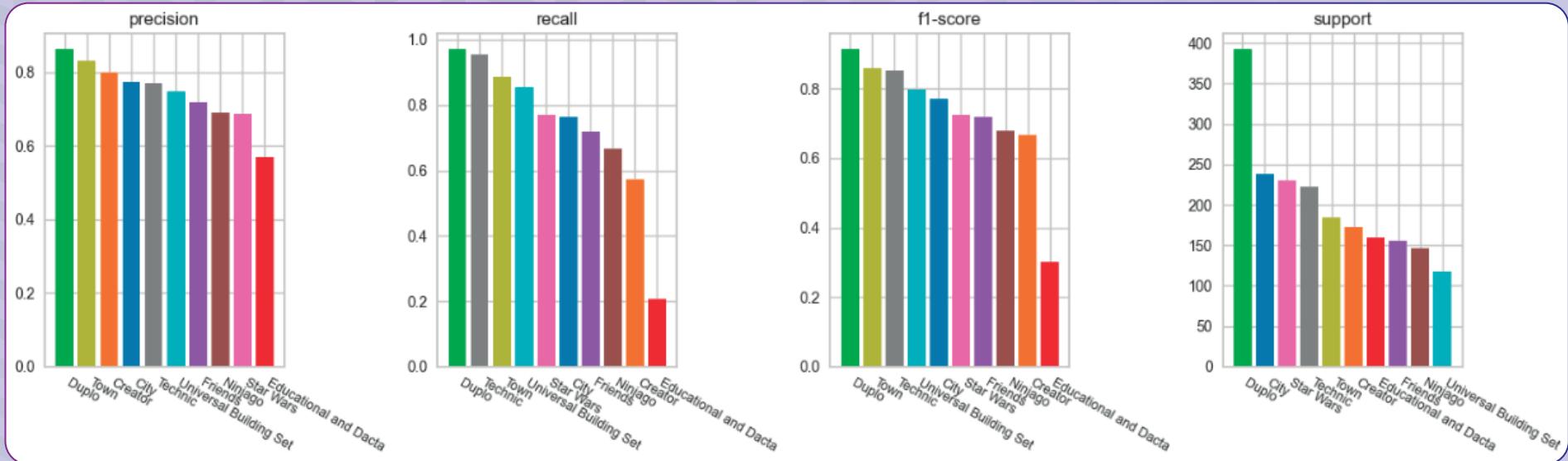


● COMPARACIÓN DE LOS MODELOS



5.4.

MÉTODO: RANDOM FOREST CLASSIFIER COMPARACIÓN DE MÉTRICAS ENTRE CLASES

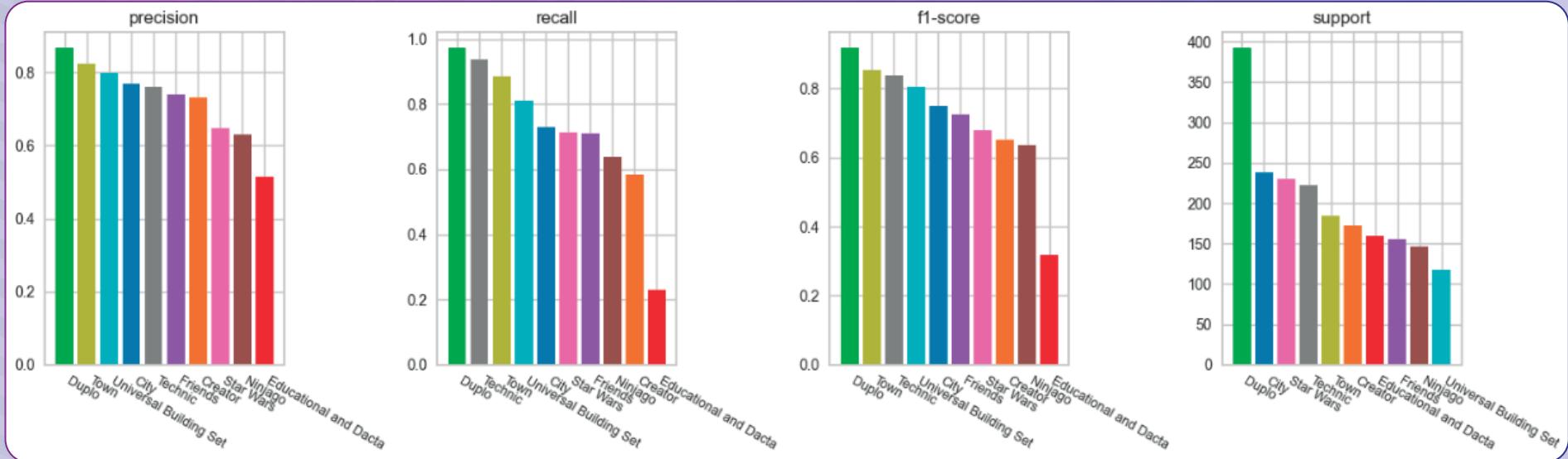


● COMPARACIÓN DE LOS MODELOS



5.5.

MÉTODO: BAGGING - TREE CLASSIFIER COMPARACIÓN DE MÉTRICAS ENTRE CLASES

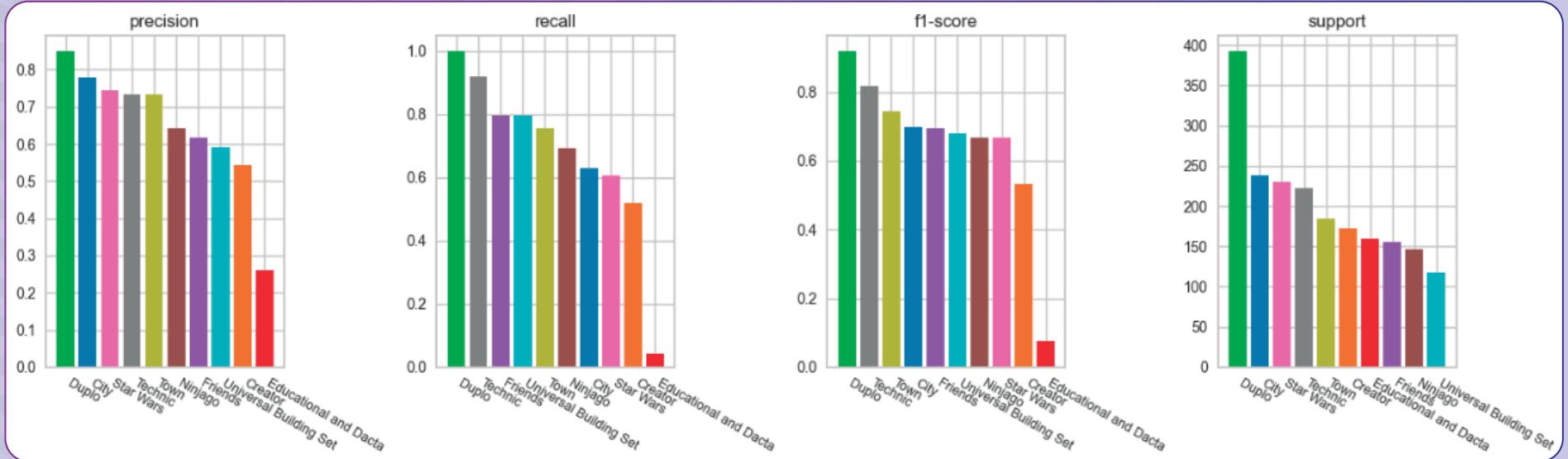


● COMPARACIÓN DE LOS MODELOS



5.6.

MÉTODO: SUPPORT VECTOR MACHINE COMPARACIÓN DE MÉTRICAS ENTRE CLASES

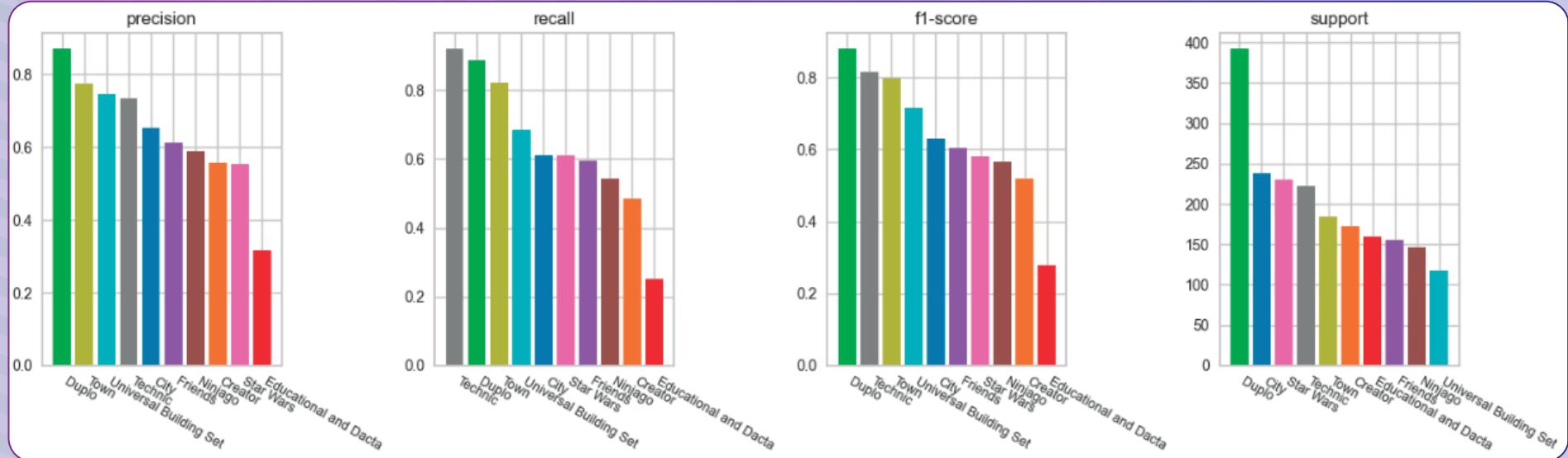


● COMPARACIÓN DE LOS MODELOS



5.7.

MÉTODO: DECISION TREE CLASSIFIER COMPARACIÓN DE MÉTRICAS ENTRE CLASES



- 1. DATASET**
- 2. EVALUACIÓN DE LOS MODELOS**
- 3. SPLIT DATASET**
- 4. MODELOS**
- 5. COMPARACIÓN DE LOS MODELOS**
- 6. AUTO ML**
- 7. CONCLUSIONES**



5.1.

COMPARACIÓN DE MODELOS

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
catboost	CatBoost Classifier	0.8784	0.9911	0.8784	0.8761	0.8742	0.8629	0.8636
lightgbm	Light Gradient Boosting Machine	0.8754	0.9881	0.8754	0.8737	0.8724	0.8596	0.8600
xgboost	Extreme Gradient Boosting	0.8741	0.9885	0.8741	0.8723	0.8706	0.8581	0.8587
rf	Random Forest Classifier	0.8510	0.9847	0.8510	0.8463	0.8423	0.8319	0.8331
et	Extra Trees Classifier	0.8497	0.9813	0.8497	0.8457	0.8408	0.8304	0.8317
gbc	Gradient Boosting Classifier	0.8495	0.9857	0.8495	0.8462	0.8445	0.8303	0.8311
lda	Linear Discriminant Analysis	0.7513	0.9460	0.7513	0.7356	0.7292	0.7192	0.7220
dt	Decision Tree Classifier	0.7494	0.8624	0.7494	0.7521	0.7490	0.7179	0.7183
ridge	Ridge Classifier	0.7209	0.0000	0.7209	0.7163	0.6834	0.6839	0.6898
lr	Logistic Regression	0.6787	0.9377	0.6787	0.6767	0.6553	0.6366	0.6420
nb	Naive Bayes	0.6776	0.9318	0.6776	0.6862	0.6487	0.6375	0.6460
qda	Quadratic Discriminant Analysis	0.6104	0.9042	0.6104	0.6468	0.5537	0.5623	0.5759
knn	K Neighbors Classifier	0.4003	0.7573	0.4003	0.4369	0.4014	0.3228	0.3280
ada	Ada Boost Classifier	0.2746	0.6976	0.2746	0.2304	0.2083	0.1704	0.2436
svm	SVM - Linear Kernel	0.2088	0.0000	0.2088	0.1519	0.1313	0.1040	0.1517
dummy	Dummy Classifier	0.1831	0.5000	0.1831	0.0335	0.0567	0.0000	0.0000



5.2.

CREACIÓN DEL MODELO CATBOOST CLASSIFIER

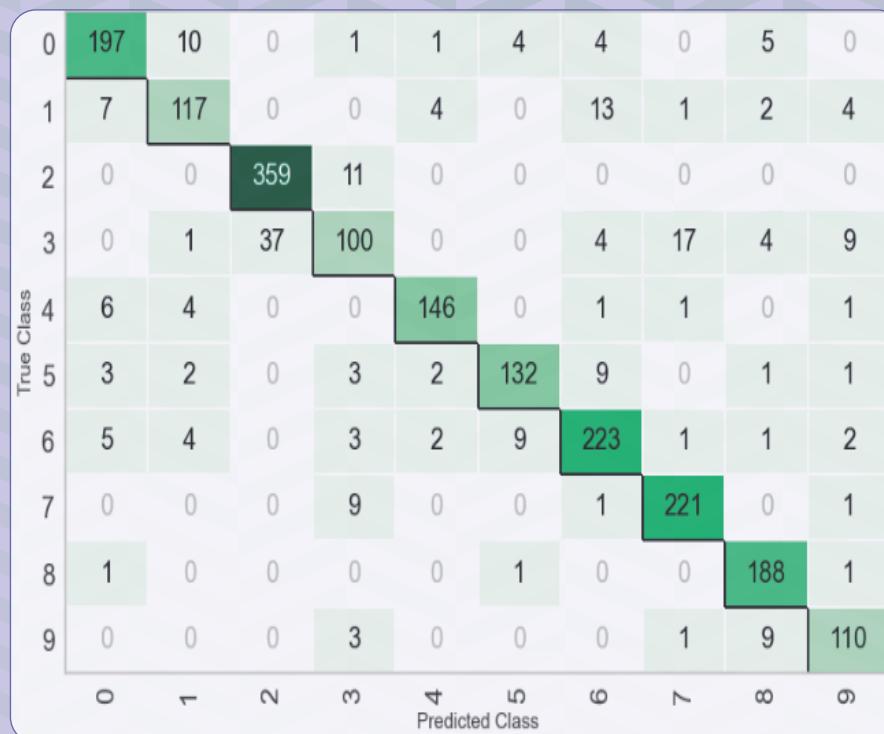
	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
Fold							
0	0.8814	0.9922	0.8814	0.8781	0.8782	0.8663	0.8667
1	0.8983	0.9908	0.8983	0.8955	0.8941	0.8854	0.8860
2	0.8854	0.9919	0.8854	0.8846	0.8795	0.8706	0.8717
3	0.8556	0.9871	0.8556	0.8516	0.8476	0.8370	0.8383
4	0.8811	0.9912	0.8811	0.8818	0.8783	0.8661	0.8667
5	0.8747	0.9912	0.8747	0.8739	0.8715	0.8588	0.8594
6	0.8556	0.9925	0.8556	0.8527	0.8519	0.8374	0.8379
7	0.8832	0.9907	0.8832	0.8784	0.8789	0.8684	0.8689
8	0.8747	0.9922	0.8747	0.8729	0.8724	0.8588	0.8592
9	0.8938	0.9908	0.8938	0.8921	0.8890	0.8802	0.8810
Mean	0.8784	0.9911	0.8784	0.8761	0.8742	0.8629	0.8636
Std	0.0134	0.0015	0.0134	0.0138	0.0138	0.0151	0.0150

5.3.

COMPARACIÓN DE MATRICES - TEST

CATBOOST | RANDOM FOREST

CATBOOST | AUTOML



RANDOM FOREST | MANUAL



- 
1. DATASET
 2. EVALUACIÓN DE LOS MODELOS
 3. SPLIT DATASET
 4. MODELOS
 5. COMPARACIÓN DE LOS MODELOS
 6. AUTO ML
 7. CONCLUSIONES

● CONCLUSIONES



En el presente trabajo se utilizaron varios modelos de clasificación para intentar predecir a qué temática pertenece un set de Lego basado en su contenido. Antes de empezar con el entrenamiento de los modelos, se realizó un análisis exploratorio, se combinaron varios datasets y se preprocesaron los datos. Luego, se seleccionaron 12 features basado en top 20 por MÍ y luego eliminando features correlacionadas entre sí (Spearman). Luego, se entrenaron diferentes modelos de aprendizaje automático (Decision Tree, Random Forest, XGBoost, entre otros) utilizando técnicas de balanceo de clases y un split train-test de 70-30.

Se observa que los modelos con mejor performance en este caso son Random Forest (accuracy de 0.77, precision de 0.76, recall de 0.77, F1 de 0.75) y CatBoost (accuracy de 0.77, precision de 0.76, recall de 0.77, F1 de 0.76). Estos resultados indican que estos modelos tienen un desempeño aceptable en la clasificación de los sets de Lego en diferentes temáticas. Esto significa que es posible predecir con cierta precisión a qué temática pertenece un set de Lego analizando su contenido, en este caso, las proporciones de materiales, colores y categorías de las piezas.

● CONCLUSIONES



La implementación de AutoML muestra que la mejor performance fue la del CatBoost (accuracy de 0.88, precision de 0.88, recall de 0.88, F1 de 0.87). Se puede concluir que hubo una mejora significativa en la performance en comparación con el modelo Random Forest Classifier previo, que tenía un accuracy de 0.77. Por un lado, utilizar AutoML nos permitió automatizar gran parte del proceso de selección y entrenamiento de modelos, lo que ahorra tiempo y recursos. Sin embargo, es importante considerar las limitaciones y desventajas asociadas, como son la pérdida de control y comprensión sobre el proceso.

¡muchas gracias!

