

PREDICCIÓN DE GÉNEROS DE MÚSICA



Machine Learning project by
Ibai Valente

¿Crees que podrías adivinar el
género de una canción sin
escucharla?

¿Crees que podrías adivinar el género de una canción sin escucharla?





Bailable

Energía

Volumen

Hablado

Acústico

Instrumental

Directo

Valencia

Popularidad

Clave

Modo

Tempo















Regresión logística

Random forest

XGBoost

...

Regresión logística

Random forest

XGBoost

...

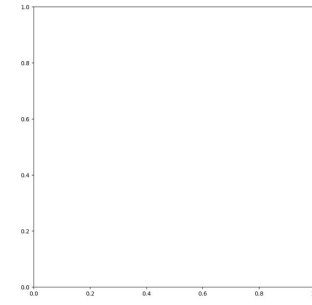
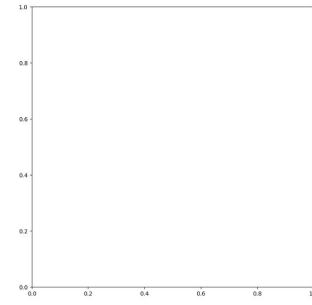
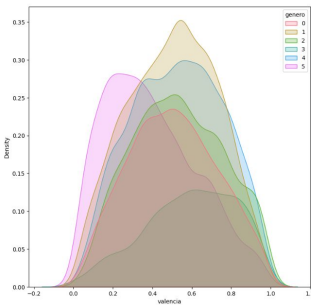
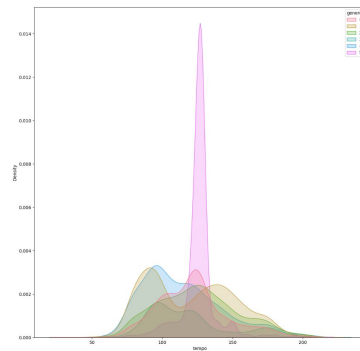
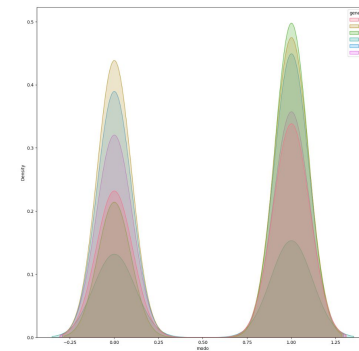
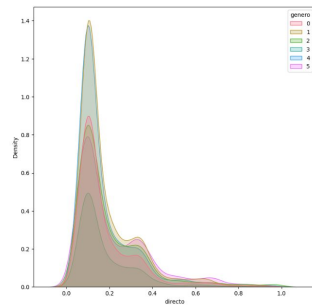
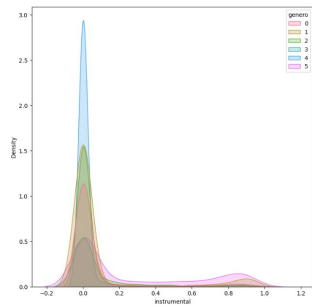
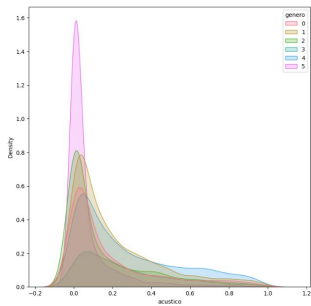
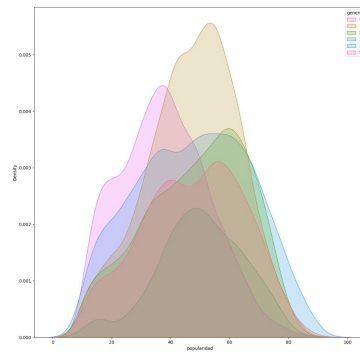
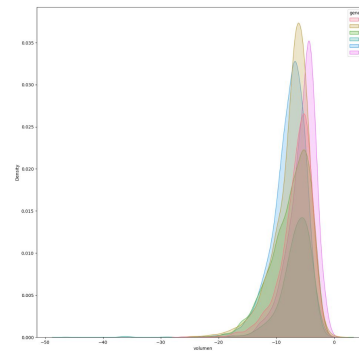
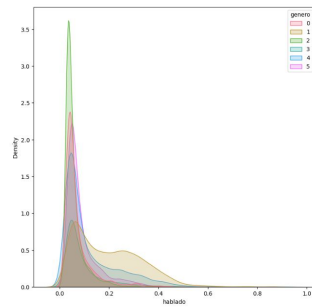
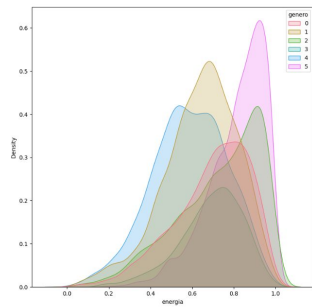
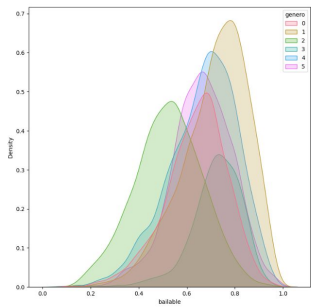
ACCURACY = 0.46

Regressi logística

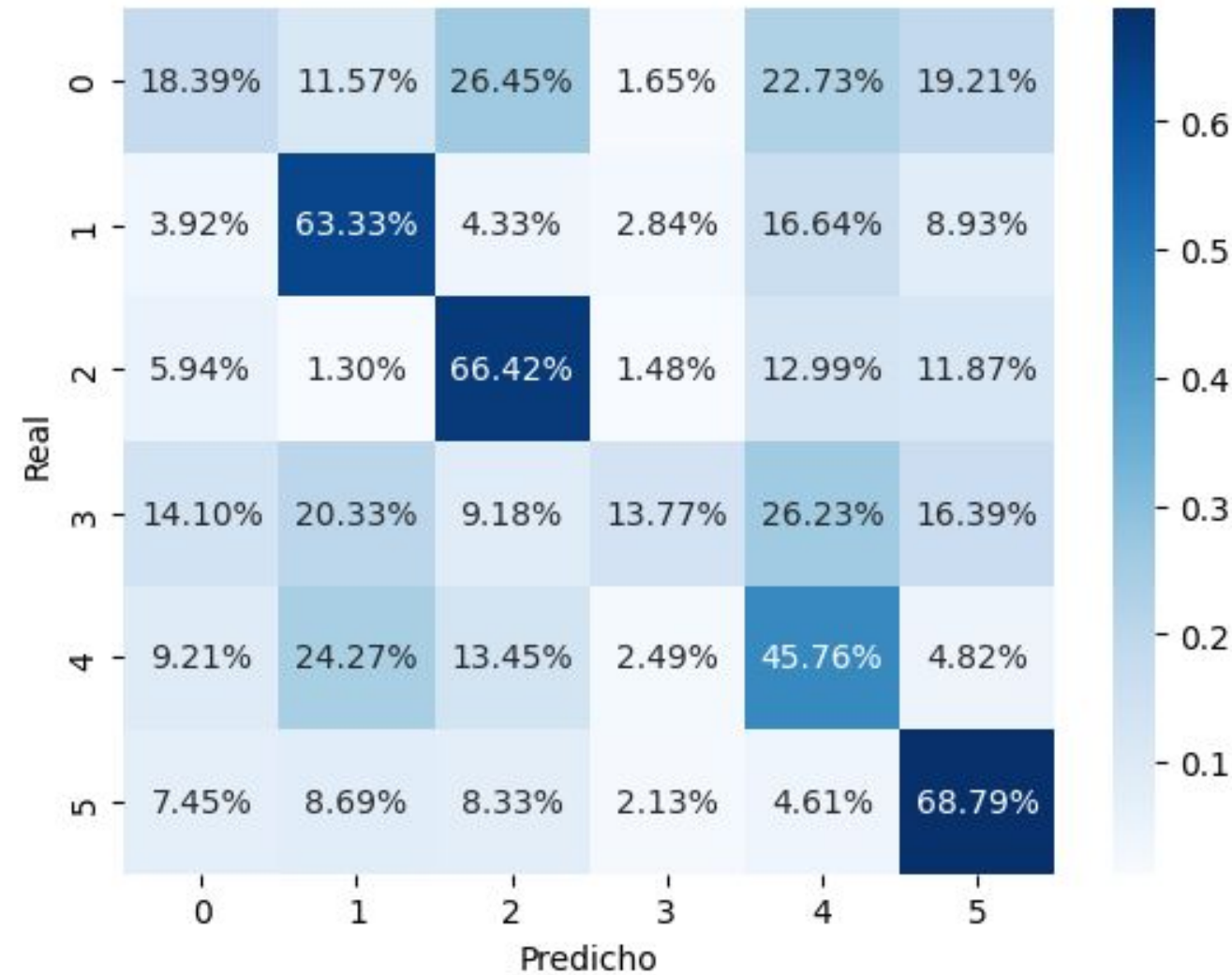
Ra = 0,46

Acc









0 = Pop



1 = Rap



2 = Rock



3 = Latina



4 = Blues



5 = Electrónica







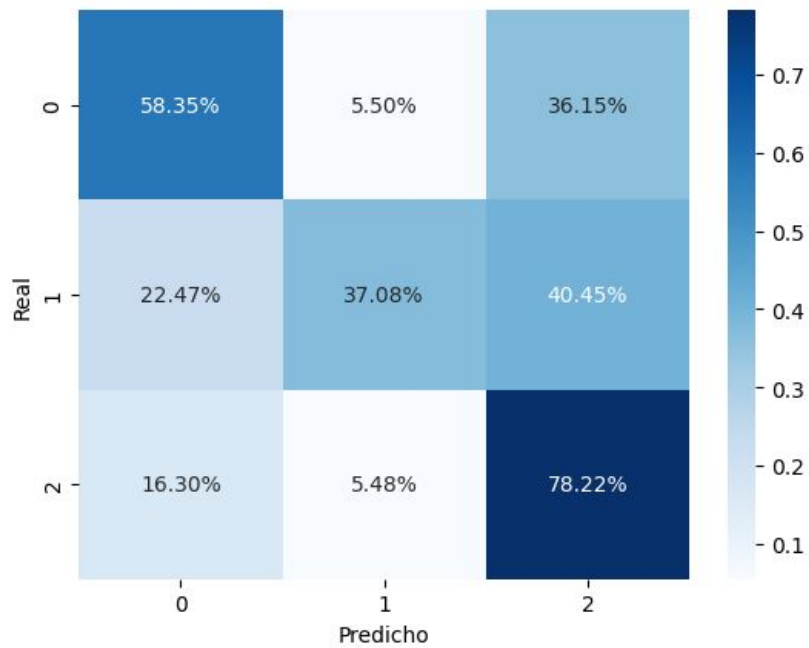
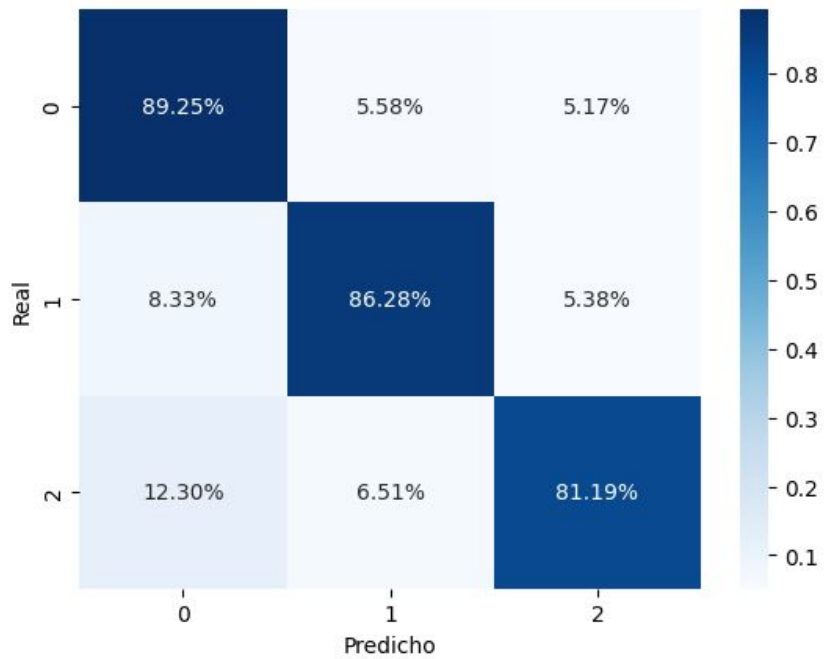












```
RandomForestClassifier {  
  max_depth = 7  
  n_estimators 100  
  max_features 10 }
```

```
XGBRFClassifier {  
  booster = gbtree  
  eta = 0,2  
  max_depth = 8  
  objective = multi:softprob  
  num_class = 3  
  eval_metric = mlogloss  
  n_estimators 500 }
```

```
XGBRFClassifier {  
  booster = gbtree  
  eta = 0,1  
  max_depth = 8  
  objective = multi:softprob  
  num_class = 3  
  eval_metric = mlogloss  
  n_estimators 300 }
```



```

1 import pandas as pd
2 import pickle
3 import datetime
4 dia_de_hoy = datetime.datetime.now().date()
5 import os
6
7 current_dir = os.path.dirname(os.path.realpath(__file__))
8
9 # Modelo de filtro preliminar
10 modelo_path_clasi = os.path.join(current_dir, '../modelos/final_clasi.pkl')
11 with open(modelo_path_clasi, 'rb') as file:
12     final_clasi = pickle.load(file)
13
14 # Modelo de clasificación de géneros buenos
15 modelo_path_buenos = os.path.join(current_dir, '../modelos/final_buenos.pkl')
16 with open(modelo_path_buenos, 'rb') as file:
17     final_buenos = pickle.load(file)
18
19 # Modelo de clasificación de géneros malos
20 modelo_path_malos = os.path.join(current_dir, '../modelos/final_malos.pkl')
21 with open(modelo_path_malos, 'rb') as file:
22     final_malos = pickle.load(file)
23
24 # Predicción de géneros de nuestro dataframe completo
25
26 os.chdir('C:\\Users\\Ibai Valente Lavado\\OneDrive\\Escritorio\\Data Science\\yo\\DS_TheBridgeBBK_S8IL2023\\3-Machine_Learning\\Entregas\\ML_project\\src\\predicciones')
27
28 generos_sin_target = pd.read_csv('../data/df_sin_target.csv')
29
30 predicciones_clasi = final_clasi.predict(generos_sin_target)
31 generos_sin_target['clasificacion'] = predicciones_clasi
32
33 # Predicciones de géneros "buenos" y "malos"
34 generos_sin_target['genero_predicho'] = None
35
36 for indice, fila in generos_sin_target.iterrows():
37     if fila['clasificacion'] == 0:
38         predicciones_malas = final_malos.predict(fila.drop(['clasificacion', 'genero_predicho']).values.reshape(1, -1))
39         predicciones_malas_transformadas = {0: 'pop', 1: 'latin', 2: 'R&B'}.get(predicciones_malas[0], predicciones_malas[0])
40         generos_sin_target.at[indice, 'genero_predicho'] = predicciones_malas_transformadas
41
42     else:
43         predicciones_buenas = final_buenos.predict(fila.drop(['clasificacion', 'genero_predicho']).values.reshape(1, -1))
44         predicciones_buenas_transformadas = {0: 'rap', 1: 'rock', 2: 'EDM'}.get(predicciones_buenas[0], predicciones_buenas[0])
45         generos_sin_target.at[indice, 'genero_predicho'] = predicciones_buenas_transformadas
46
47 # Guardar el DataFrame actualizado
48 generos_sin_target.to_csv(f'../predicciones/predicciones{dia_de_hoy}.csv', index=False)

```



Gracias por vuestra atención :)