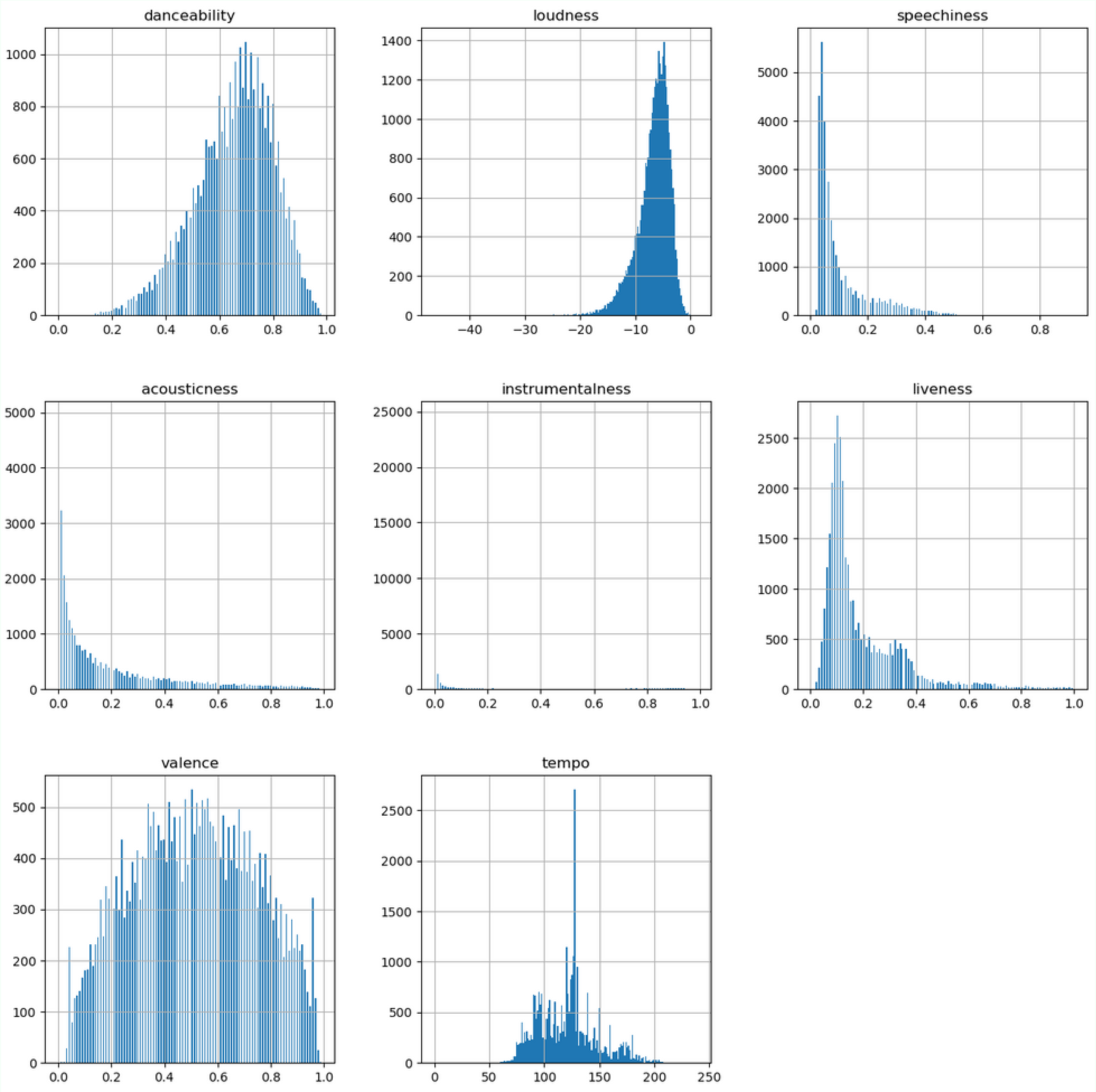
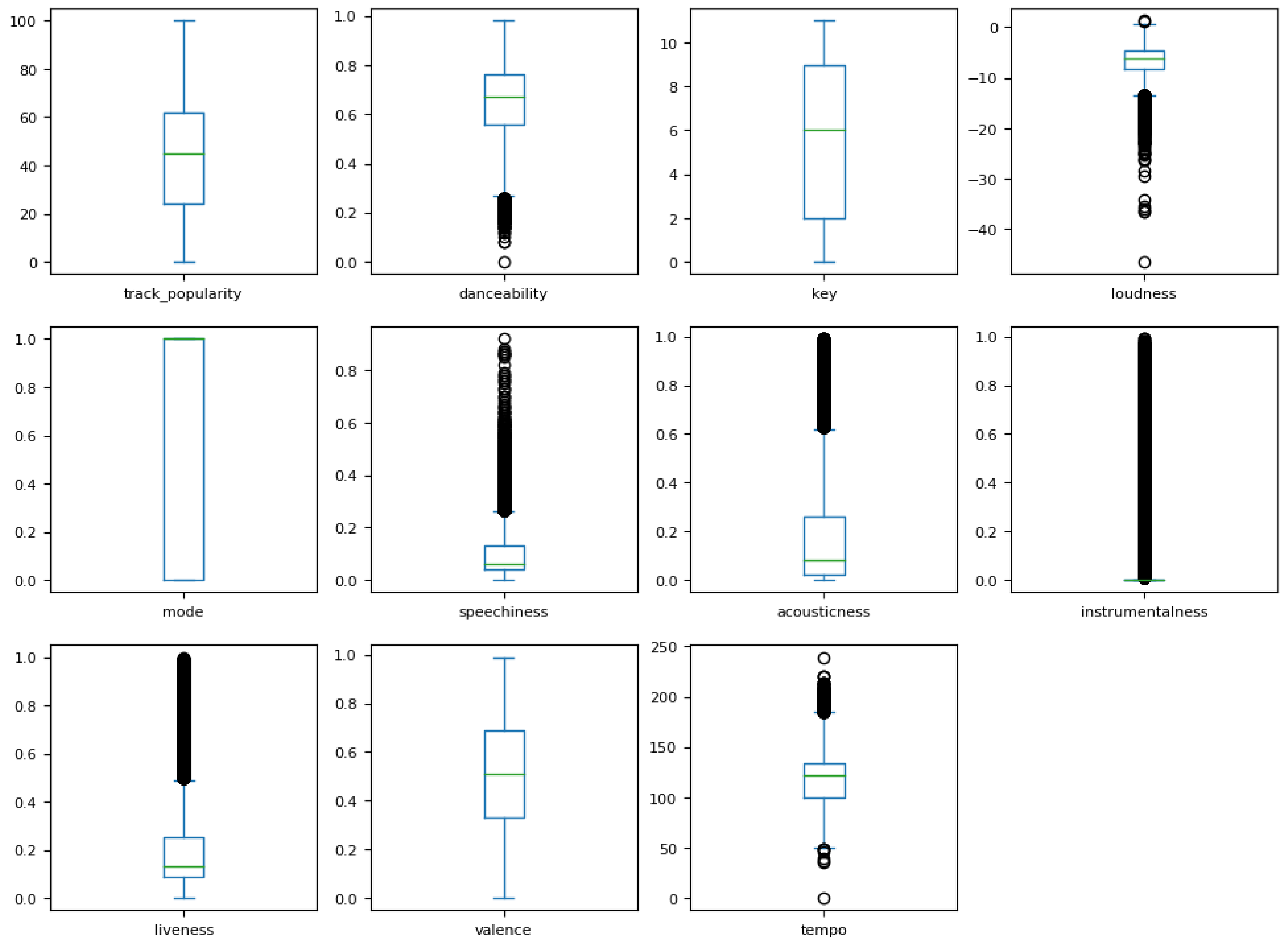
Hit PREDICTOR

Binary classification on a Spotify Dataset

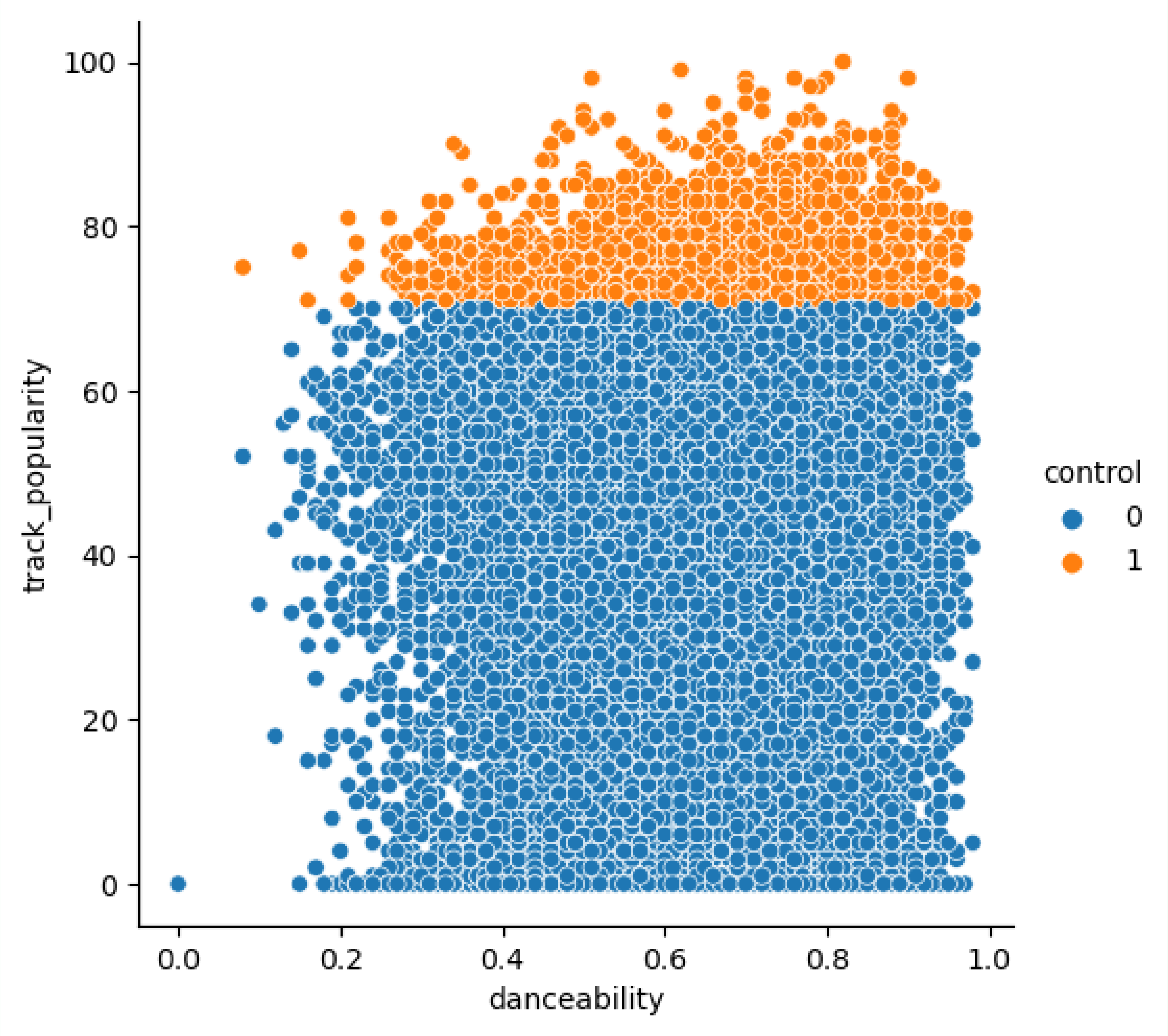
Texto

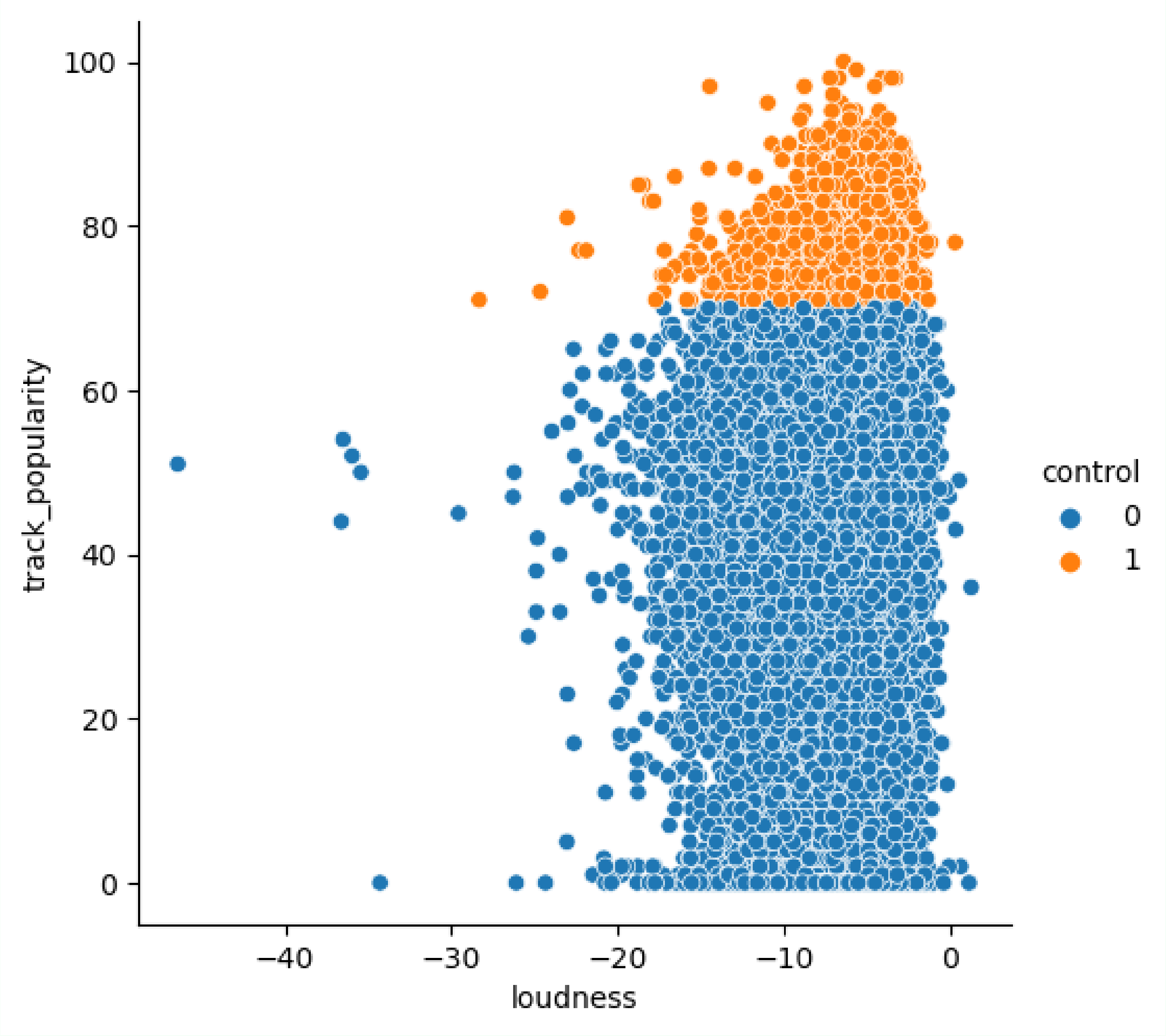
columns_numericas = df.select_dtypes(include=['float64'])

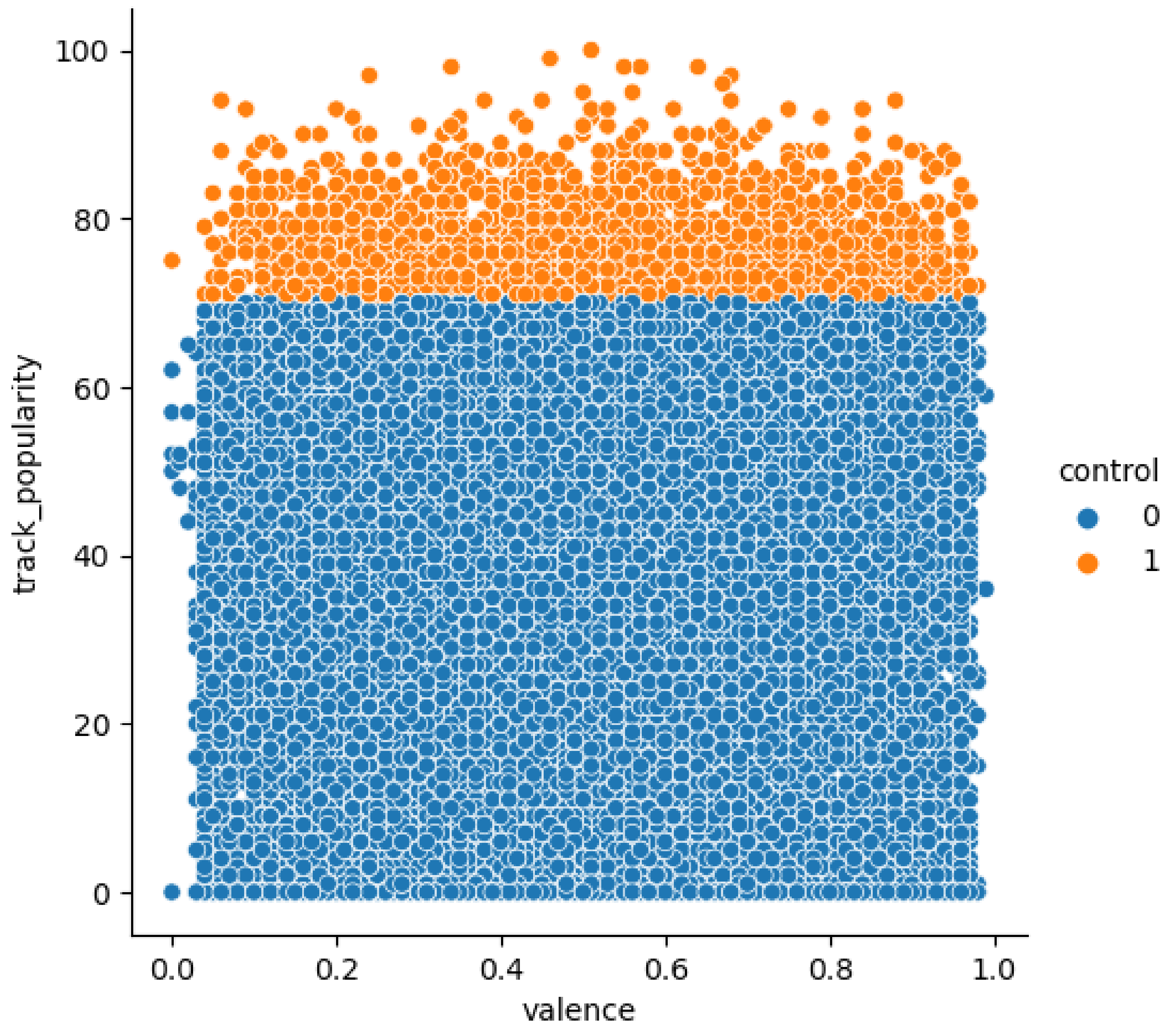


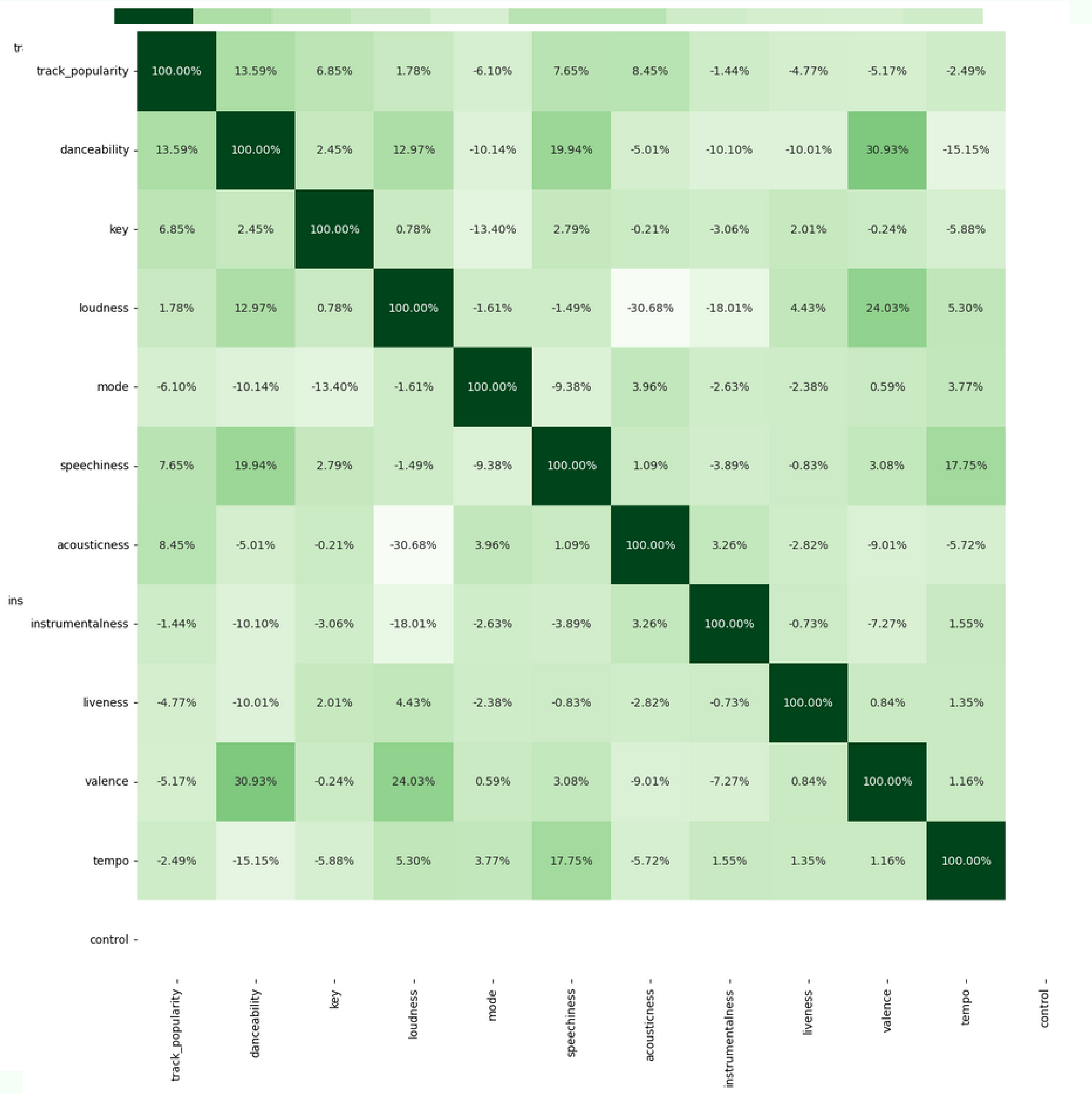


track_popularity
0
20
40
60
80
100
danceability
0.0
0.2
0.4
0.6
0.8
1.0
key
0
2
4
6
8
10
loudness
0
−10
−20
−30
−40
mode
0.0
0.2
0.4
0.6
0.8
1.0
speechiness
0.0
0.2
0.4
0.6
0.8
acousticness
0.0
0.2
0.4
0.6
0.8
1.0
instrumentalness
0.0
0.2
0.4
0.6
0.8
1.0
liveness
0.0
0.2
0.4
0.6
0.8
1.0
valence
0.0
0.2
0.4
0.6
0.8
1.0
tempo
0
50
100
150
200
250


track_popularity
100
80
60
40
20
0
0.0
0.2
0.4
0.6
0.8
1.0
danceability
control
● 0
● 1


100
80
60
40
20
0
track_popularity
−40
−30
−20
−10
0
loudness
control
● 0
● 1


track_popularity
100
80
60
40
20
0
0.0
0.2
0.4
0.6
0.8
1.0
valence
control
● 0
● 1



	track_popularity	danceability	key	loudness	mode	speechiness	acousticness	instrumentalness	liveness	valence	tempo	control
track_popularity	100.00%	13.59%	6.85%	1.78%	-6.10%	7.65%	8.45%	-1.44%	-4.77%	-5.17%	-2.49%	
danceability	13.59%	100.00%	2.45%	12.97%	-10.14%	19.94%	-5.01%	-10.10%	-10.01%	30.93%	-15.15%	
key	6.85%	2.45%	100.00%	0.78%	-13.40%	2.79%	-0.21%	-3.06%	2.01%	-0.24%	-5.88%	
loudness	1.78%	12.97%	0.78%	100.00%	-1.61%	-1.49%	-30.68%	-18.01%	4.43%	24.03%	5.30%	
mode	-6.10%	-10.14%	-13.40%	-1.61%	100.00%	-9.38%	3.96%	-2.63%	-2.38%	0.59%	3.77%	
speechiness	7.65%	19.94%	2.79%	-1.49%	-9.38%	100.00%	1.09%	-3.89%	-0.83%	3.08%	17.75%	
acousticness	8.45%	-5.01%	-0.21%	-30.68%	3.96%	1.09%	100.00%	3.26%	-2.82%	-9.01%	-5.72%	
instrumentalness	-1.44%	-10.10%	-3.06%	-18.01%	-2.63%	-3.89%	3.26%	100.00%	-0.73%	-7.27%	1.55%	
liveness	-4.77%	-10.01%	2.01%	4.43%	-2.38%	-0.83%	-2.82%	-0.73%	100.00%	0.84%	1.35%	
valence	-5.17%	30.93%	-0.24%	24.03%	0.59%	3.08%	-9.01%	-7.27%	0.84%	100.00%	1.16%	
tempo	-2.49%	-15.15%	-5.88%	5.30%	3.77%	17.75%	-5.72%	1.55%	1.35%	1.16%	100.00%	
control												

No vemos grandes diferencias en cuanto a correlaciones x feature y grupo de control. Si acaso :

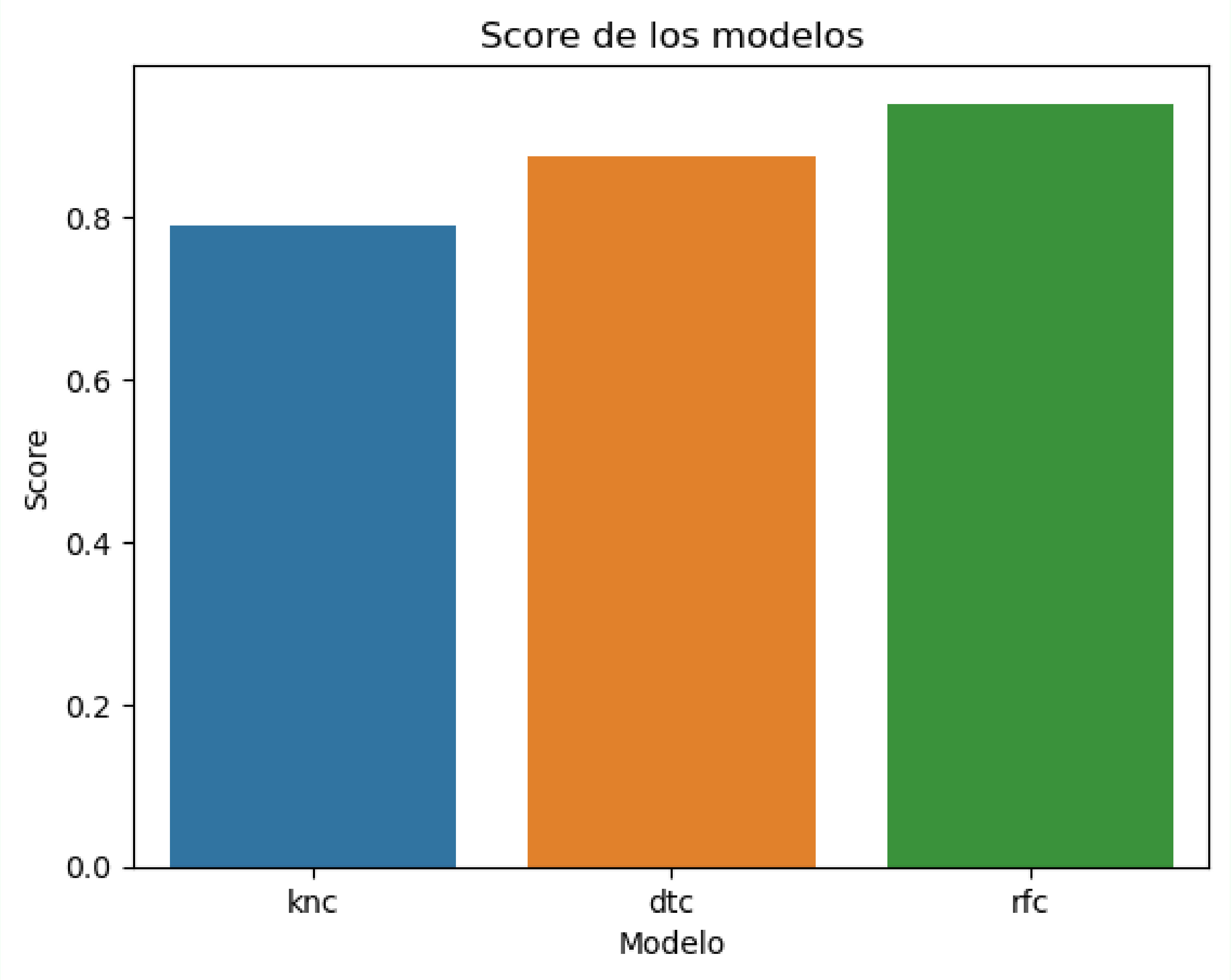
Danceability : 0.13 TOP vs 0.03 BOTTOM ...las canciones TOP suelen ser ligeramente más bailables.

Speechness : 0.07 TOP vs 0.004 BOTTOM ...las canciones TOP suelen incluir más partes "habladas", mas "Lyrics".

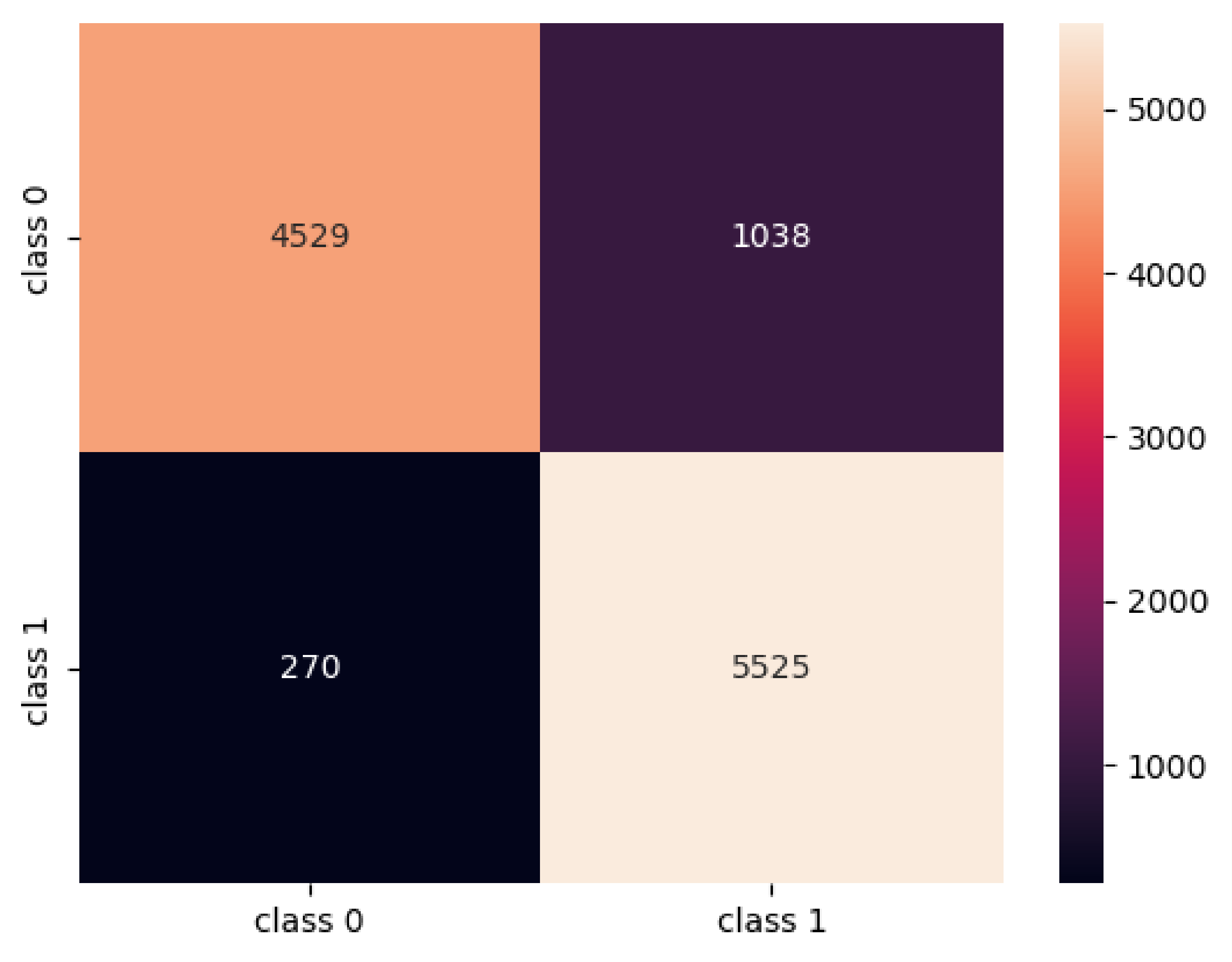
Instrumentalness : -0.014 TOP vs - 0.09 BOTTOM .. las canciones TOP son menos instrumentales que las canciones BOTTOM. Confirma en cierta forma la importancia de las lyrics en cuanto a la popularidad de una canción (que no calidad)

Valence : -0.051 TOP vs 0.02 BOTTOM ...las canciones TOP guardan una mínima correlación negativa con Valence y las canciones BOTTOM guardan una mínima correlación positiva con VALENCE. las canciones TOP suelen ser más alegres que las canciones BOTTOM.

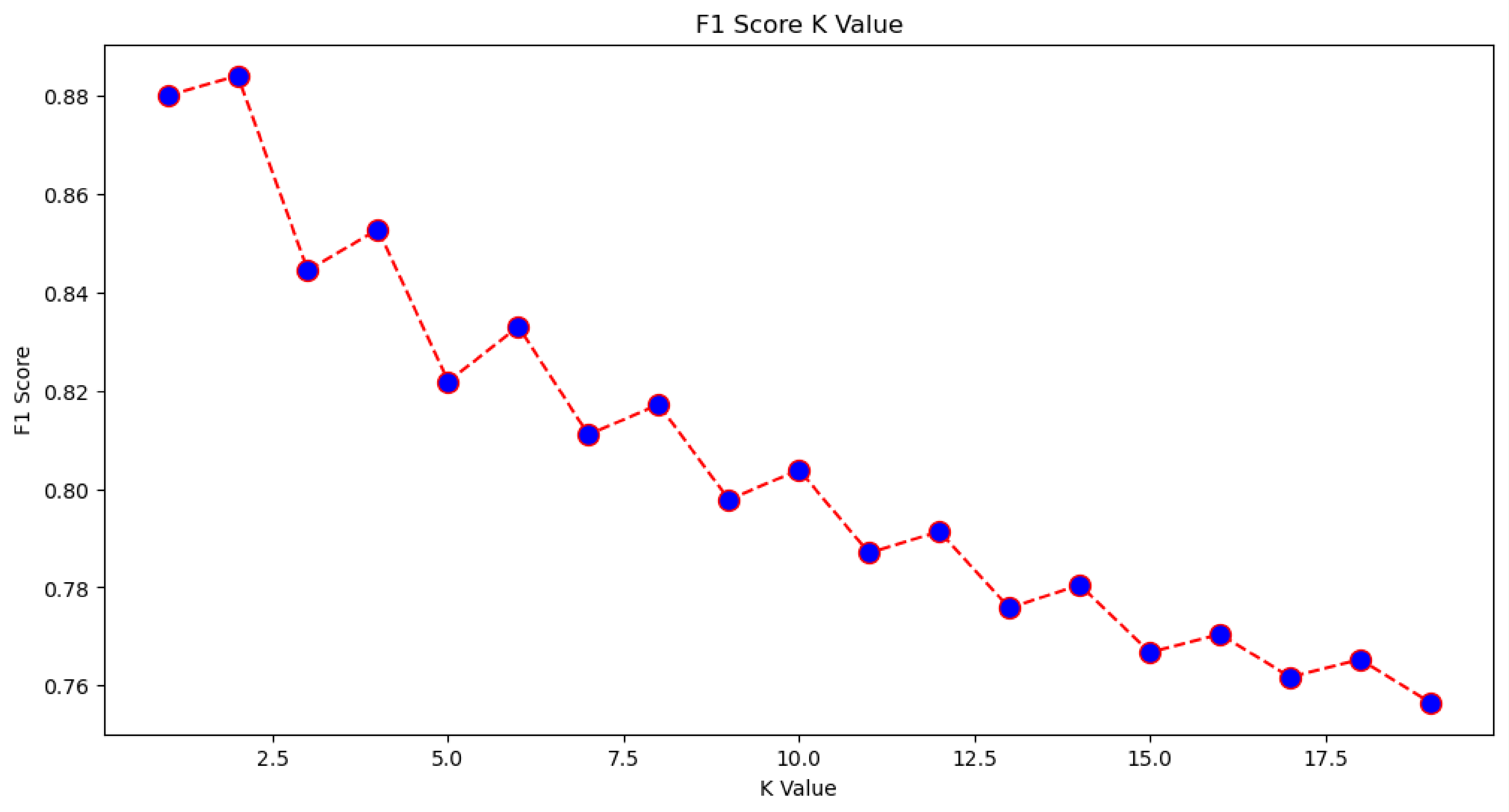
T



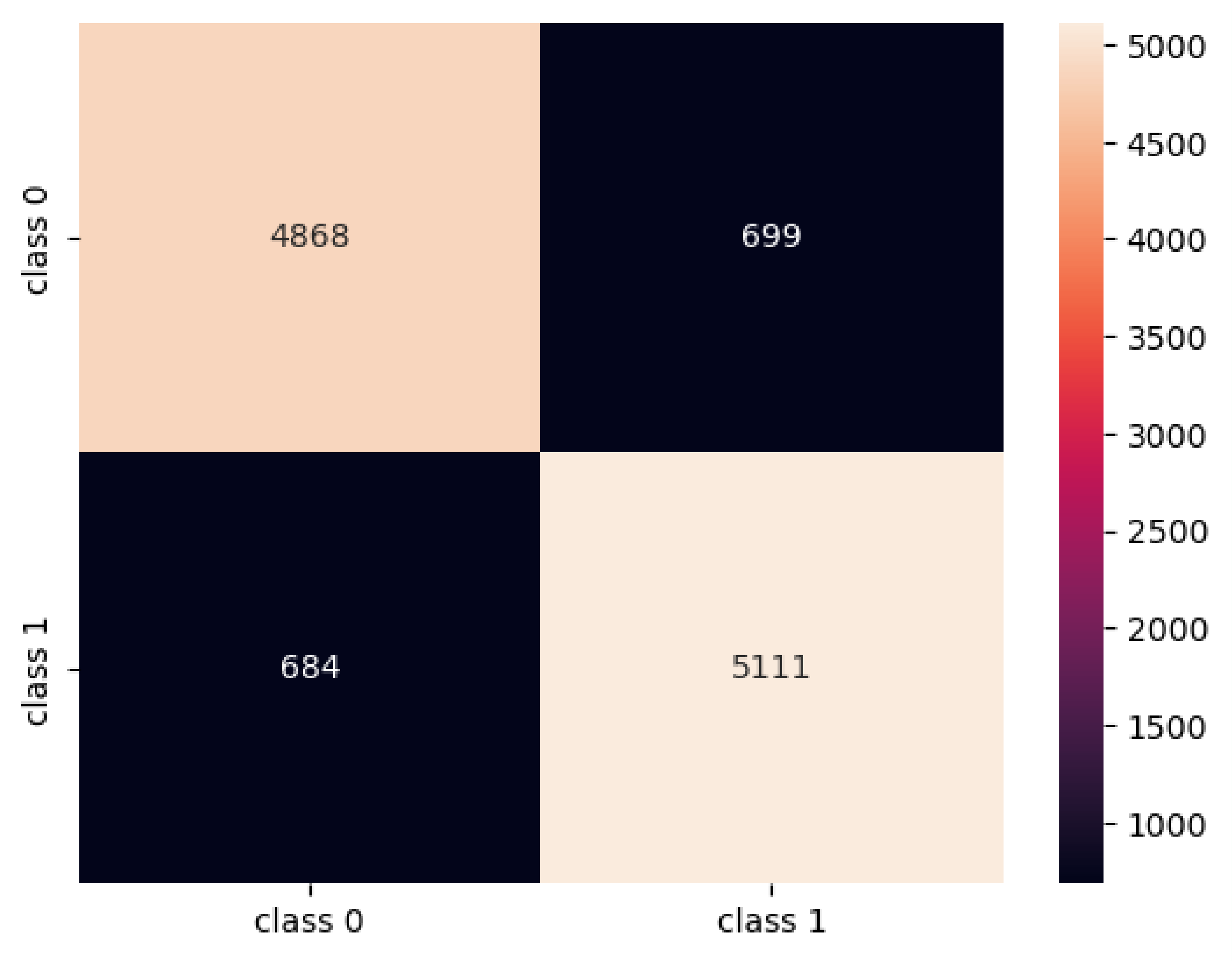
Scores analísis previo Modelos CrossValidation



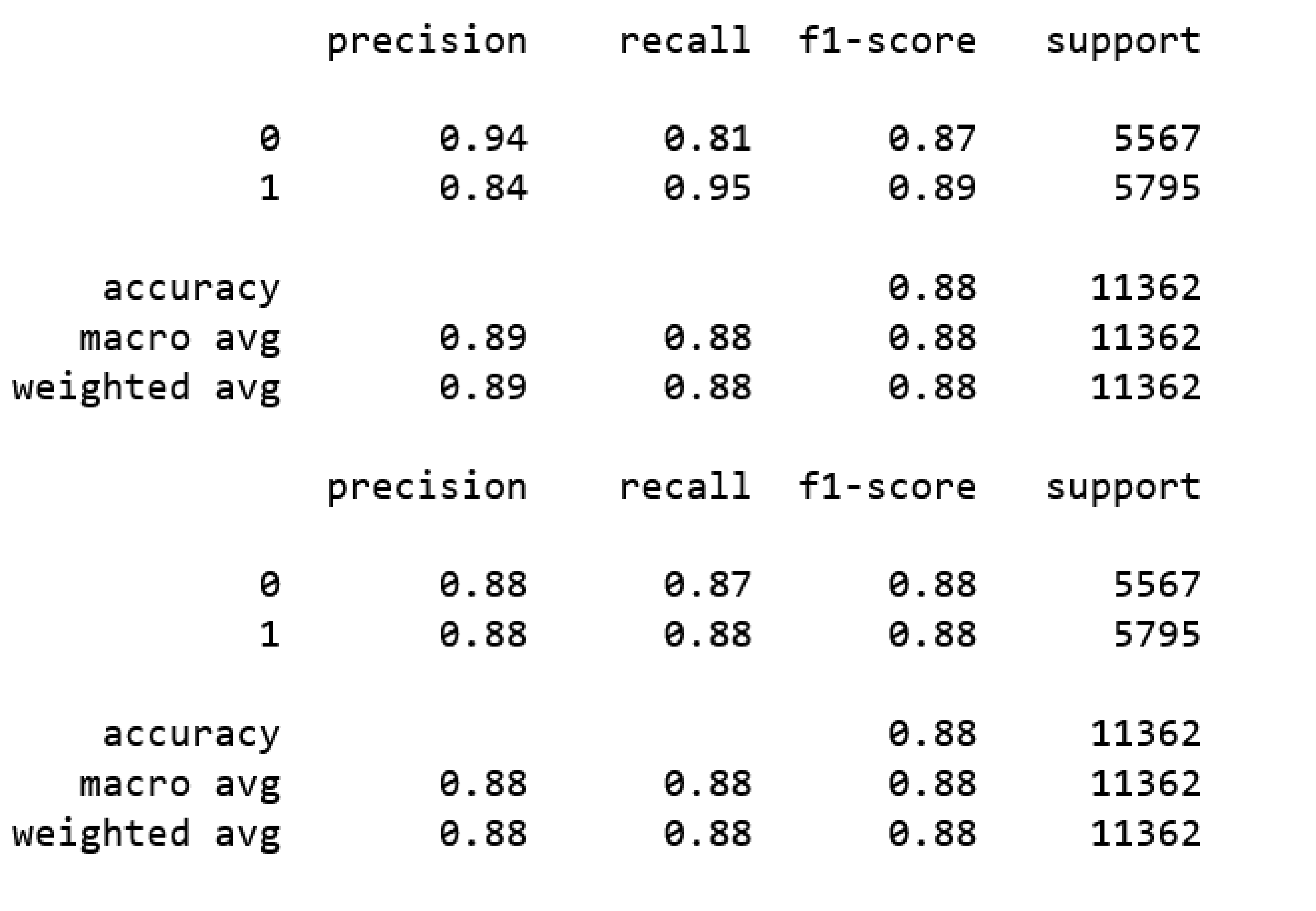
Heatmap Accuracy para Best Model



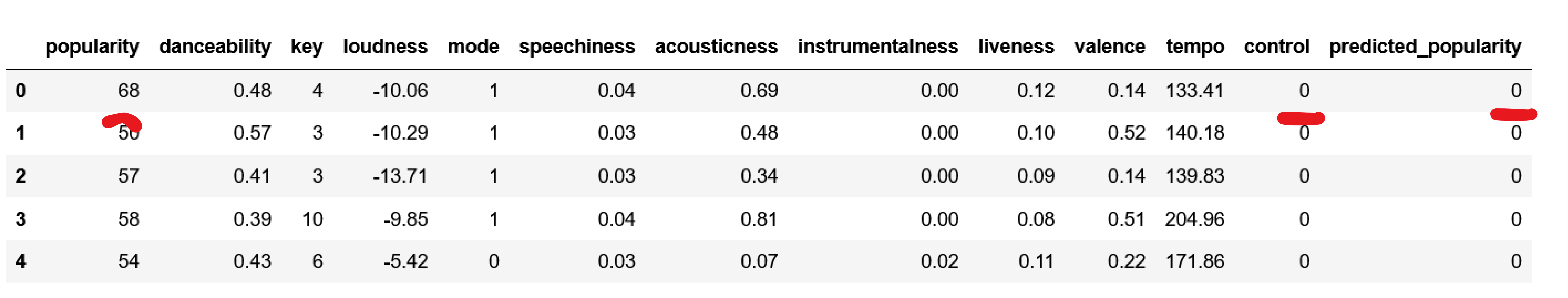
Método K.Elbow para k.óptimo = 2



Heatmap Accuracy para K = 2



```
              precision    recall  f1-score   support

           0       0.94      0.81      0.87      5567
           1       0.84      0.95      0.89      5795

    accuracy                           0.88     11362
   macro avg       0.89      0.88      0.88     11362
weighted avg       0.89      0.88      0.88     11362

              precision    recall  f1-score   support

           0       0.88      0.87      0.88      5567
           1       0.88      0.88      0.88      5795

    accuracy                           0.88     11362
   macro avg       0.88      0.88      0.88     11362
weighted avg       0.88      0.88      0.88     11362
```

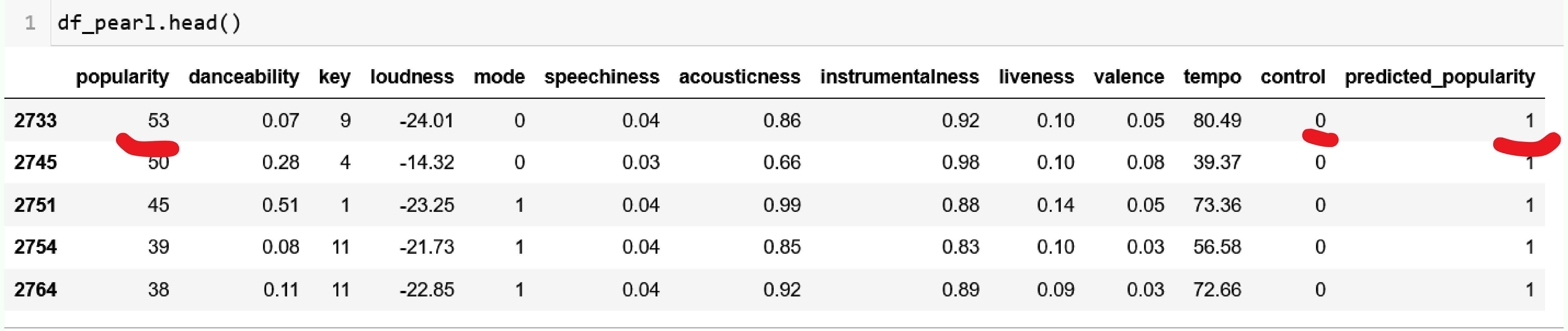


	popularity	danceability	key	loudness	mode	speechiness	acousticness	instrumentalness	liveness	valence	tempo	control	predicted_popularity
0	68	0.48	4	-10.06	1	0.04	0.69	0.00	0.12	0.14	133.41	0	0
1	50	0.57	3	-10.29	1	0.03	0.48	0.00	0.10	0.52	140.18	0	0
2	57	0.41	3	-13.71	1	0.03	0.34	0.00	0.09	0.14	139.83	0	0
3	58	0.39	10	-9.85	1	0.04	0.81	0.00	0.08	0.51	204.96	0	0
4	54	0.43	6	-5.42	0	0.03	0.07	0.02	0.11	0.22	171.86	0	0

```
df2['control'] = df2['popularity'].apply(lambda x: 1 if x > 70 else  0)
```

```
predicted_popularity = best_model_knn.predict(df3)
```

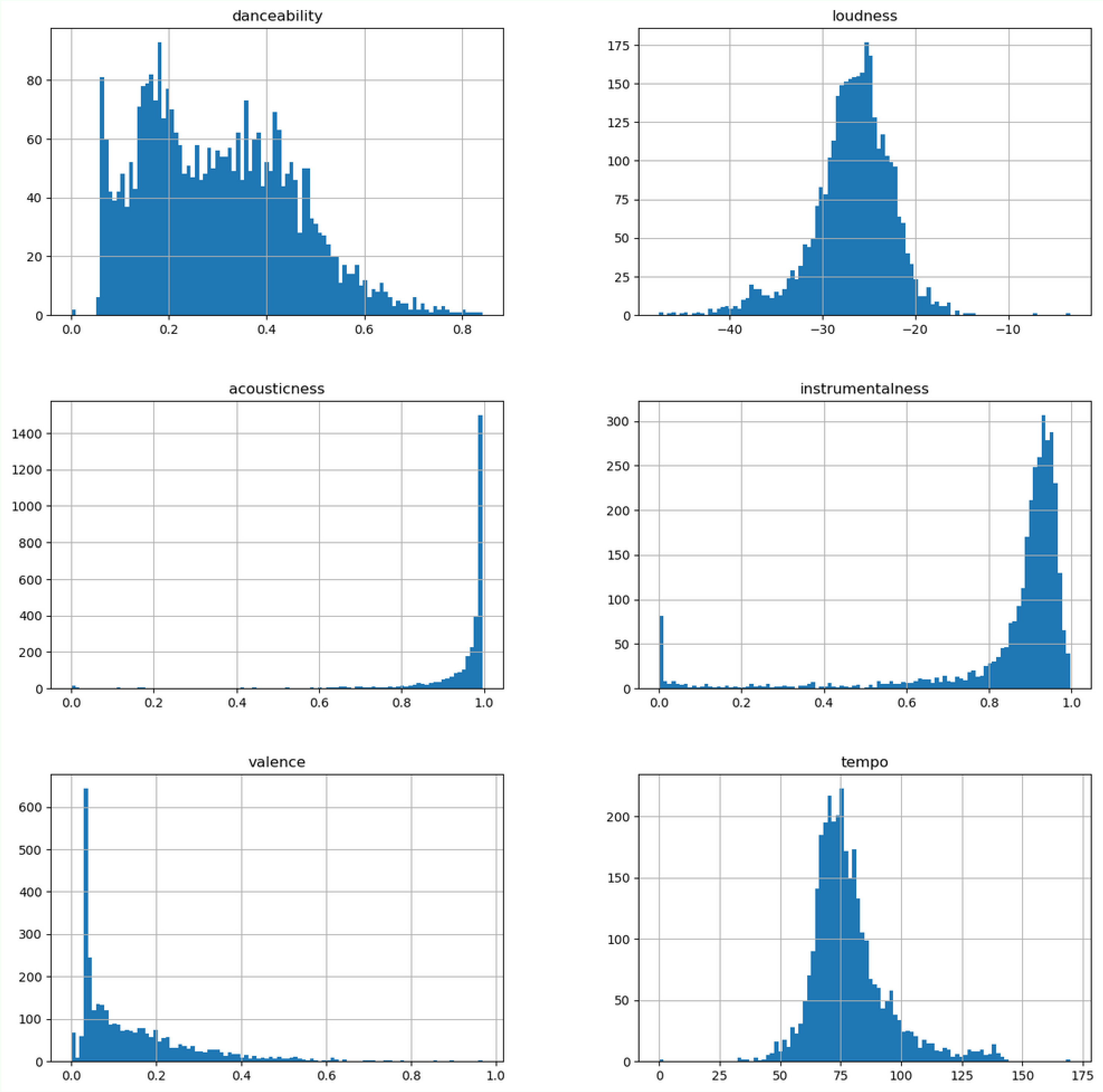
```
# Buscamos las "perlas", canciones que por características ( segun algoritmo de class por features sónicas) deberían ser populares ( predicted_popularity = 1) y que sin embargo no alcanzaron popularidad ( control = 0)

df_pearl = df2.query('control == 0 & predicted_popularity == 1 & 30 <popularity < 60')
```



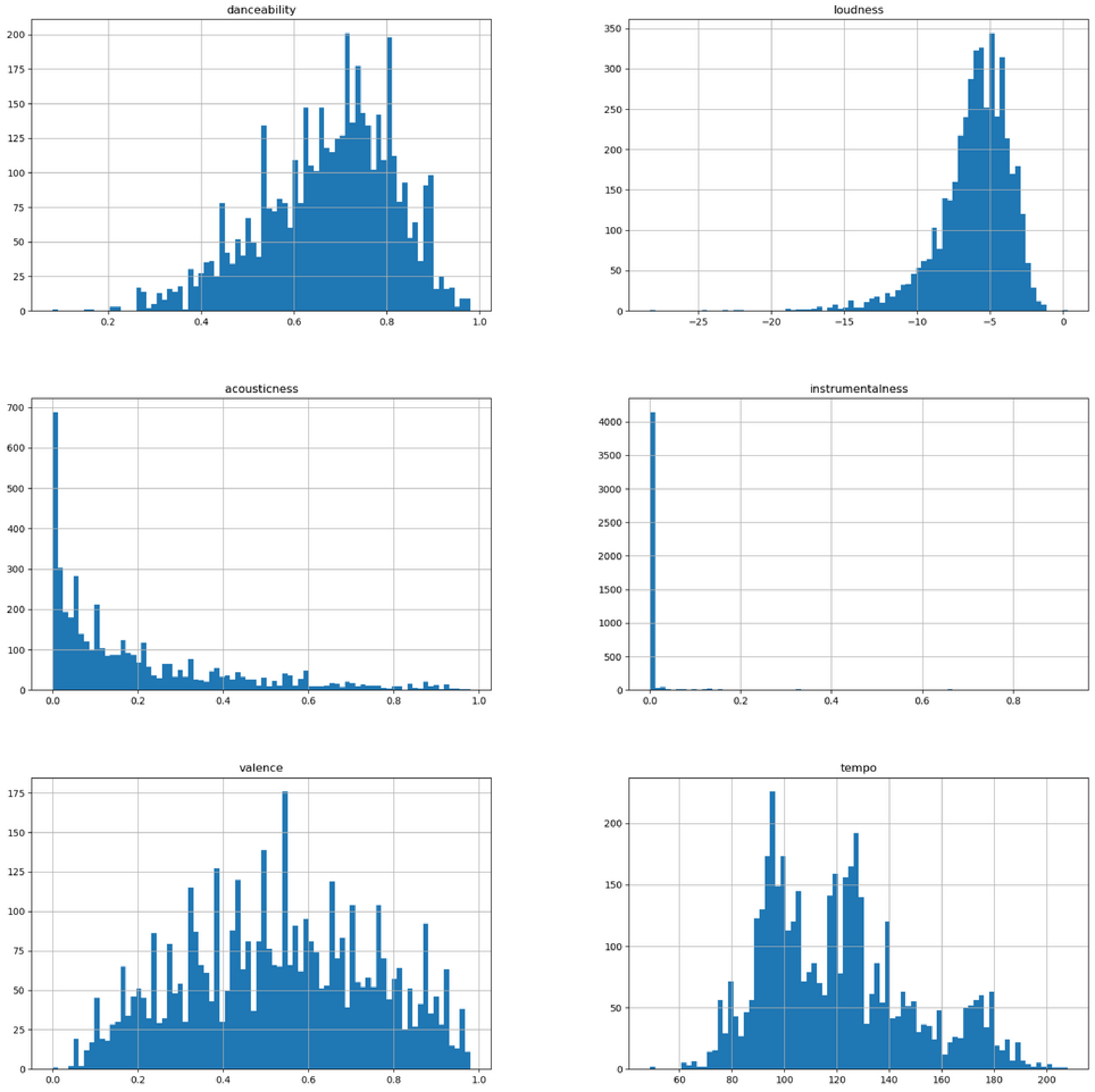
```
df_pearl.head()
```

	popularity	danceability	key	loudness	mode	speechiness	acousticness	instrumentalness	liveness	valence	tempo	control	predicted_popularity
2733	53	0.07	9	-24.01	0	0.04	0.86	0.92	0.10	0.05	80.49	0	1
2745	50	0.28	4	-14.32	0	0.03	0.66	0.98	0.10	0.08	39.37	0	1
2751	45	0.51	1	-23.25	1	0.04	0.99	0.88	0.14	0.05	73.36	0	1
2754	39	0.08	11	-21.73	1	0.04	0.85	0.83	0.10	0.03	56.58	0	1
2764	38	0.11	11	-22.85	1	0.04	0.92	0.89	0.09	0.03	72.66	0	1

Son 3000 canciones “ olvidadas” aprox.



HITS POPULARES



PERLAS OLVIDADAS