Centro de Investigación en Matemáticas (CIMAT). Unidad Monterrey

Tarea 6

Marcelo Alberto Sanchez Zaragoza

25 de mayo de 2021

1. Problema 1

Nos menciona que ahora tomemos los datos de los dígitos escritos a mano y digitalizados($MNIST$) de 28x28 pixeles, que previamente habiamos trabajado. Se observa que hay un total de 10 grupos de datos ya que tenemos K $= 0, 1, ..., 9$ en la figura 1.1 se ilustra como son los distintos tipos de datos.

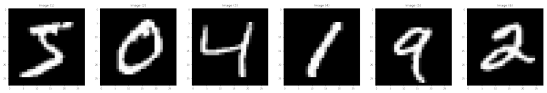

Figura 1.1: Dígitos escritos a mano.

Se nos pide aplicar los métodos de clasificación basados en: Redes neuronales, Maquinas de Soporte Vectorial, Arboles de clasificación y AdaBoost, donde usaremos validación cruzada(K-FOLD CV) como criterio para elegir los mejores parámetros y poder comparar nuestros métodos. El total de imagenes es de 70,000 con 784 variables o caracteristicas, al realizar una exploración previa se observo que al tomar todas las caracteristicas los distintos métodos de clasificación que mencionamos tardaban mucho por lo que se opto por una representación adecuada de los datos, es decir, reducir la dimensión de los datos para poder manipularlos mejor. Cabe mencionar que el conjunto de datos se dividio en dos grupos, los dos grupos que tomamos fue de entrenamiento y de prueba, para el primer grupo tomamos el 90 % y el segundo de 10 %.

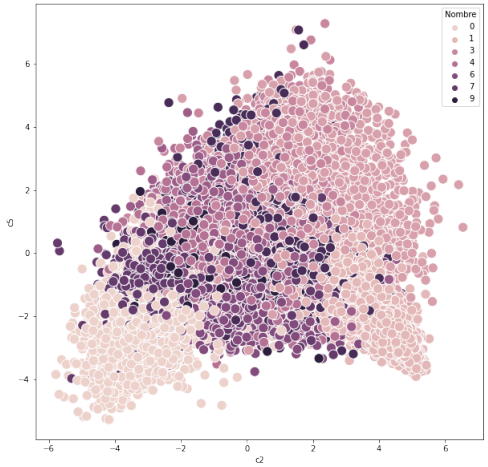

Figura 1.2: Representación en el segundo y quinto componentes en LDA.

Al buscar varias representaciones contramos la transformación LDA, donde nos reduce los datos en 9 caracteristicas por lo cual es mejor para nuestros métodos de clasificación. En la figura 1.2 se ilustra solo la repre-

sentación en dos componentes(segundo y quito).
Una vez que tenemos nuestros datos en una dimensión menor, procedemos a ocupar los distintos métodos que nos solicitan.

Redes Neuronales

Al realizar la aplicación de Redes Neuronales encontramos los mejores parámetros para el modelo, los cuales son los siguientes: número de capas = [10] y alpha = 0.05. Dandonos una proporción de datos clasificados correctamente de 0.90.
En la figura 1.3 se muestra la matriz de confunsión, se puede observar que muchos de los valores encontrados coinciden con los originales.

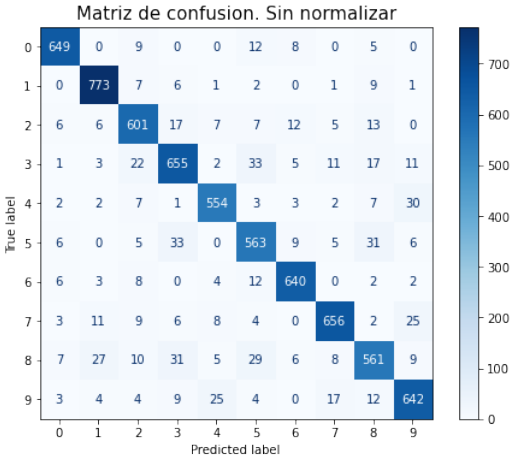

Figura 1.3: Matriz de confunsión-Redes Neuronales.

En la figura 1.4 mostramos como nuestro método fue bueno al buscar el digito de cada dato ya que las barras se muestran muy cercanas.

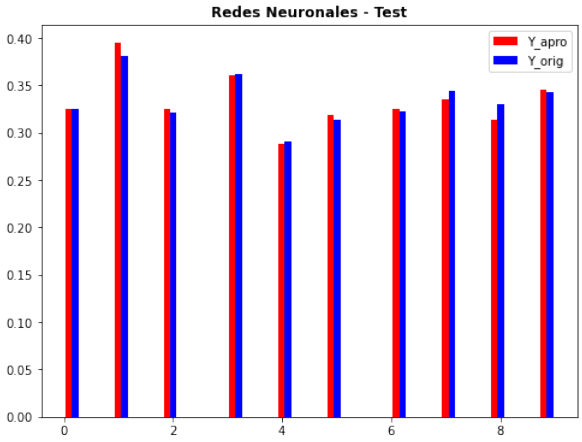

Figura 1.4: Redes Neuronales con datos de prueba.

En la siguiente tabla observamos que nuestros valores para las distintas métricas. Los valores que se observan en las métricas fueron bueno para cada digito.

	precision	recall	f1-score	support
0	0.95	0.95	0.95	683
1	0.93	0.97	0.95	800
2	0.88	0.89	0.89	674
3	0.86	0.86	0.86	760
4	0.91	0.91	0.91	611
5	0.84	0.86	0.85	658
6	0.94	0.95	0.94	677
7	0.93	0.91	0.92	724
8	0.85	0.81	0.83	693
9	0.88	0.89	0.89	720
accuracy			0.90	7000

macro avg	0.90	0.90	0.90	7000
weighted avg	0.90	0.90	0.90	7000

Máquinas de Soporte Vectorial

Al buscar los mejores parámetros para nuestro modelo encontramos que el kernel adecuado es gaussiano y un valor de C = 2.5. El resultado de ocupar dicho parámetros nos dio una proporción de datos clasificados correctamente de 0.92. En la figura 1.5 se muestra la matriz de confunsión.

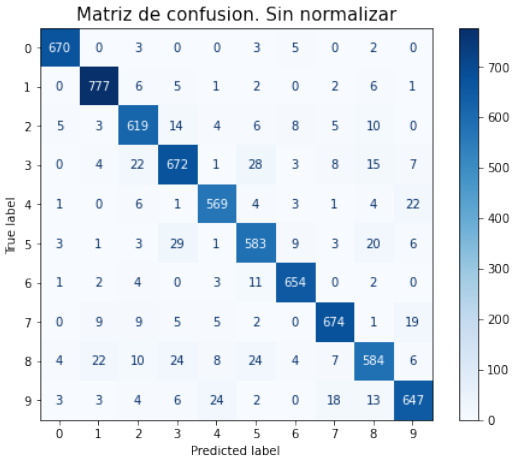

Figura 1.5: Matriz de confunsión-Máquinas de Soporte Vectorial.

En la figura 1.6 se ilustra el gráfico de barras, los resultados fueron buenos igual para este método pero mejores que con Redes Neuronales.

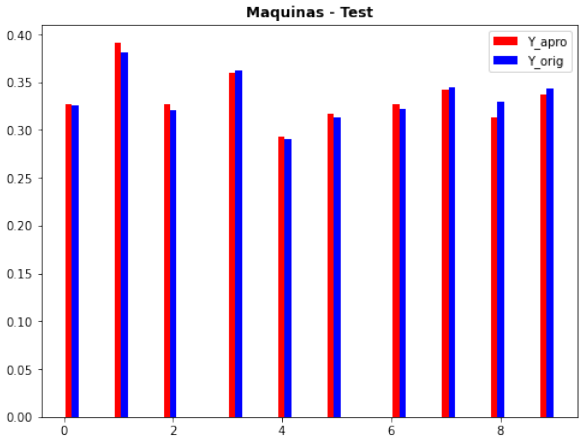

Figura 1.6: Máquinas de Soporte Vectorial con datos de prueba.

En la siguiente tabla observamos que nuestros valores para las distintas métricas. Los valores que se observan en las métricas fueron bueno para cada digito.

	precision	recall	f1-score	support
0	0.98	0.98	0.98	683
1	0.95	0.97	0.96	800
2	0.90	0.92	0.91	674
3	0.89	0.88	0.89	760
4	0.92	0.93	0.93	611
5	0.88	0.89	0.88	658
6	0.95	0.97	0.96	677
7	0.94	0.93	0.93	724
8	0.89	0.84	0.87	693
9	0.91	0.90	0.91	720
accuracy			0.92	7000
macro avg	0.92	0.92	0.92	7000

```
weighted avg     0.92      0.92      0.92      7000
```

Arboles de clasificación

Al buscar los mejores parámetros para nuestro modelo encontramos que la profundidad máxima, elementos mínimos en nodos y un alpha de $1 * 10^{-5}$. El resultado de ocupar dicho parámetros nos dio una proporción de datos clasificados correctamente de 0.88. En la figura 1.7 se muestra la matriz de confunsión.

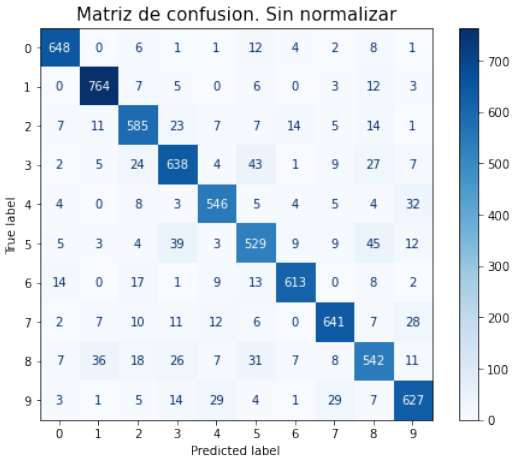

Figura 1.7: Matriz de confunsión-Arboles de clasificación.

En la figura 1.8 se ilustra el gráfico de barras, los resultados no fueron buenos como para los anteriores métodos.

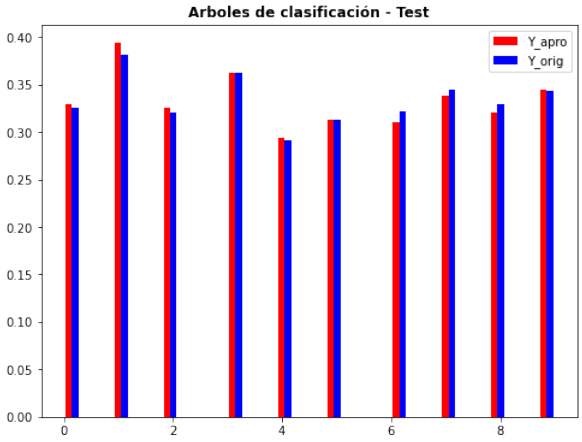

Figura 1.8: Arboles de clasificación con datos de prueba.

Finalmente en la siguiente tabla observamos que nuestros valores para las distintas métricas. Los valores que se observan en las métricas no fueron tan buenos para cada digito.

	precision	recall	f1-score	support
0	0.94	0.95	0.94	683
1	0.92	0.95	0.94	800
2	0.86	0.87	0.86	674
3	0.84	0.84	0.84	760
4	0.88	0.89	0.89	611
5	0.81	0.80	0.81	658
6	0.94	0.91	0.92	677
7	0.90	0.89	0.89	724
8	0.80	0.78	0.79	693
9	0.87	0.87	0.87	720
accuracy			0.88	7000

macro avg	0.88	0.88	0.88	7000
weighted avg	0.88	0.88	0.88	7000

AdaBoost

Al buscar los mejores parámetros para nuestro modelo encontramos los siguientes: algorimo: SAMME.R y DecisionTreeClassifier(max_depth=5). El resultado de ocupar dicho parámetros nos dio una proporción de datos clasificados correctamente de 0.84. En la figura 1.9 se muestra la matriz de confunsión.

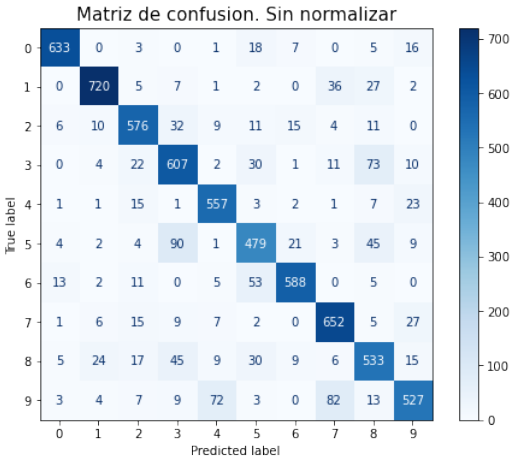

Figura 1.9: Matriz de confunsión-AdaBoost.

En la figura 1.10 se ilustra el gráfico de barras, los resultados tampoco fueron buenos como para los anteriores métodos.

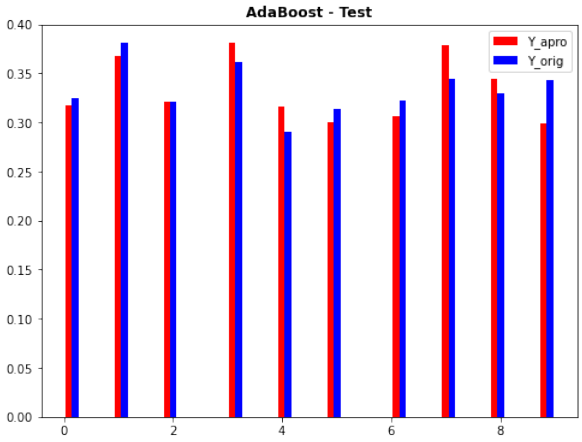

Figura 1.10: AdaBoost con datos de prueba.

En la siguiente tabla observamos que nuestros valores para las distintas métricas. Los valores que se observan en las métricas no fueron tan buenos para cada digito.

	precision	recall	f1-score	support
0	0.95	0.93	0.94	683
1	0.93	0.90	0.92	800
2	0.85	0.85	0.85	674
3	0.76	0.80	0.78	760
4	0.84	0.91	0.87	611
5	0.76	0.73	0.74	658
6	0.91	0.87	0.89	677
7	0.82	0.90	0.86	724
8	0.74	0.77	0.75	693
9	0.84	0.73	0.78	720
accuracy			0.84	7000

macro avg	0.84	0.84	0.84	7000
weighted avg	0.84	0.84	0.84	7000

Conclusiones

Al realizar cada uno de los métodos que nos solicitaron encontramos que la mejor proporción de datos clasificados correctamente fue de 0.92 con el método de Maquinas de soporte vectorial. Para cada método se agrego la matriz de confunsión, las metricas a evaluar y un gráfico ilustrativo sobre los resultados finales. En el caso de Maquinas de Soporte Vectorial el grafico fue de mucha ayuda visual ya que las barras de color rojo y azul se asemejan mucho comparado con los otros métodos. Al menos para este conjunto de datos que previamente habiamos analizado, el mejor método que encontramos fue de Maquinas de Soporte Vectorial.

2. Problema 2

Para el siguiente ejercicio partimos de los datos que previamente habiamos trabajado, recordando tomamos cerca de 4,000 palabras para hacer nuestro analisis anterior y en la figura 2.1 mostramos que con esa cantidad tenemos poco más del 80 %.

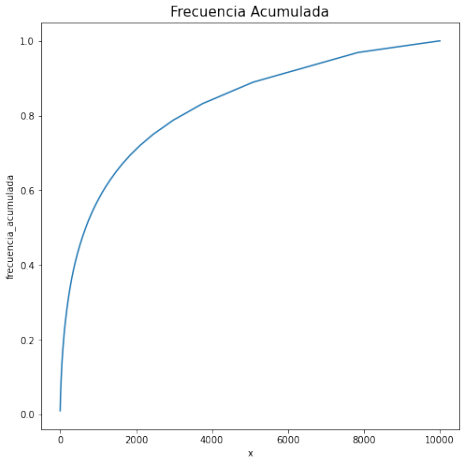

Figura 2.1: Gráfico de frecuencia acumulada de las letras.

Como resultado de realizar el proceso y tomar en cuenta aquellas palabras que se repiten un número considerado de veces tenemos la siguiente tabla 2.1, solo con la intención de ilustrar esta representación se muestra la vectorización que se le realizo a cada texto.

Para empezar nuestro ejercicio tomados nuevamente dos grupos de datos, un grupo de entreamiento y uno de prueba. Para el primer grupo tomamos el 90 % de los datos y el resto lo dejamos para el grupo de prueba. Para realizar el siguiente ejercicio se repitio el procedimiento que en el ejercicio 1 solo que en este caso vamos a tomar dos casos para cada método, en uno tomamos en cuenta la categoria de los textos y en otro el sentimiento.

Redes Neuronales

Para este primer método tomando en cuenta la categoria, encontramos los mejores parámetros para el modelo, los cuales son los siguientes: número de capas = [10] y alpha = 0.05. Dandonos una proporción de datos clasificados

	si	pelicula	solo	asi	bien	..	legal
0	3	0	0	1	0	...	0
1	8	0	0	6	4	...	0
2	0	1	0	0	2	...	0
...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...
398	0	0	0	0	1	...	0
399	0	1	0	0	0	...	0

Cuadro 2.1: Fruta disponible

correctamente de 0.91.
En la figura 2.2 se muestra la matriz de confunsión, se puede observar que muchos de los valores encontrados coinciden con los originales.

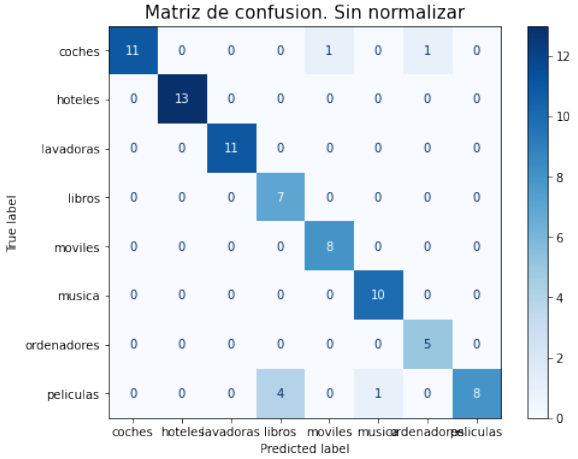

Figura 2.2: Matriz de confunsión-Redes Neuronales(Categoria).

En la figura 2.3 se ilustra el gráfico de barras, los resultados se observan

marcados pero es debido a que los datos que estamos probando son solo 80 por lo que hace notar que faltan muchos.

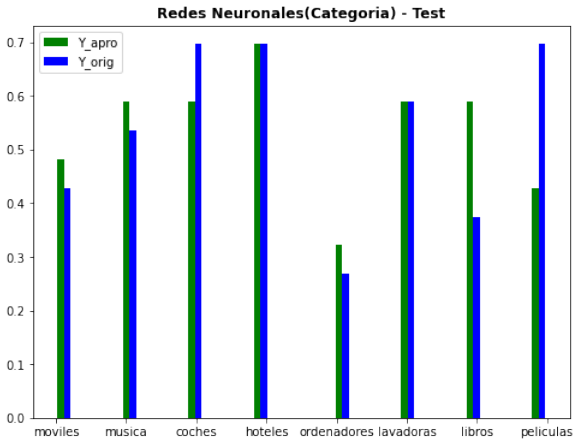

Figura 2.3: Redes Neuronales con datos de prueba(Categoria).

En la siguiente tabla observamos que nuestros valores para las distintas métricas. En algunos resultados vemos que nuestro modelo clasifico muy bien los de una categoria y en otros fallo.

	precision	recall	f1-score	support
coches	1.00	0.85	0.92	13
hoteles	1.00	1.00	1.00	13
lavadoras	1.00	1.00	1.00	11
libros	0.64	1.00	0.78	7
moviles	0.89	1.00	0.94	8
musica	0.91	1.00	0.95	10
ordenadores	0.83	1.00	0.91	5
peliculas	1.00	0.62	0.76	13

```
accuracy                           0.91        80
macro avg      0.91      0.93      0.91        80
weighted avg   0.94      0.91      0.91        80
```

Ahora tomando el sentimiento encontramos que los parámetros para el modelo, los cuales son los siguientes: número de capas = [10] y alpha = 0.01. Dandonos una proporción de datos clasificados correctamente de 0.64.
En la figura 2.4 se muestra la matriz de confunsión, se puede observar que muchos de los valores encontrados coinciden con los originales.

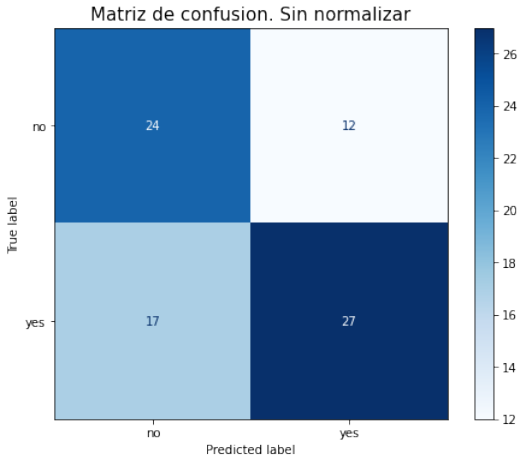

Figura 2.4: Matriz de confunsión-Redes Neuronales(Sentimiento).

En la figura 2.5 se ilustra el gráfico de barras. Se observa algo pegados pero el porcentaje antes mencionado nos dice que los resultados si son cambiaron mucho.

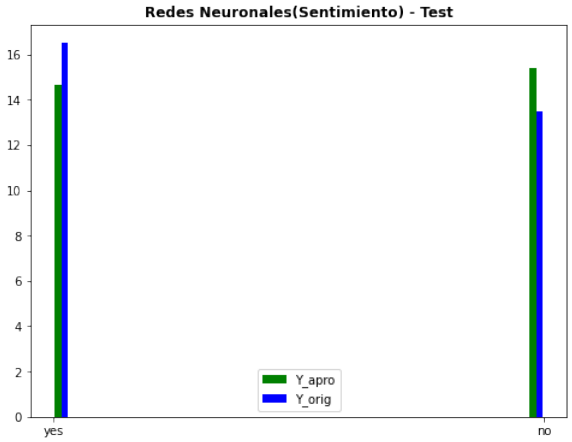

Figura 2.5: Redes Neuronales con datos de prueba(Sentimiento).

En la siguiente tabla observamos que nuestros valores para las distintas métricas. En algunos resultados vemos que nuestro modelo clasifico muy bien los de una categoria y en otros fallo.

	precision	recall	f1-score	support
no	0.59	0.67	0.62	36
yes	0.69	0.61	0.65	44
accuracy			0.64	80
macro av	0.64	0.64	0.64	80
weighted avg	0.64	0.64	0.64	80

Maquinas de Soporte Vectorial

Al buscar los mejores parámetros para nuestro modelo encontramos que el kernel adecuado es linear y un valor de C = 2.5. El resultado de ocupar dicho parámetros nos dio una proporción de datos clasificados correcta-

mente de 0.90. En la figura 2.6 se muestra la matriz de confunsión.

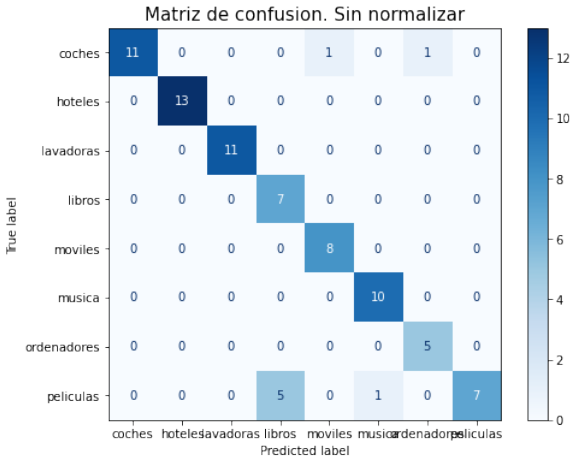

Figura 2.6: Matriz de confunsión-Maquinas de Soporte Vectorial(Categoria).

En la figura 2.7 se ilustra el gráfico de barras. En algunas categorias se observa que fallaron los resultados y en otros si los identifico muy bien.

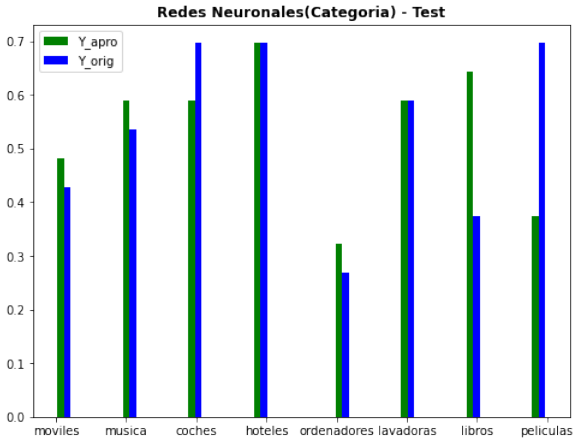

Figura 2.7: Maquinas de Soporte Vectorial con datos de prueba(Categoria).

En la siguiente tabla observamos que nuestros valores para las distintas métricas. En algunos resultados vemos que nuestro modelo clasifico muy bien los de una categoria y en otros fallo.

	precision	recall	f1-score	support
coches	1.00	0.85	0.92	13
hoteles	1.00	1.00	1.00	13
lavadoras	1.00	1.00	1.00	11
libros	0.58	1.00	0.74	7
moviles	0.89	1.00	0.94	8
musica	0.91	1.00	0.95	10
ordenadores	0.83	1.00	0.91	5
peliculas	1.00	0.54	0.70	13
accuracy			0.90	80
macro avg	0.90	0.92	0.89	80

```
weighted avg    0.93      0.90      0.90         80
```

Ahora buscacamos los parámetros nuevamente para el sentimiento los mejores parámetros para nuestro modelo encontramos que el kernel adecuado es gaussiano y un valor de C = 2.5. El resultado de ocupar dicho parámetros nos dio una proporción de datos clasificados correctamente de 0.69. En la figura 2.8 se muestra la matriz de confunsión.

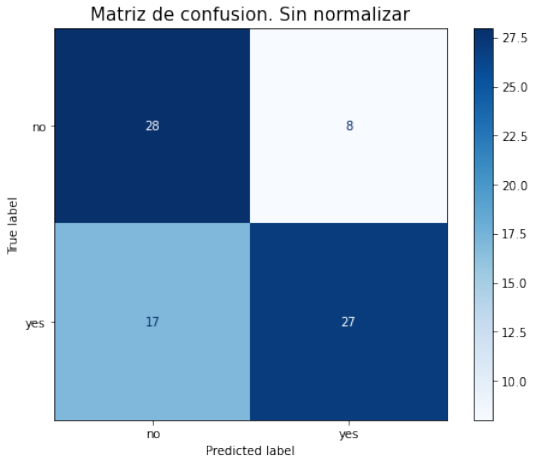

Figura 2.8: Matriz de confunsión-Maquinas de Soporte Vectorial(Sentimiento).

En la figura 2.9 se ilustra el gráfico de barras. Se observa algo pegados pero el porcentaje antes mencionado nos dice que los resultados si son cambiaron mucho.

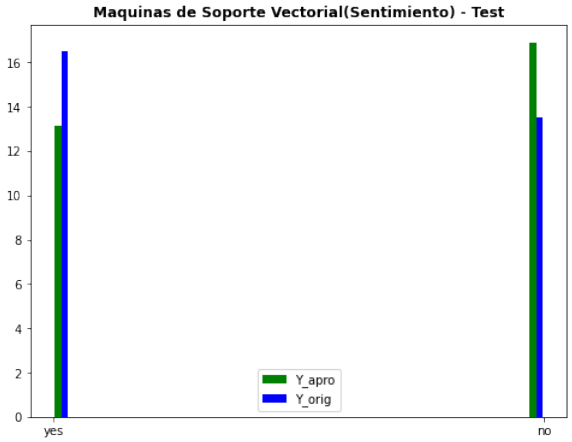

Figura 2.9: Maquinas de Soporte Vectorial con datos de prueba(Sentimiento).

Finalmente mostramos el resultado de las metricas que ocupamos en la siguiente tabla.

	precision	recall	f1-score	support
no	0.62	0.78	0.69	36
yes	0.77	0.61	0.68	44
accuracy			0.69	80
macro avg	0.70	0.70	0.69	80
weighted avg	0.70	0.69	0.69	80

Arboles de clasificación

Al buscar los mejores parámetros para nuestro modelo encontramos que la profundidad máxima, elementos mínimos en nodos y un alpha de $1 * 10^{-5}$. El resultado de ocupar dicho parámetros nos dio una proporción de datos

clasificados correctamente de 0.82. En la figura 2.10 se muestra la matriz de confunsión.

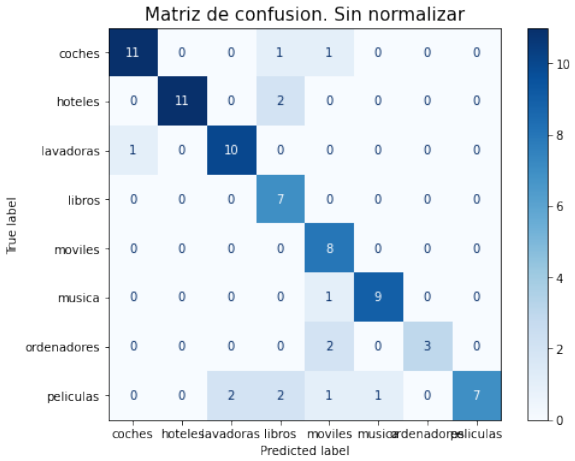

Figura 2.10: Matriz de confunsión-Arboles de clasificación(Categoria).

En la figura 2.11 se ilustra el gráfico de barras. Se observa algo pegados pero el porcentaje antes mencionado nos dice que los resultados si son cambiaron mucho.

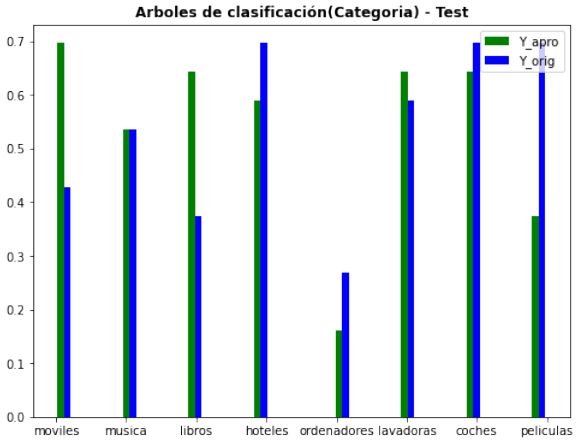

Figura 2.11: Arboles de clasificación con datos de prueba(Categoria).

Finalmente mostramos el resultado de las metricas que ocupamos en la siguiente tabla.

	precision	recall	f1-score	support
coches	0.92	0.85	0.88	13
hoteles	1.00	0.85	0.92	13
lavadoras	0.83	0.91	0.87	11
libros	0.58	1.00	0.74	7
moviles	0.62	1.00	0.76	8
musica	0.90	0.90	0.90	10
ordenadores	1.00	0.60	0.75	5
peliculas	1.00	0.54	0.70	13
accuracy			0.82	80
macro avg	0.86	0.83	0.81	80
weighted avg	0.88	0.82	0.83	80

Ahora para el sentimiento los mejores parámetros para nuestro modelo

encontramos que la profundidad máxima, elementos mínimos en nodos y un alpha de $1 * 10^{-5}$. El resultado de ocupar dicho parámetros nos dio una proporción de datos clasificados correctamente de 0.56. En la figura 2.12 se muestra la matriz de confunsión.

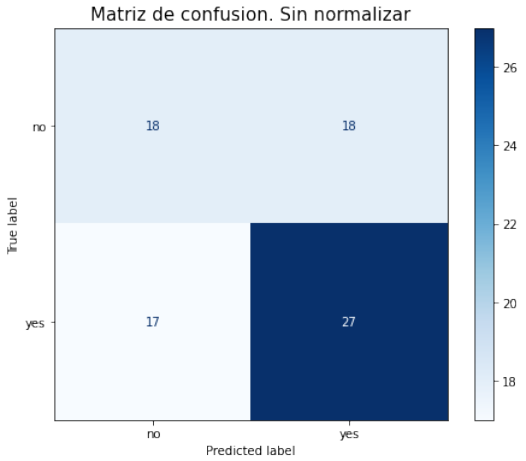

Figura 2.12: Matriz de confunsión-Arboles de clasificación(Sentimiento).

En la figura 2.13 se ilustra el gráfico de barras. Se observa algo pegados pero el porcentaje antes mencionado nos dice que los resultados si son cambiaron mucho.

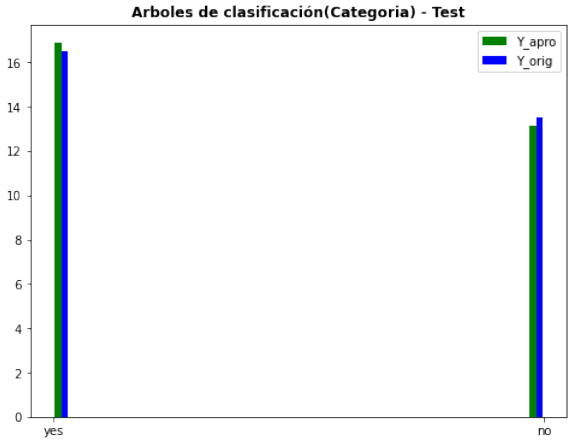

Figura 2.13: Arboles de clasificación con datos de prueba(Sentimiento).

También mostramos las metricas que resultaron de tomar en cuenta el sentimiento en la siguiente tabla.

	precision	recall	f1-score	support
no	0.51	0.50	0.51	36
yes	0.60	0.61	0.61	44
accuracy			0.56	80
macro avg	0.56	0.56	0.56	80
weighted avg	0.56	0.56	0.56	80

AdaBoost

Al buscar los mejores parámetros para nuestro modelo encontramos los siguientes: algorimo: SAMME.R y DecisionTreeClassifier(max_depth=5). El resultado de ocupar dicho parámetros nos dio una proporción de datos clasificados correctamente de 0.88. En la figura 2.14 se muestra la matriz

de confunsión.

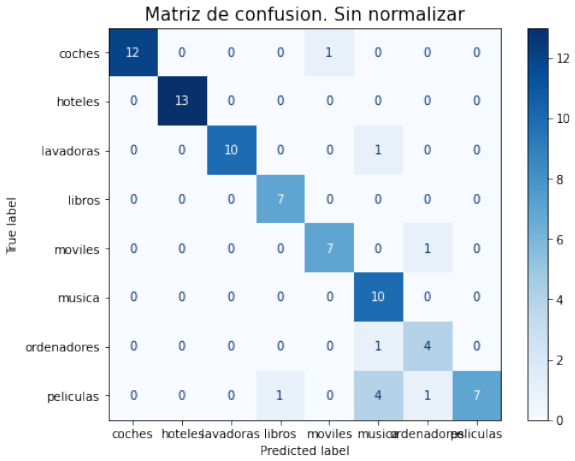

Figura 2.14: Matriz de confunsión-AdaBoost(Categoria).

En la figura 2.15 se ilustra el gráfico de barras, los resultados tampoco fueron buenos como para los anteriores métodos.

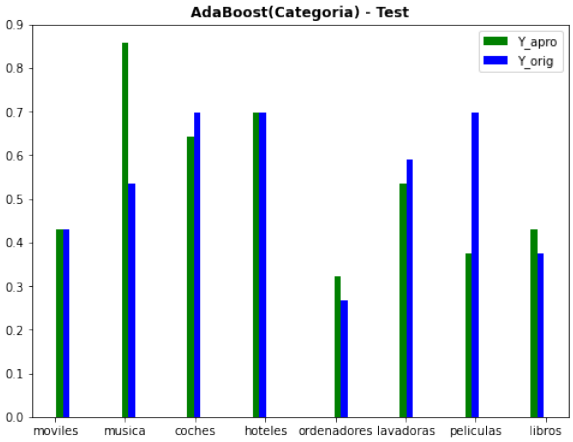

Figura 2.15: AdaBoost con datos de prueba(Categoria).

Agregamos en las siguiente tabla las metricas que ocupamos para evaluar que tan bueno fueron nuestros resultados con el método de AdaBoost.

	precision	recall	f1-score	support
coches	1.00	0.92	0.96	13
hoteles	1.00	1.00	1.00	13
lavadoras	1.00	0.91	0.95	11
libros	0.88	1.00	0.93	7
moviles	0.88	0.88	0.88	8
musica	0.62	1.00	0.77	10
ordenadores	0.67	0.80	0.73	5
peliculas	1.00	0.54	0.70	13
accuracy			0.88	80
macro avg	0.88	0.88	0.86	80
weighted avg	0.91	0.88	0.87	80

Ahora para el sentimiento los mejores parámetros para nuestro mode-

lo encontramos los siguientes: algorimo: SAMME.R y DecisionTreeClassifier(max_depth=5). El resultado de ocupar dicho parámetros nos dio una proporción de datos clasificados correctamente de 0.65. En la figura 2.16 se muestra la matriz de confunsión.

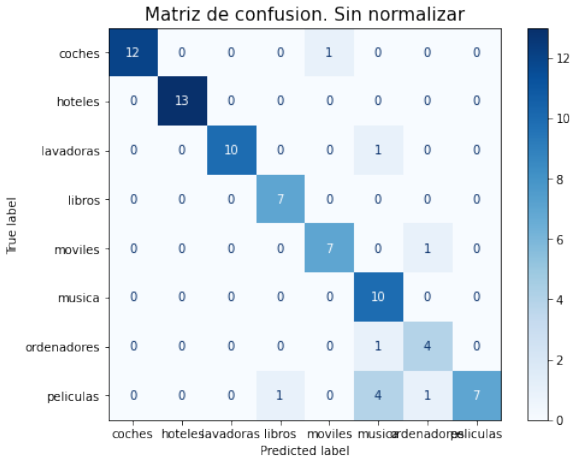

Figura 2.16: Matriz de confunsión-AdaBoost(Sentimiento).

En la figura 2.17 se ilustra el gráfico de barras, los resultados tampoco fueron buenos como para los anteriores métodos.

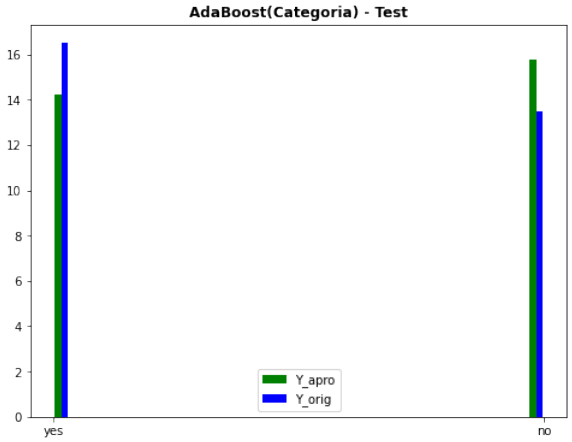

Figura 2.17: AdaBoost con datos de prueba(Sentimiento).

Volvemos a mostrar las metricas que nos van a servir para tomar en cuenta que método fue el mejor para para el sentimiento.

	precision	recall	f1-score	support
no	0.60	0.69	0.64	36
yes	0.71	0.61	0.66	44
accuracy			0.65	80
macro avg	0.65	0.65	0.65	80
weighted avg	0.66	0.65	0.65	80

Conclusiones

Al observar cada uno de los métodos con los datos de los textos observamos que en muchos de ellos en la proporción de datos clasificados correctamente en el sentimiento que trae cada texto no fue superior al 0.70, afortunadamente no sucedio lo mismo para la categoria ya que la mejor proporción

fue de 0.91 lo que nos da muy buen resultado.
Tomando en cuenta lo anterior el método que mejor nos ayudo fue Redes Neuronales ya que nos dio una proporción igual a 0.91, se pensaria que igual fue el mejor para identificar el sentimiento pero no lo fue. En el caso del sentimiento encontramos que el método que mejor nos ayudo fue Maquinas de soporte Vectorial ya que nos dio una proporción igual a 0.69, no es muy alta pero al menos rebasa el 0.50 y comparado con los demás es el mejor. Así para poder encontrar la categoria escogemos el método de Redes Neuronales y para el sentimiento Maquinas de Soporte Vectorial.