

Memoria del Proyecto Final

1. Introducción	2
1.1. Descripción general del proyecto	2
1.2. Objetivo del análisis	2
1.3. Decisión de los modelos empleados	2
2. Análisis del Dataset	3
2.1 Formato Dataset	3
2.2 Procesamiento Dataset	3
2.3 Análisis del dataset	3
3. Modelos de IA Aplicados	6
3.1. Modelo KNN-Mbappé	6
3.2. Modelo MLP	7
Estructura	7
Transformaciones en el dataset:	7
Entrenamiento:	7
3.3. Modelo CNN- Denzel Washington	9
Estructura	9
Transformaciones en el dataset:	10
Entrenamiento:	10
3.4. CNN- Lola Lolita	12
Estructura:	12
Transformaciones en el dataset:	12
Entrenamiento	12
4. Resultados y Comparaciones	15
4.1 KNN-Mbappe	15
4.2 MLP-Napoleon	18
4.3 CNN-Denzel Washington	21
4.4 CNN-Lola Lolita	23
5. Interpretabilidad (XAI - Explicabilidad de los Modelos)	26
5.1. Modelo KNN-Mbappé	27
5.2. Modelo MLP	28
5.3. Modelo CNN- Denzel Washington	30
5.4. Modelo CNN- Lola Lolita	32
5.5. Conclusiones	34
6. División de tareas	35
7. Conclusiones	36

1. Introducción

1.1. Descripción general del proyecto

El proyecto que hemos decidido llevar a cabo se centra en la predicción de la edad, raza y género a partir de un [dataset](#) de 23.706 imágenes de personas, cada una etiquetada con sus respectivas categorías. Nos resultó especialmente interesante aplicar técnicas de XAI para identificar los rasgos más relevantes que influyen en la toma de decisiones del modelo. Además como el proyecto contiene imágenes de famosos hemos decidido asignarle un nombre a cada modelo.

1.2. Objetivo del análisis

El objetivo de este análisis es desarrollar una red neuronal que logre un buen desempeño en la clasificación de imágenes. Para ello, se propondrán distintos tipos de arquitecturas neuronales, con el fin de comparar sus resultados y determinar cuáles ofrecen un mejor rendimiento. Además, se emplearán técnicas de XAI para extraer conclusiones sobre las características más determinantes en el proceso de decisión de cada clasificación.

1.3. Decisión de los modelos empleados

Hemos decidido proponer redes convolucionales (CNN), MLP y k-NN para este problema de clasificación de edad, raza y género con imágenes. Nos parece una buena decisión porque permite abordar el problema desde diferentes perspectivas y comparar enfoques con distintas fortalezas

Redes convolucionales: son especialmente buenas para trabajar con imágenes porque captan patrones espaciales (bordes, formas, texturas) gracias a las capas convolucionales

MLP: son más simples y rápidos de entrenar que las CNN si las imágenes ya están aplanadas y normalizadas.

KNN: no requiere entrenamiento, solo almacena los datos y compara nuevas imágenes con los ejemplos más cercanos.

2. Análisis del Dataset

2.1 Formato Dataset

El dataset está compuesto por imágenes RGB de 200x200 píxeles y cada una de ellas tiene el siguiente formato:

edad_género_raza_fecha.jpg

donde:

- Edad: edad de la persona en cuestión (0-120)
- Género: género de la persona (Hombre o Mujer)
- Raza: raza de la persona (Blanco, Negro, Asiático, Indio, Otros)
- Fecha: fecha en la que fue tomada la imagen

Sin embargo, hemos decidido prescindir de este último atributo.

Cabe destacar que durante la elaboración del proyecto nos encontramos con 3,4 fotos cuyo formato no era correcto y provocaba errores durante el entrenamiento de los modelos, por lo que decidimos eliminar aquellas fotos. Teniendo en cuenta esto, es muy IMPORTANTE usar el conjunto de imágenes que vienen con el propio proyecto incluida en la carpeta UTKFace.

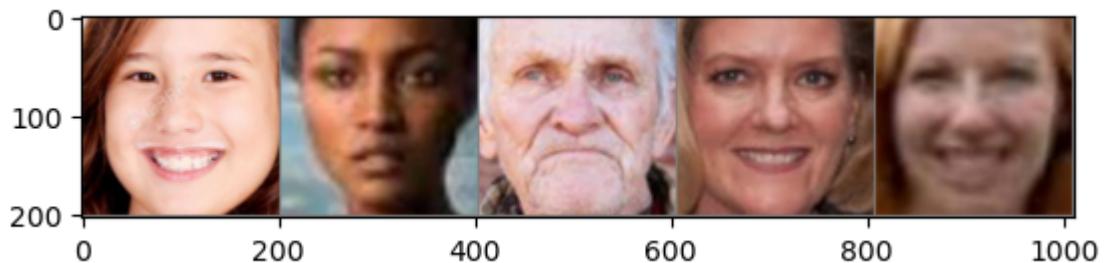
2.2 Procesamiento Dataset

Con el fin de acceder a cada una de las imágenes y a sus correspondientes atributos hemos decidido implementar una clase llamada Dataset. Esta clase hereda de la propia clase “Dataset” perteneciente a Pytorch y tiene 3 métodos:

- `__init__`: inicialización del dataset donde se indica la ruta en la que están las imágenes, y las posibles funciones o “transformadores” que permitirán convertir los datos en estructuras de datos válidas (tensores)
- `__len__`: devuelve el número de imágenes que contiene el dataset
- `__getitem__`: permite acceder a una función en particular y con los “transformadores” de la inicialización se convierten en tensores los atributos

2.3 Análisis del dataset

Una muestra de algunas imágenes del dataset es la siguiente:



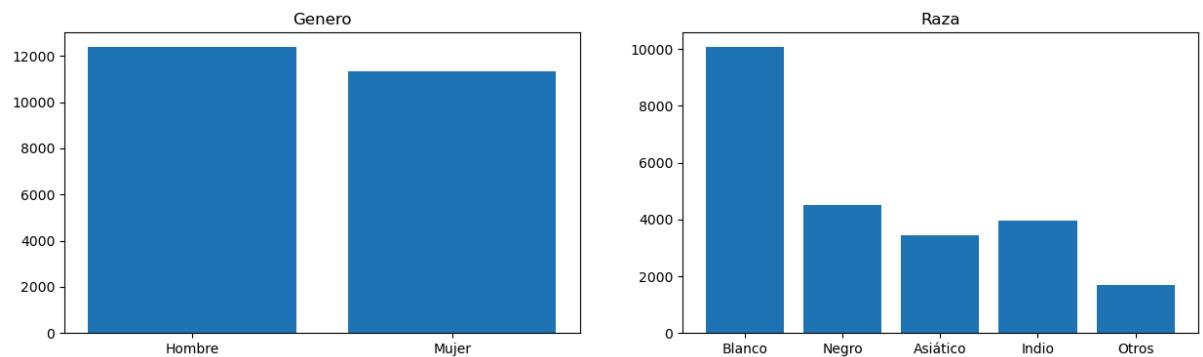
Las etiquetas reales de estas imágenes son respectivamente las que siguen:

Edad: 12.0 25.0 90.0 43.0 33.0

Genero: Mujer Mujer Hombre Mujer Mujer

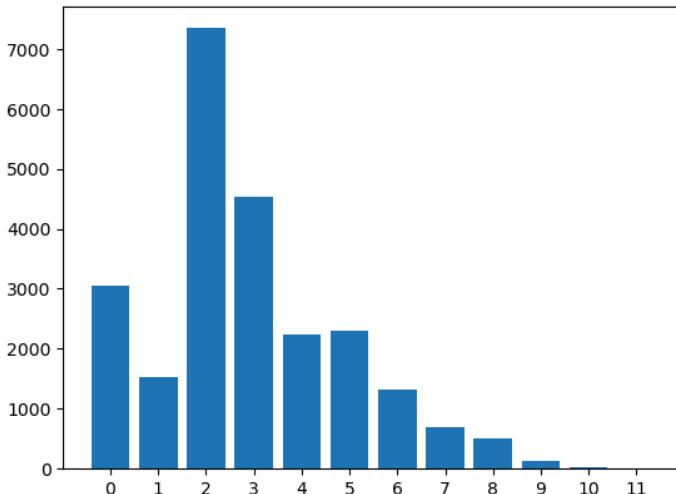
Raza: Asiático Negro Blanco Blanco Blanco

Con el fin de evaluar la calidad de la distribución de los datos, llevamos a cabo un análisis exploratorio. Nuestro objetivo era comprobar que las distintas categorías estuviesen equilibradas y que no existiera un sesgo significativo hacia alguna de ellas. A continuación adjuntamos una imagen sobre la distribución de los datos en cuanto a género y raza:



Observamos que, en la categoría de género, ambas clases presentan una distribución bastante equilibrada. Sin embargo, en la categoría de raza se evidencia un fuerte desequilibrio, con una predominancia notable de imágenes de personas blancas en comparación con las demás clases.

Una vez estudiadas las categorías género y raza , mostramos la distribución de las edades en el conjunto de datos. Sin embargo, como el rango de edades es tan amplio (0-120) decidimos agruparlas en grupos o rangos de 10 años; en consecuencia, se han definido un total de 12 clases etarias, organizadas de la siguiente manera: la clase 0 agrupa a personas de entre 0 y 9 años, la clase 1 de 10 a 19 años, y así sucesivamente en intervalos de diez años.



También se puede apreciar una marcada desproporción en la clase 2 (personas de 20 a 29 años), la cual presenta un número significativamente mayor de muestras en comparación con el resto de grupos de clases.

En resumen, al analizar la distribución de los datos vimos que hay ciertos desequilibrios, sobre todo en las categorías de raza y edad. Por ejemplo, hay muchas más imágenes de personas blancas y del grupo de 20 a 29 años, lo que puede hacer que el modelo aprenda a identificar mejor esas clases y tenga más dificultades con las menos representadas. Esto podría afectar al rendimiento general y hacer que los resultados no sean igual de buenos para todas las categorías.

3. Modelos de IA Aplicados

3.1. Modelo KNN-Mbappé

Para implementar este modelo hemos decidido utilizar las librerías de SCIKIT-LEARN, las cuales permiten trabajar cómodamente con tensores de pytorch y no es necesario hacer un tratamiento nuevo de las imágenes.

El modelo recibirá imágenes RGB de 200x200 pixeles normalizadas y la edad de cada imagen será clasificada en rangos de 10 años, por lo que el modelo no será capaz de predecir la edad exacta sino una estimación.

Además, el conjunto total será dividido en un 80% entrenamiento y un 20%

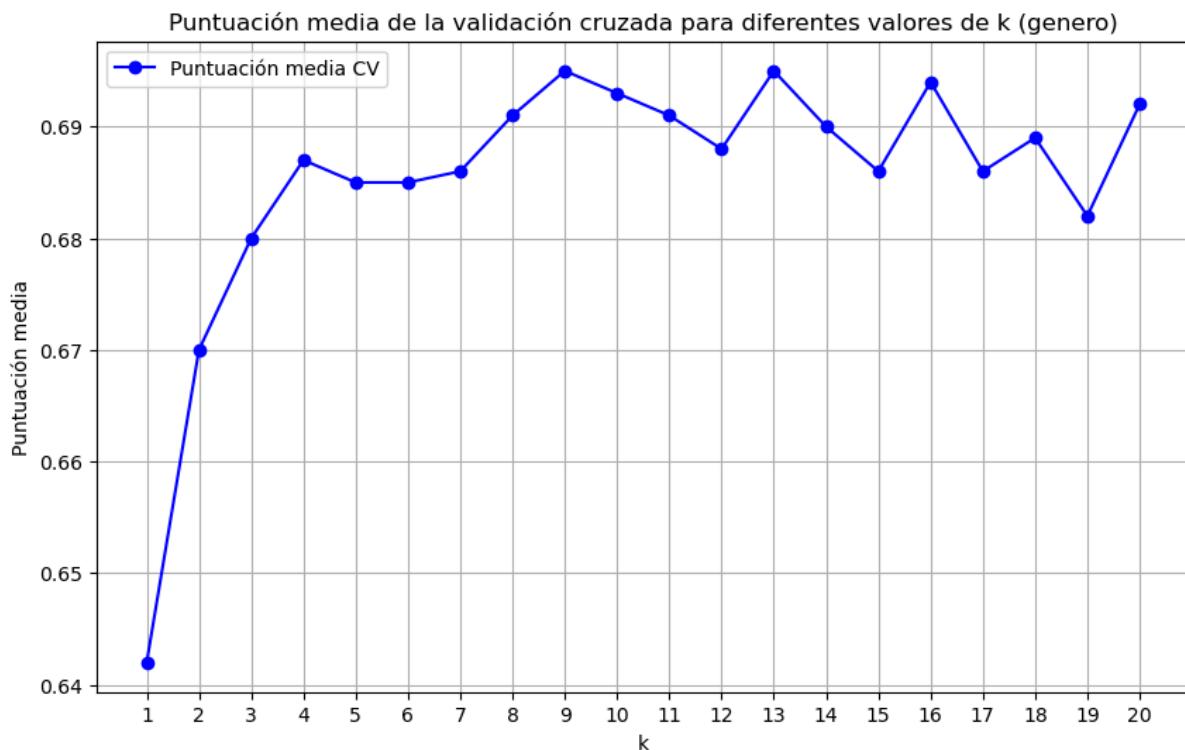
Por último, en KNN es necesario definir o asignar el famoso número K de vecinos por lo que decidimos implementar una función llamada

```
encontrar_mejor_k(dataset,categoría,num_imagenes,k_range)
```

Esta función se encarga de calcular empíricamente el mejor K para cada problema de clasificación(género,raza o edad) mediante cross-validation. Sin embargo, no se hace sobre el conjunto total de entrenamiento debido a la ineficiencia de SCIKIT-LEARN que requiere cargar todos los datos en RAM y hacia que, en determinados momentos de la ejecución, el equipo dejase de funcionar; por eso el argumento “num_imagenes” el cual selecciona num_imagenes del conjunto de entrenamiento y aplica cross-validation.

Obviamente, lo ideal hubiera sido aplicarlo sobre todo el conjunto de entrenamiento pero era imposible por el volumen de datos necesarios a cargar en RAM (>13 GB)

La siguiente imagen muestra la gráfica y el comportamiento del modelo género según los distintos K:



3.2. Modelo MLP

En este modelo implementaremos una red neuronal convencional usando Pytorch.

Estructura

El modelo utilizado está compuesto por tres capas totalmente conectadas. Teniendo en cuenta que recibirá imágenes RGB de 64x64 aplazadas, la estructura es la siguiente:

- 1º capa: fc1 (in = 64x64x3, out=512)
- 2º capa: fc2 (in = 512, out = 256)
- 3º capa : fc2 (in = 256, out = depende del problema)

Entre las capas ocultas se emplea la función de activación ReLU, lo que permite introducir no linealidad, reducir la complejidad del modelo y mejorar la eficiencia durante el entrenamiento.

Para definir esta arquitectura, se ha utilizado la librería PyTorch, debido a que proporciona herramientas potentes, eficientes y flexibles para el procesamiento de imágenes.

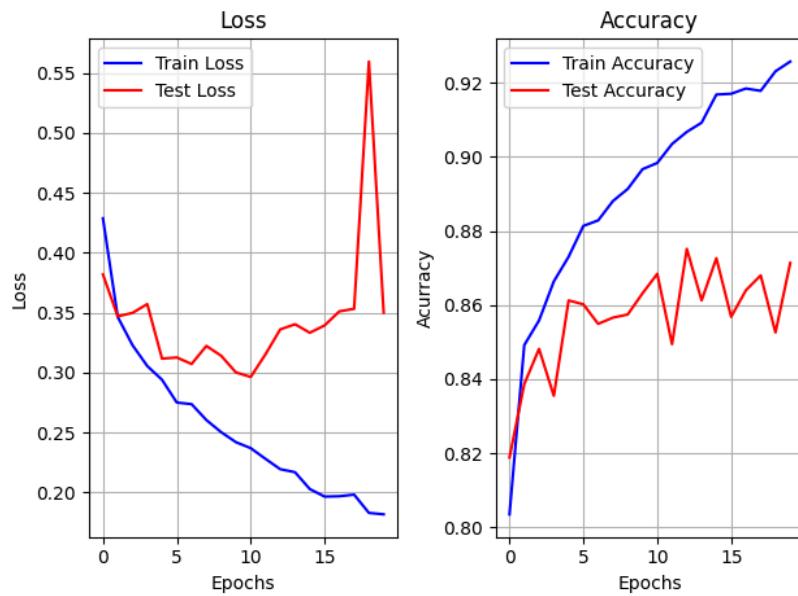
Transformaciones en el dataset:

Se dividen los datos en entrenamiento y prueba con una proporción 80/20, y un batch size de 4. Como ya hemos comentado, las imágenes se redimensionan a 64x64 para reducir coste computacional y se normalizan. En el MLP, la entrada debe ser un vector unidimensional. Por lo tanto, la imagen de tamaño 64x64x3 se aplana a un vector de 12.288 elementos. Además hay que recalcar que la edad de cada imagen será clasificada en rangos de 10 años.

Entrenamiento:

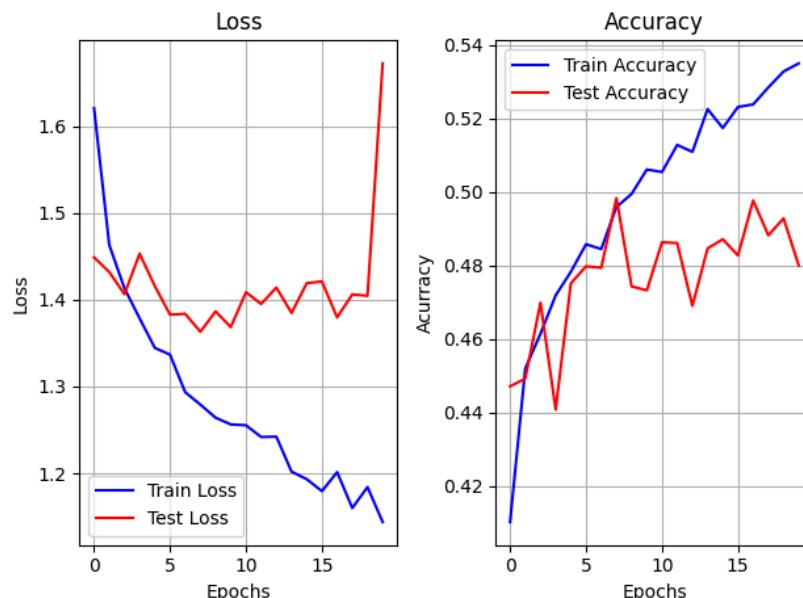
Para el entrenamiento, es importante destacar que utilizamos la función de pérdida *CrossEntropyLoss* en todos los modelos, ya que todos los problemas planteados fueron abordados como tareas de clasificación.

Nos pareció útil generar gráficas de precisión durante el entrenamiento para identificar en qué momento el modelo deja de mejorar y comienza a sobre aprender. A continuación, adjuntamos las imágenes de la evaluación del modelo género:



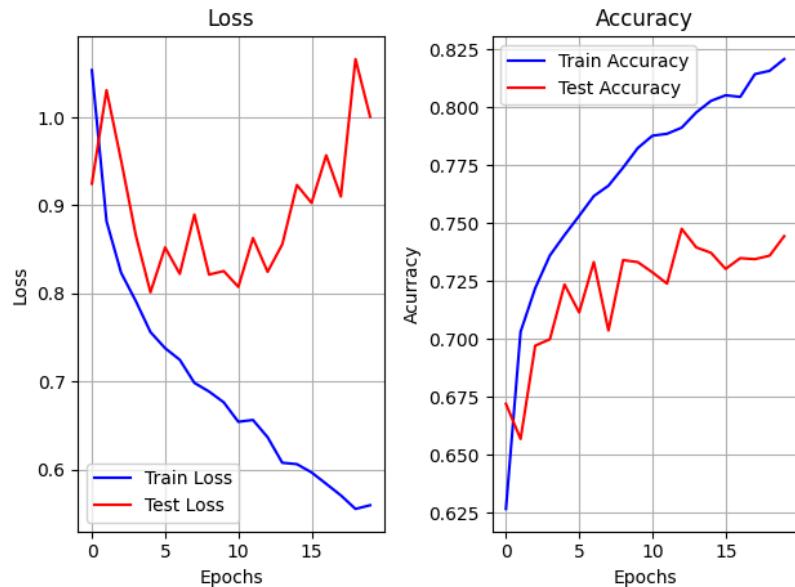
Este modelo se entrenó durante 20 épocas. Se observa que, a partir de la época 12, la precisión en el conjunto de prueba deja de mejorar y no vuelve a alcanzar los valores de esa época, lo que sugiere que detener el entrenamiento en ese punto sería lo más adecuado para obtener los mejores resultados.

A continuación, adjuntamos la imagen de la evolución del entrenamiento del modelo para la clasificación de edad:



Este modelo se entrenó durante 20 épocas. Se observa que, alrededor de la época 7, el conjunto de prueba alcanza la precisión máxima durante todo el entrenamiento.

Adjuntamos la imagen de la evolución del entrenamiento del modelo para la clasificación de raza:



Este modelo se entrenó durante 20 épocas. Se observa que a partir de la época 12 (aproximadamente), la precisión en el conjunto de prueba toma el valor máximo.

Estos modelos están entrenados en CPU y tardaron todos aproximadamente 1 hora y 30 minutos en entrenarse cada uno

3.3. Modelo CNN- Denzel Washington

En el tercer modelo decidimos implementar una red convolucional. Al igual que el modelo anterior utilizamos Pytorch.

Estructura

Esta red convolucional tiene la siguiente estructura:

- 1 filtro/kernel: MaxPooling con tamaño (2, 2)
- 1º capa convolucional “conv1(in= 3, out= 6,kernel_size= 5)”
- 2º capa convolucional “conv2(in= 6, out= 12,kernel_size= 5)”
- 1 filtro/kernel “pool2 = MaxPool2D(4,4)”
- 1º capa lineal “fc1(in = 12x11x11, out = 500)
- capa salida “fcout(in = 500, out=depende del problema)”

Para entender mejor la red veamos como sería el procesamiento de una imagen. Para ello tenemos que tener en cuenta que la red recibirá imágenes RGB de 200x200 pixeles normalizadas:

- Recibe la imagen: 3 canales de 200x200
- Aplicamos el filtro pool1 => obtenemos 3 mapas de 100x100
- La 1º capa convolucional recibe los 3 mapas => 6 mapas de 96x96
- Aplicamos otra vez el filtro pool1 => obtenemos 6 mapas de 48x48
- La 2º capa convolucional recibe los 6 mapas => 12 mapas de 44x44
- Aplicamos el filtro pool2 => 12 mapas de 11x11
- Pasamos el resultado por las capas lineales hasta llegar a la capa final

Transformaciones en el dataset:

Como ya hemos comentado, cada una de las imágenes RGB de 200x200 pixeles se normaliza. Sin embargo, a diferencia de los anteriores modelos, la edad no se agrupa en rangos, es decir, el modelo intentará predecir la edad exacta.

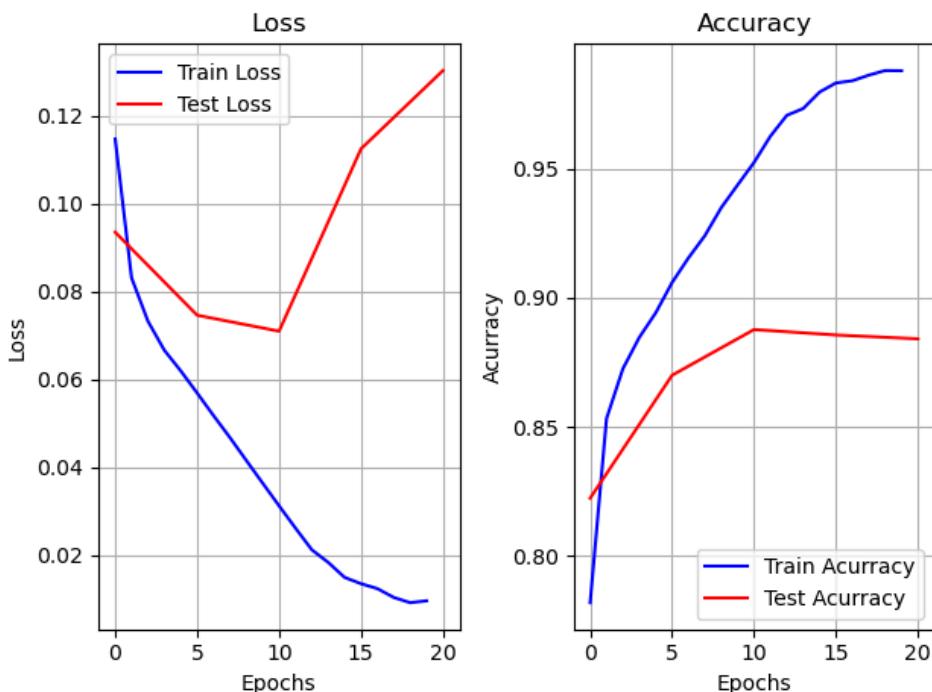
El dataset se dividirá en un 80% entrenamiento y 20% test y en batches de 4 imágenes

Entrenamiento:

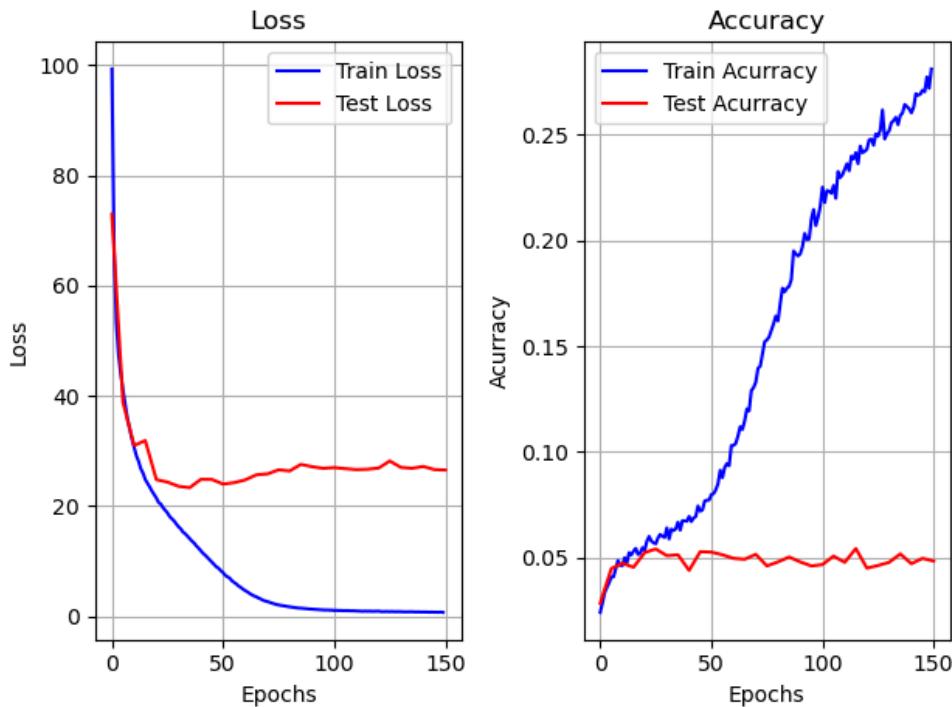
Para el entrenamiento hemos usado CrossEntropyLoss para los modelos encargados de predecir el género y la raza, mientras que para la edad hemos decidido usar MSELoss (Error Cuadrático Medio) pues recordemos que el modelo intentará predecir la edad exacta (aquí la edad no es un problema de clasificación)

Al igual que en el anterior modelo decidimos generar gráficas para ver la progresión del entrenamiento:

Gráfica para el modelo género:

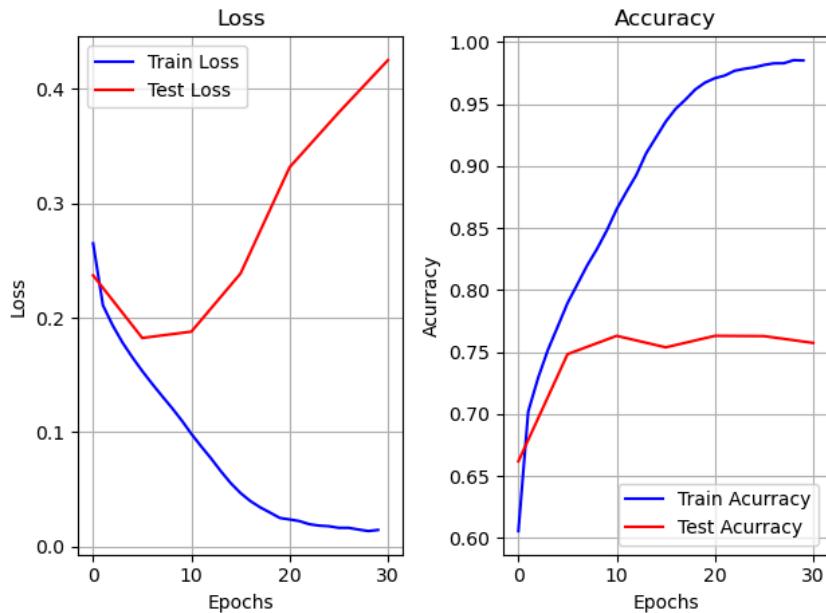


Gráfica del modelo Edad:



En esta gráfica observamos que el número de epochs tan alto era innecesario pues a partir del epoch 25 aproximadamente no hay mejora alguna sobre el conjunto de test. Aun así, el modelo tampoco sobre aprende tanto pues la tasa de acierto es de un 25%

Gráfica del modelo Raza:



El entrenamiento del género y la raza llevó aproximadamente unos 30 minutos mientras que el de la edad fueron más de 2 horas. Dicho entrenamiento se realizó en GPU

3.4. CNN- Lola Lolita

En el cuarto modelo, decidimos implementar otra red convolucional pero más compleja que la anterior, es decir, con más capas convolucionales y filtros con el fin de intentar captar o aprender más patrones. Volvemos a utilizar Pytorch

Estructura:

Este nuevo modelo tiene la siguiente estructura:

- 1º filtro pool1: MaxPooling con tamaño (2, 2).
- 1º capa convolucional “conv1(in= 3, out= 6,kernel_size= 3)”
- 2º capa convolucional “conv2(in= 6, out= 12,kernel_size= 4)”
- 3º capa convolucional “conv3(in= 12, out= 24,kernel_size= 6)”
- 4º capa convolucional “conv4(in= 24, out= 48,kernel_size= 16)”
- 2º filtro pool2: MaxPooling con tamaño (3, 3)
- 1º capa lineal “fc1(in = 46x10x10, out = 1200)
- 2º capa lineal “fc2(in = 1200, out = 850)
- Salida: “fcout(in =850, out= depende del problema”

Para entender cómo funciona esta red, consideraremos el procesamiento de una imagen. Recibimos una imagen RGB de 200x200 píxeles normalizada. El procesamiento es el siguiente:

- Recibe la imagen: 3 canales de 200x200
- Aplicamos el 1º filtro pool1 => obtenemos 3 mapas de 100x100
- La 1º capa convolucional recibe los 3 mapas => 6 mapas de 98x98
- La 2º capa convolucional recibe los 6 mapas => 12 mapas de 95x95
- La 3º capa convolucional recibe los 12 mapas => 24 mapas de 90x90
- Aplicamos el 1º filtro pool1 => 24 mapas de 45x45
- La 4º capa recibe los 24 mapas => 46 mapas de 30x30
- Aplicamos el 2º filtro pool2 => 46 mapas de 10x10
- Aplanamos obteniendo 46x10x10 nodos y los pasamos por las capas lineales

Transformaciones en el dataset:

Cada una de las imágenes RGB de 200x200 pixeles se normaliza. Este modelo también intentará predecir la edad exacta.

El dataset se dividirá en un 80% entrenamiento y 20% test y en batches de 4 imágenes

Entrenamiento

A continuación adjuntamos imágenes al igual que en los modelos anteriores del proceso de entrenamiento.

Gráfico de género:

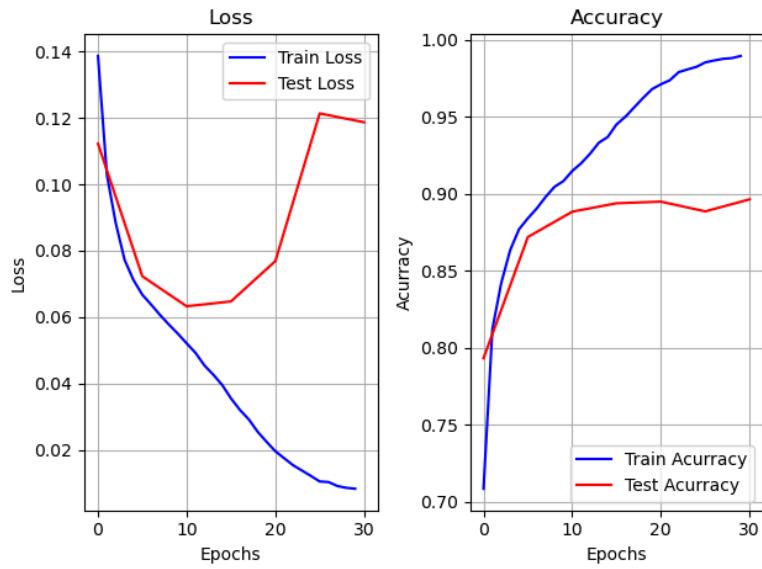


Gráfico de edad:

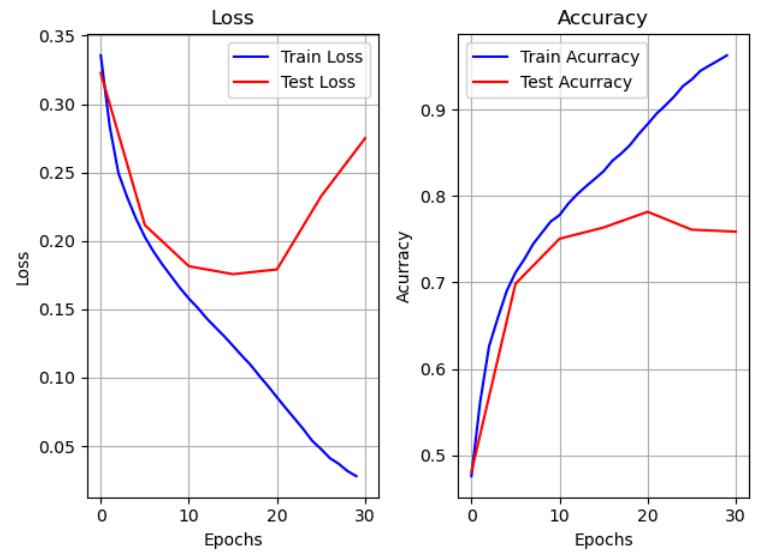
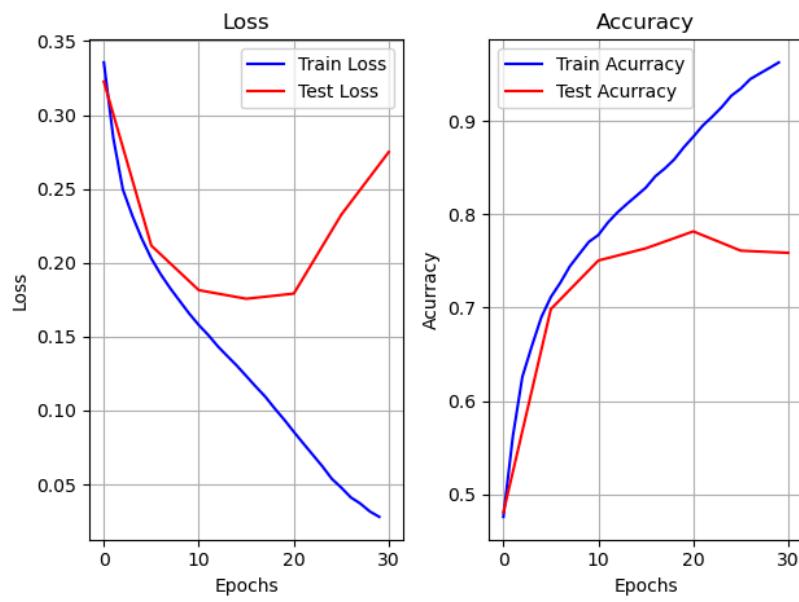


Gráfico de raza:



El entrenamiento del género y la raza llevó aproximadamente una hora mientras que el de la edad fueron casi 3 horas. El entrenamiento se realizó en GPU

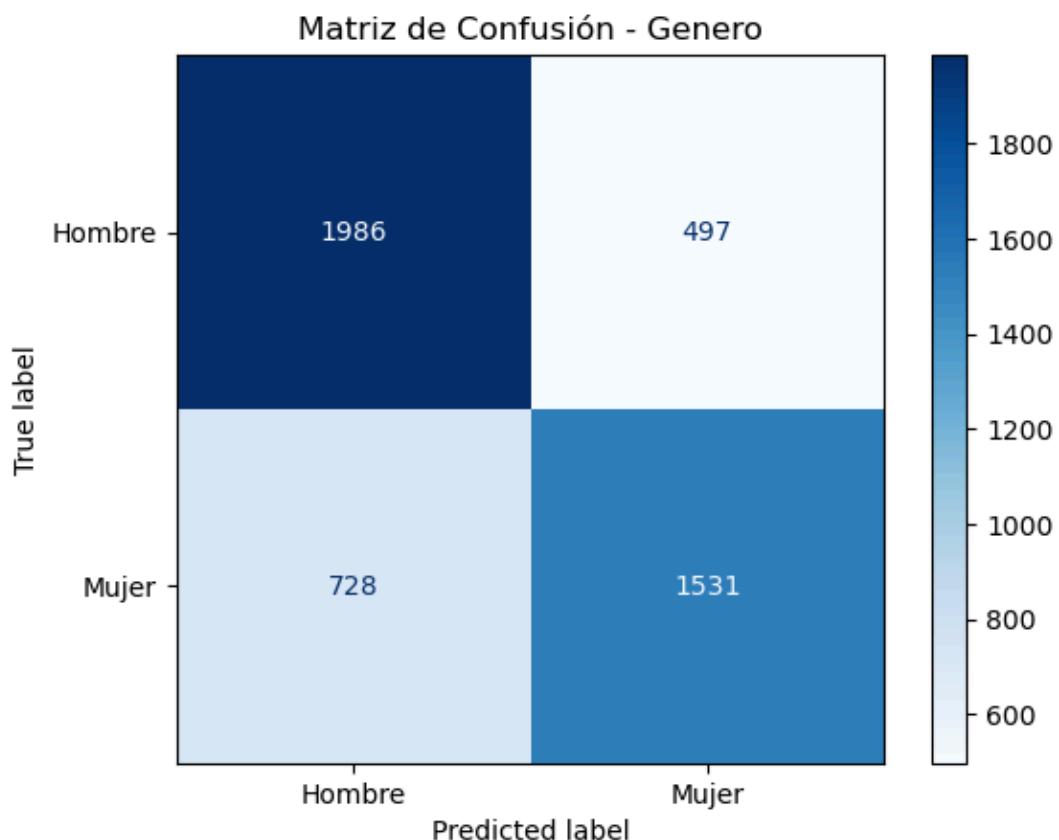
4. Resultados y Comparaciones

En este apartado vamos a mostrar y comparar los resultados obtenidos de los distintos modelos:

4.1 KNN-Mbappe

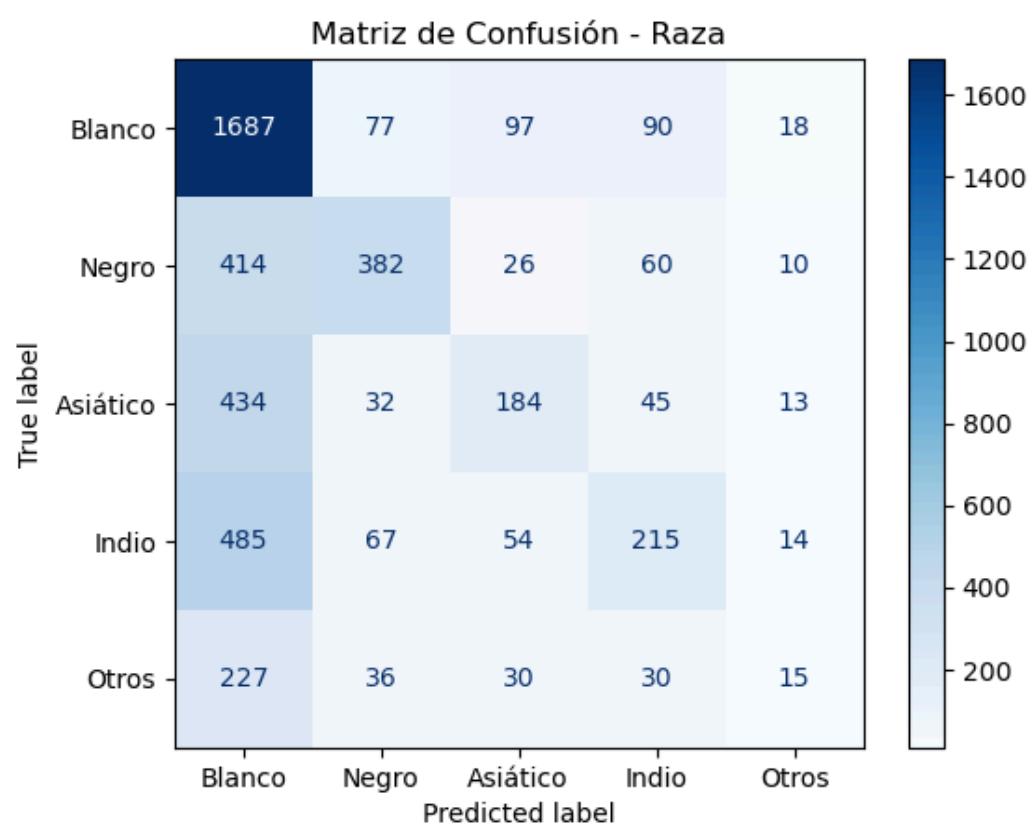
Género:

	precision	recall	f1-score	support
Hombre	0.73	0.80	0.76	2483
Mujer	0.75	0.68	0.71	2259
accuracy			0.74	4742
macro avg	0.74	0.74	0.74	4742
weighted avg	0.74	0.74	0.74	4742



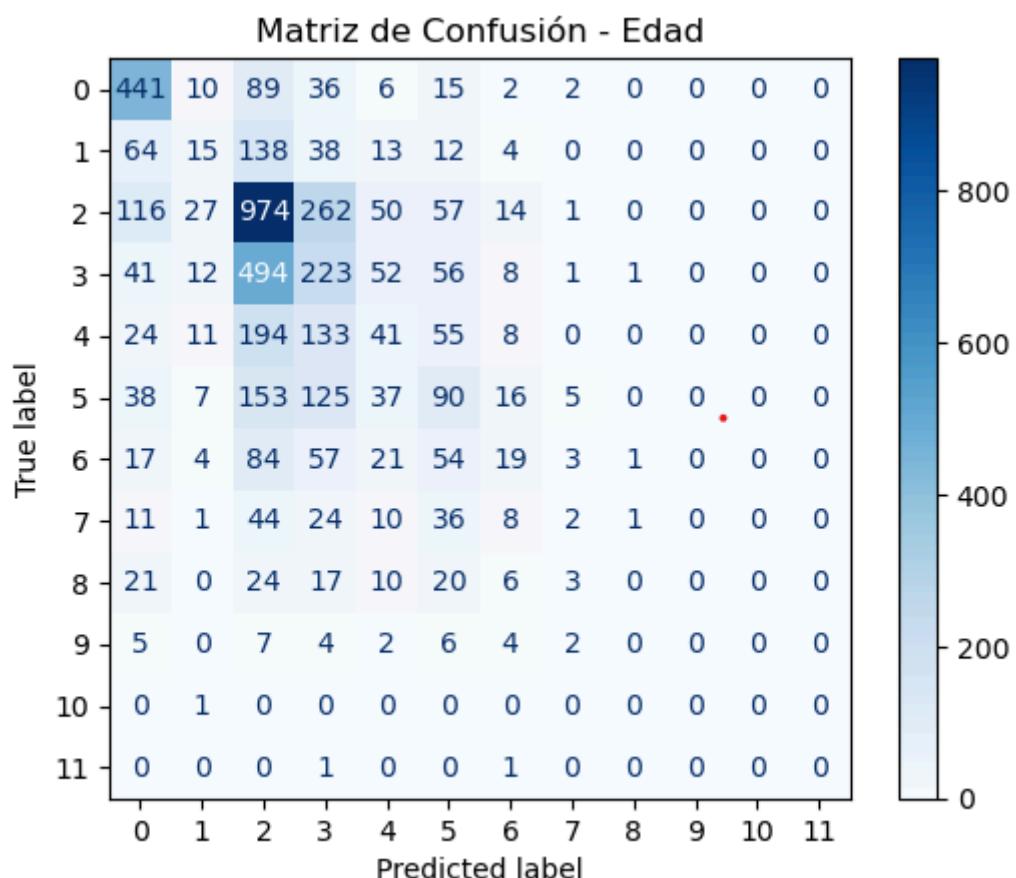
Raza:

	precision	recall	f1-score	support
Blanco	0.52	0.86	0.65	1969
Negro	0.64	0.43	0.51	892
Asiático	0.47	0.26	0.33	708
Indio	0.49	0.26	0.34	835
Otros	0.21	0.04	0.07	338
accuracy			0.52	4742
macro avg	0.47	0.37	0.38	4742
weighted avg	0.51	0.52	0.48	4742



Edad:

	precision	recall	f1-score	support
0-9	0.57	0.73	0.64	601
10-19	0.17	0.05	0.08	284
20-29	0.44	0.65	0.53	1501
30-39	0.24	0.25	0.25	888
40-49	0.17	0.09	0.12	466
50-59	0.22	0.19	0.21	471
60-69	0.21	0.07	0.11	260
70-79	0.11	0.01	0.03	137
80-89	0.00	0.00	0.00	101
90-99	0.00	0.00	0.00	30
100-109	0.00	0.00	0.00	1
110-120	0.00	0.00	0.00	2
accuracy			0.38	4742
macro avg	0.18	0.17	0.16	4742
weighted avg	0.32	0.38	0.34	4742



Obtenemos resultados decentes:

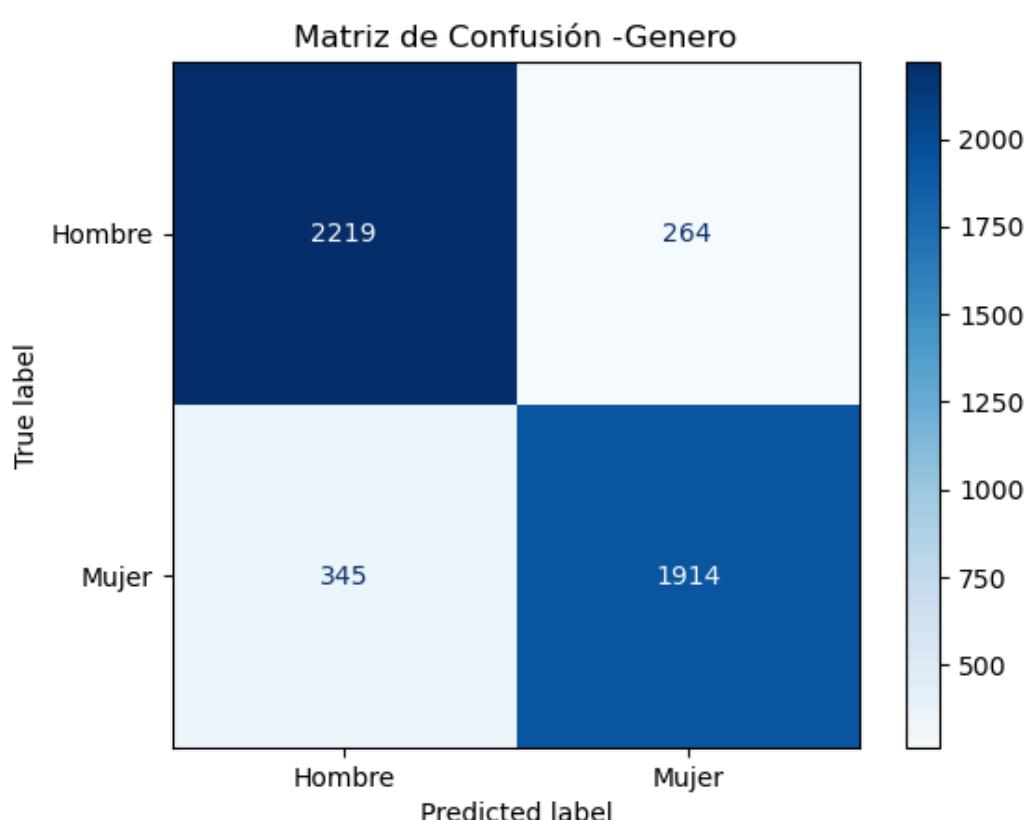
- GENERO : buen comportamiento con una tasa de aciertos del 75%% tanto para hombres como mujeres.Aun así, vemos que el modelo es bastante mas exhaustivo con hombres y en consecuencia es capaz de predecir mas hombres correctamente que mujeres
- RAZA: roza casi el aprobado con un 47% de tasa de acierto. La clase que mejor se le da es BLANCO
- EDAD: deja bastante que desear, pero no se la del todo mal la clase 0 y la clase 2: 0-9 años y 20-29 años respectivamente. Justo las clases que mas abundan en el dataset

Los resultados no son malos del todo, pero tienen margen de mejora.

4.2 MLP-Napoleon

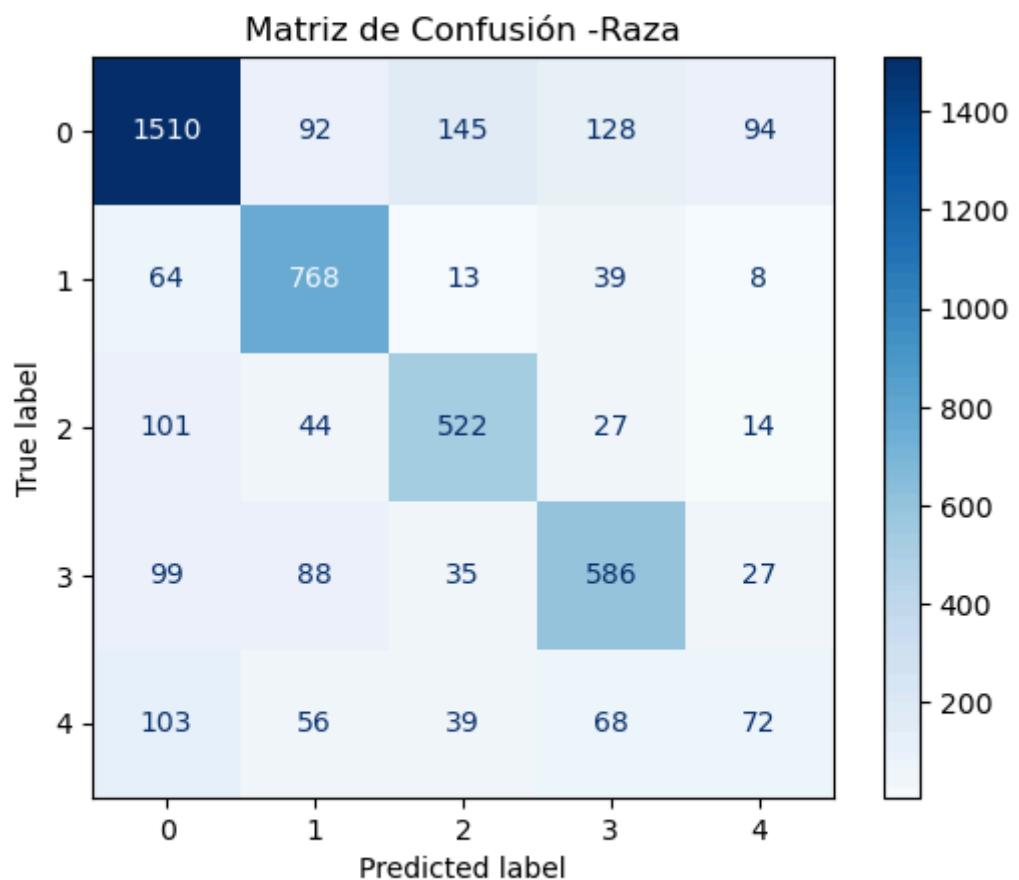
Género:

	precision	recall	f1-score	support
Hombre	0.87	0.89	0.88	2483
Mujer	0.88	0.85	0.86	2259
accuracy			0.87	4742
macro avg	0.87	0.87	0.87	4742
weighted avg	0.87	0.87	0.87	4742



Raza:

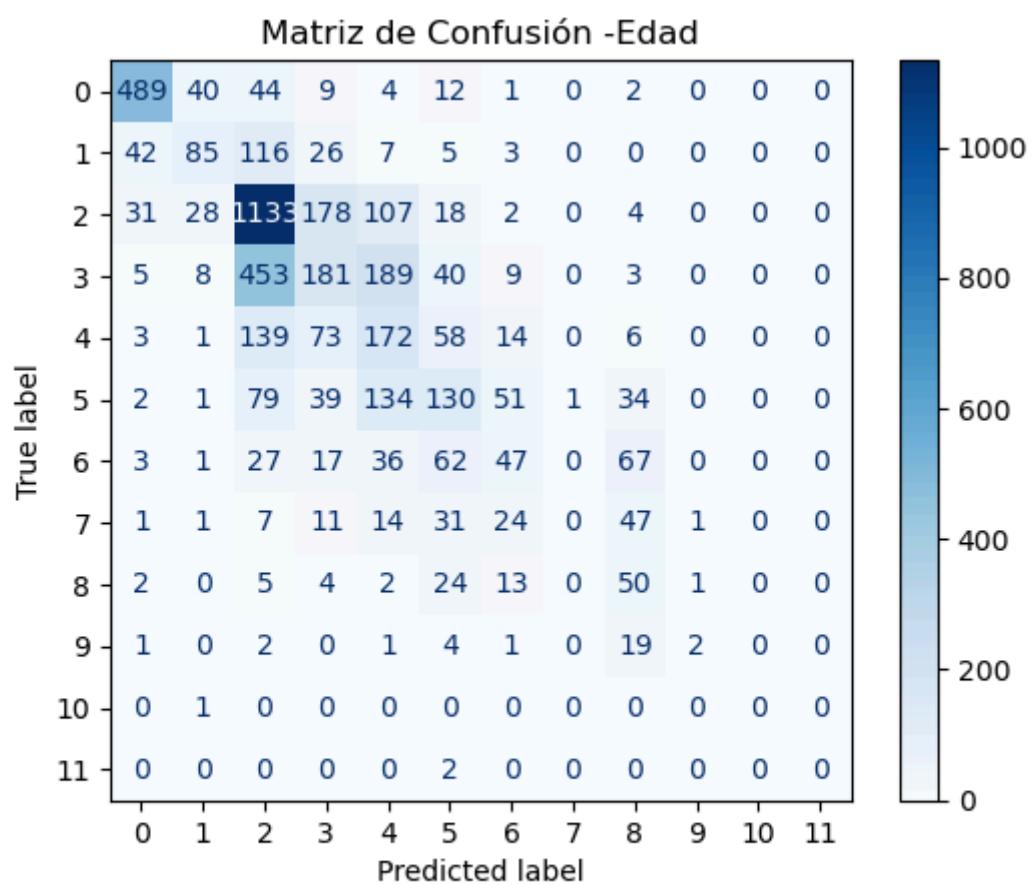
	precision	recall	f1-score	support
Blanco	0.80	0.77	0.79	1969
Negro	0.73	0.86	0.79	892
Asiático	0.69	0.74	0.71	708
Indio	0.69	0.70	0.70	835
Otros	0.33	0.21	0.26	338
accuracy			0.73	4742
macro avg	0.65	0.66	0.65	4742
weighted avg	0.72	0.73	0.72	4742



«

Edad

	precision	recall	f1-score	support
0-9	0.84	0.81	0.83	601
10-19	0.51	0.30	0.38	284
20-29	0.57	0.75	0.65	1501
30-39	0.34	0.20	0.25	888
40-49	0.26	0.37	0.30	466
50-59	0.34	0.28	0.30	471
60-69	0.28	0.18	0.22	260
70-79	0.00	0.00	0.00	137
80-89	0.22	0.50	0.30	101
90-99	0.50	0.07	0.12	30
100-109	0.00	0.00	0.00	1
110-120	0.00	0.00	0.00	2
accuracy			0.48	4742
macro avg	0.32	0.29	0.28	4742
weighted avg	0.46	0.48	0.46	4742



En este modelo obtenemos en general, mejores resultados que el primero:

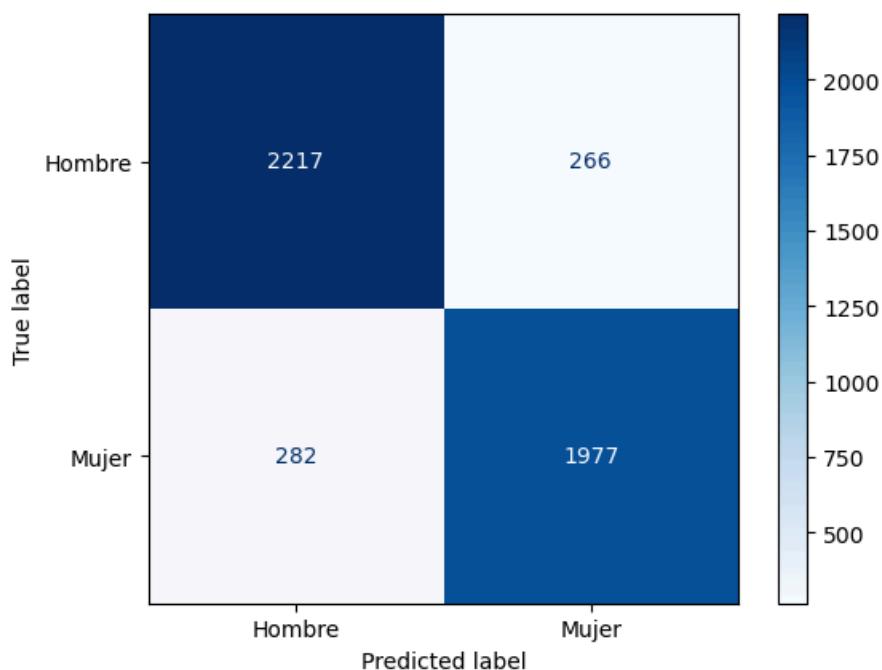
- GENERO : muy buen comportamiento con una tasa de aciertos del 87% tanto para hombres como mujeres. Sin embargo, vemos que el modelo es más exhaustivo con hombres
- RAZA: buen comportamiento, aunque no tanto como el GENERO pues tenemos una tasa de aciertos del 65%
- EDAD: el comportamiento en general no es muy bueno; sin embargo, es bastante bueno con edades de entre 0 y 29 años, en especial entre 0 y 9. Lo cual tiene sentido pues si recordamos el análisis del dataset eran las clases que más abundaban en el dataset

En general, hemos conseguido mejorar bastante los resultados con respecto al primer modelo.

4.3 CNN-Denzel Washington

Género:

	precision	recall	f1-score	support
Hombre	0.89	0.89	0.89	2483
Mujer	0.88	0.88	0.88	2259
accuracy			0.88	4742
macro avg	0.88	0.88	0.88	4742
weighted avg	0.88	0.88	0.88	4742

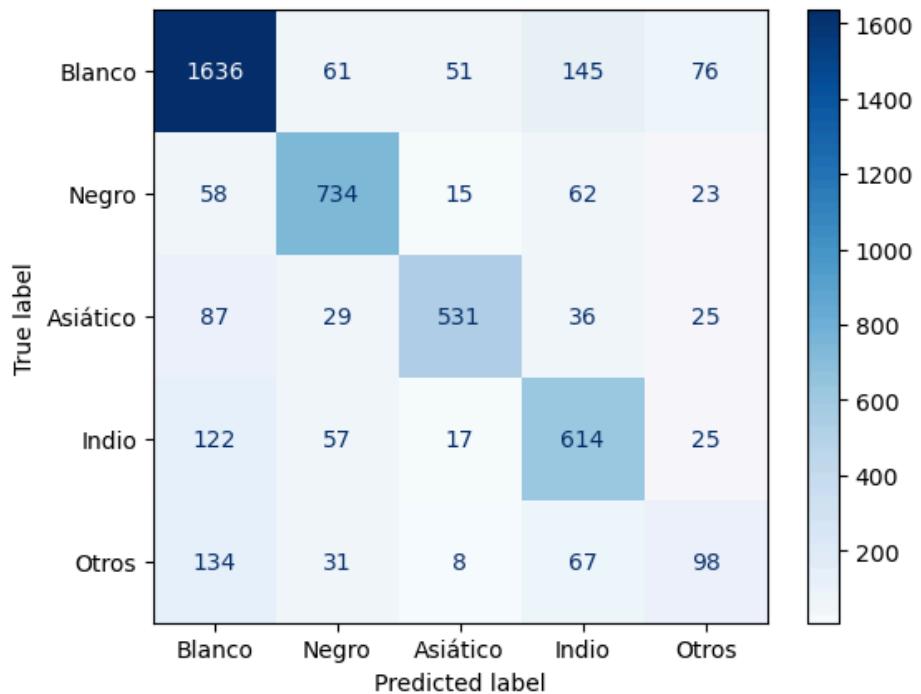


Edad:

Para la edad, al no ser un problema de clasificación no tiene sentido hacer una matriz de confusión. Vemos entonces que el modelo tiene una tasa de acierto de 4.8291%. Sin embargo, esto no es muy representativo pues acertar la edad exacta es bastante complicado y nos interesa saber cuánto se equiva. Para ello calculamos el error absoluto medio: 7.5643. Es decir, de media, el modelo tiene un margen de error de 7 años.

Raza:

	precision	recall	f1-score	support
Blanco	0.80	0.83	0.82	1969
Negro	0.80	0.82	0.81	892
Asiático	0.85	0.75	0.80	708
Indio	0.66	0.74	0.70	835
Otros	0.40	0.29	0.34	338
accuracy			0.76	4742
macro avg	0.70	0.69	0.69	4742
weighted avg	0.76	0.76	0.76	4742



Volvemos a obtener mejores resultados:

- GENERO : muy buen comportamiento con una tasa de aciertos del 88%. Sin embargo, se le da mejor predecir hombres que mujeres, pero la diferencia es mínima.

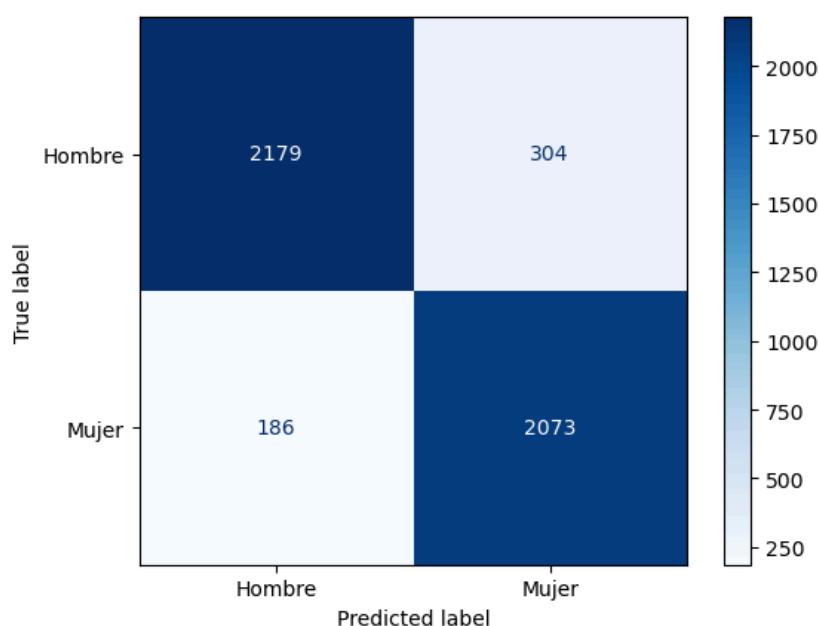
- RAZA: buen comportamiento, aunque no tanto como el GENERO pues tenemos una tasa de aciertos del 70%. Vemos que, comparados con las otras clases, le cuesta la clase INDIO y OTROS, en especial esta última
- EDAD: si nos centramos solo en la tasa de acierto, obviamente el resultado es muy malo, peor incluso que los anteriores modelos. Aun así, tenemos que tener en cuenta que el modelo intenta predecir la edad exacta, por lo que tenemos que usar otra métrica, es decir, tenemos que ver cuánto se equivoca: en este caso, el margen de error es de 7 años lo cual no es tan malo

En general, al igual que antes hemos conseguido mejorar los resultados con respecto a los anteriores, pero la diferencia es mínima. Sin embargo, creemos que este modelo puede ser mejor que los otros en tema de la EDAD, pues los intervalos en los que se clasifican las edades de los primeros modelos son de 10 años y aquí el margen error es de 7 años

4.4 CNN-Lola Lolita

Género:

	precision	recall	f1-score	support
Hombre	0.92	0.88	0.90	2483
Mujer	0.87	0.92	0.89	2259
accuracy			0.90	4742
macro avg	0.90	0.90	0.90	4742
weighted avg	0.90	0.90	0.90	4742



Raza:

	precision	recall	f1-score	support
Blanco	0.83	0.80	0.81	1969
Negro	0.82	0.80	0.81	892
Asiático	0.76	0.84	0.80	708
Indio	0.69	0.73	0.71	835
Otros	0.33	0.30	0.31	338
accuracy			0.76	4742
macro avg	0.69	0.69	0.69	4742
weighted avg	0.76	0.76	0.76	4742



Edad:

Tasa de aciertos de la red en EDAD: 7.001265288907634 %

Error absoluto medio(Edad): 6.655208772669759

Seguimos mejorando los resultados:

- GENERO : muy buen comportamiento con una tasa de aciertos del 90%. Aqui el modelo tiene mayor tasa de aciertos hombres pero es mas exhaustivo con las mujeres, es decir,tiene mejor capacidad de detectar a todas las mujeres posibles
- RAZA: buen comportamiento, aunque no tanto como el GENERO pues tenemos una tasa de aciertos del 69%. Vemos que, comparados con las otras clases, las clases que se le complican son INDIO y OTROS, en especial esta última
- EDAD: tenemos una tasa de acierto de un 7% pero como hemos comentado antes esto no nos dice nada. Vemos entonces que el margen de error es de 6,66 años

Finalmente hemos conseguido mejorar aún más los resultados, pero las mejoras son ínfimas

5. Interpretabilidad (XAI - Explicabilidad de los Modelos)

Hemos utilizado LIME para analizar, en cada modelo y categoría, cuáles son las características más relevantes que influyen en las decisiones que toma cada modelo. Decidimos usar LIME porque es útil para explicaciones XAI en imágenes ya que permite entender qué partes específicas de una imagen influyen más en la decisión del modelo. LIME explica predicciones individuales, mostrando qué regiones de la imagen fueron más importantes para una clasificación concreta. Además, no depende del tipo de modelo, por lo que puede aplicarse tanto a redes convolucionales, como a MLP o k-NN.

La instancia a la que le hemos aplicado LIME es la siguiente imagen:

Edad: 10.0, Género: Hombre, Raza: Otros



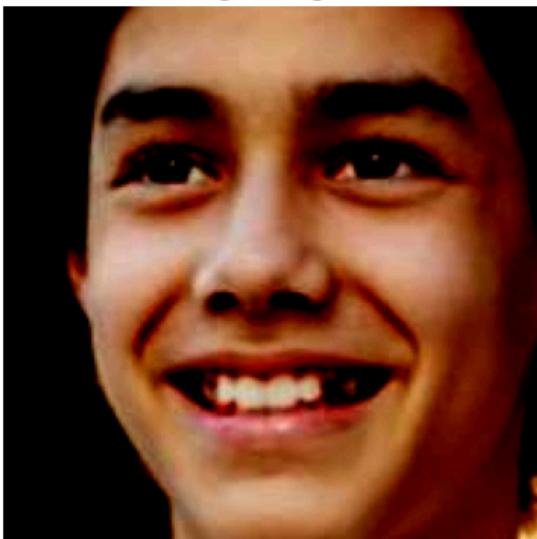
Su edad es 10 años, su género es hombre y su raza es desconocida.

5.1. Modelo KNN-Mbappé

Modelo para raza: la predicción de este modelo para la imagen es “Blanco”. La predicción es incorrecta.

A continuación, adjuntamos la imagen destacando los rasgos más significativos a la hora de realizar la predicción:

Imagen Original



Explicación LIME

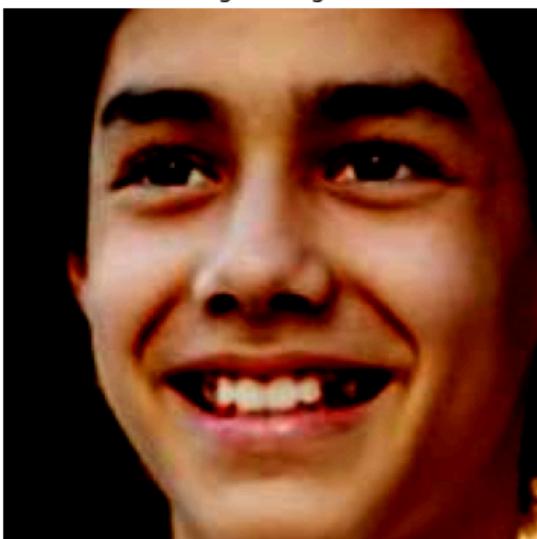


Los rasgos más relevantes para la predicción de la raza "Blanco" en este modelo se encuentran principalmente en el tercio superior del rostro.

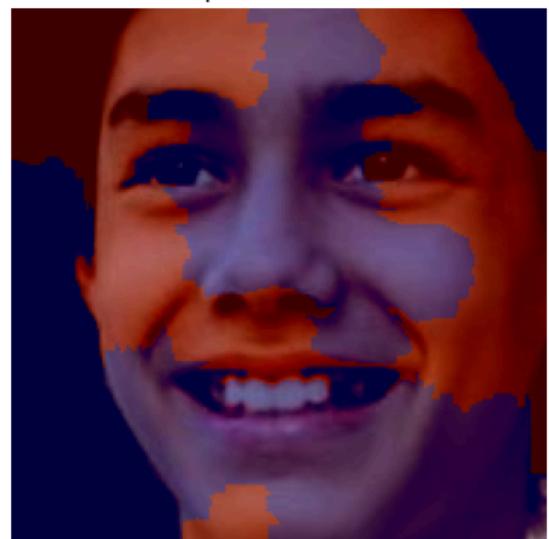
Modelo para la Edad: la predicción de este modelo es la clase 3 . La predicción es incorrecta ya que la imagen pertenece a la clase 1.

A continuación, adjuntamos la imagen destacando los rasgos más significativos a la hora de realizar la predicción:

Imagen Original



Explicación LIME



Los rasgos más relevantes para la predicción del grupo 3 en este modelo se encuentran principalmente en el tercio inferior del rostro y la zona central de la cara.

Modelo para el Genero: la predicción de este modelo es la clase "Mujer" . La predicción es incorrecta.

A continuación, adjuntamos la imagen destacando los rasgos más significativos a la hora de realizar la predicción:

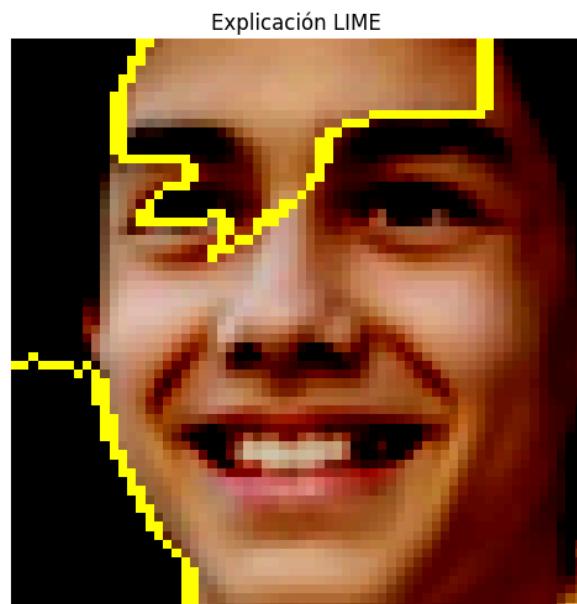
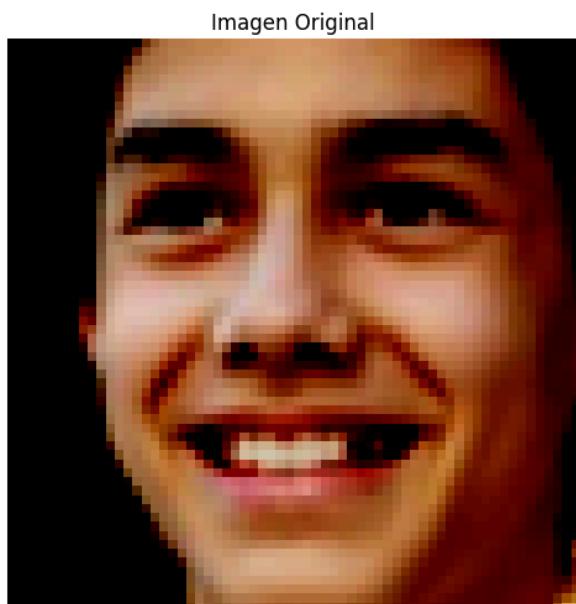


Los rasgos más relevantes para la predicción del género “Mujer” en este modelo se encuentran principalmente en el tercio medio del rostro.

5.2. Modelo MLP-Napoleón

Modelo para raza: la predicción de este modelo para la imagen es “Asiático”. La predicción es incorrecta.

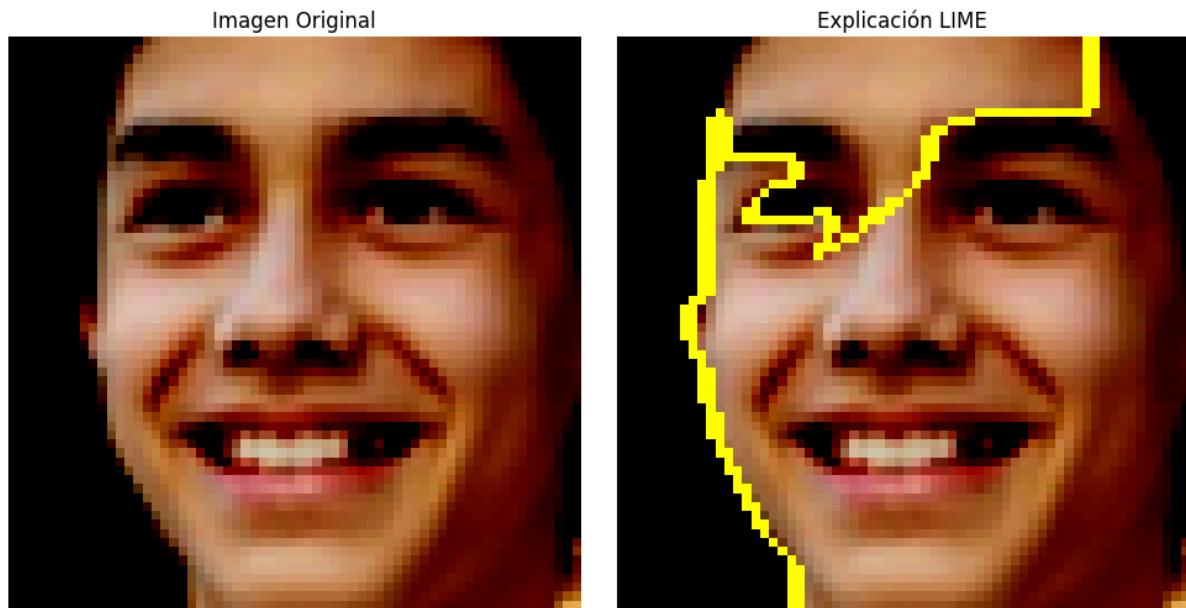
A continuación, adjuntamos la imagen destacando los rasgos más significativos a la hora de realizar la predicción:



Los rasgos más relevantes para la predicción de la raza "Asiático" en este modelo se encuentran principalmente en el tercio superior del rostro.

Modelo para la Edad: la predicción de este modelo es la clase 1 . La predicción es correcta.

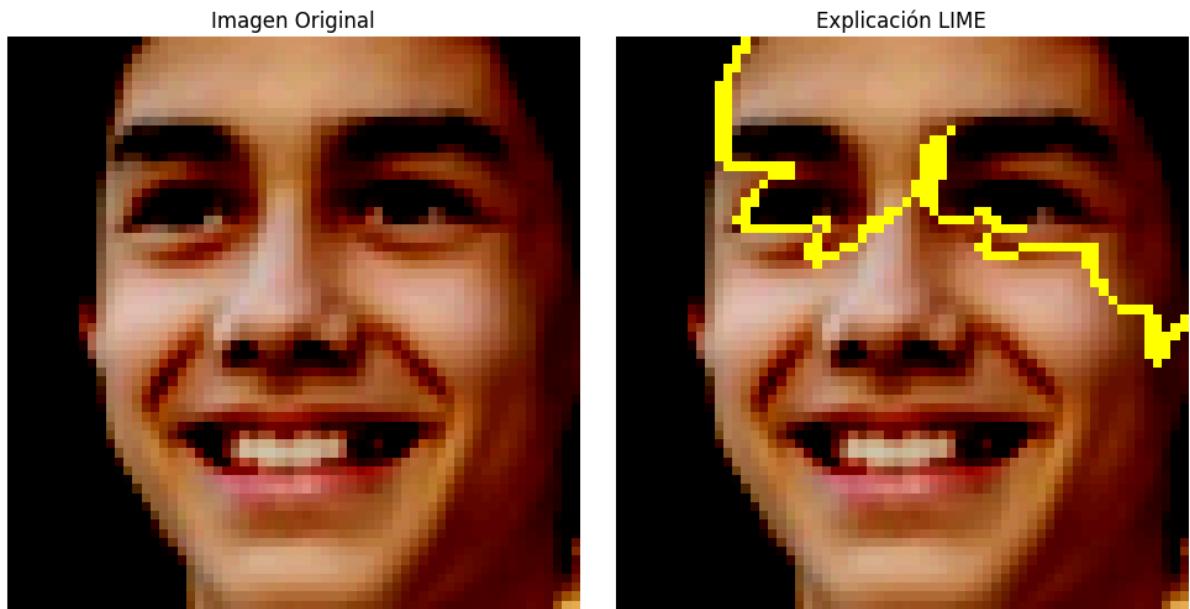
A continuación, adjuntamos la imagen destacando los rasgos más significativos a la hora de realizar la predicción:



Los rasgos más relevantes para la predicción del grupo 3 en este modelo se encuentran principalmente en el tercio superior del rostro junto al contorno de la cara.

Modelo para el Genero: la predicción de este modelo es la clase "Hombre" . La predicción es correcta.

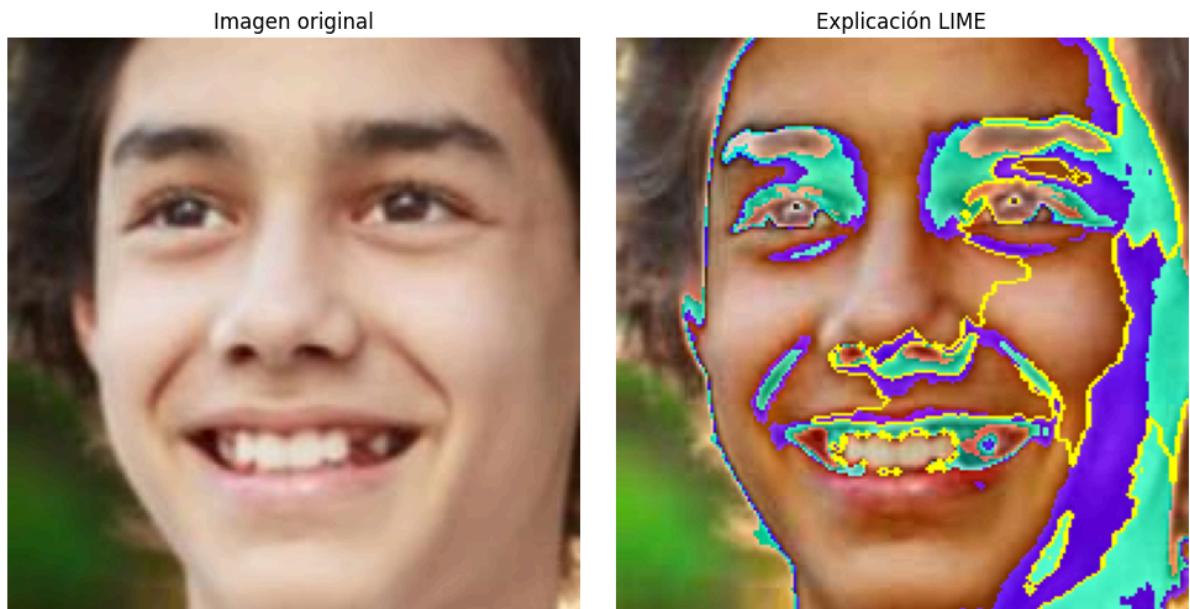
A continuación, adjuntamos la imagen destacando los rasgos más significativos a la hora de realizar la predicción:



Los rasgos más relevantes para la predicción del género “Hombre” en este modelo se encuentran principalmente en el tercio superior del rostro incluyendo la zona de los ojos.

5.3. Modelo CNN- Denzel Washington

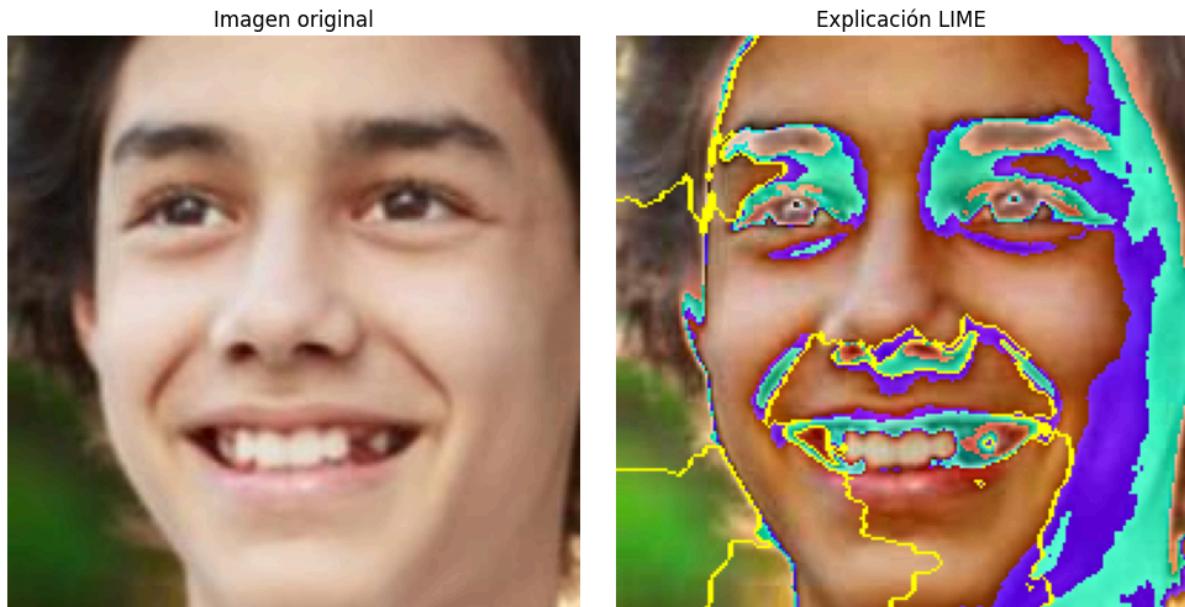
Modelo para el Genero: la predicción de este modelo es la clase “Hombre”. La predicción es correcta.



Los rasgos más relevantes para la predicción del género “Hombre” en este modelo se encuentran principalmente en los ojos y en la forma de la cara.

Modelo para raza: la predicción de este modelo para la imagen es “Otros” . La predicción es correcta.

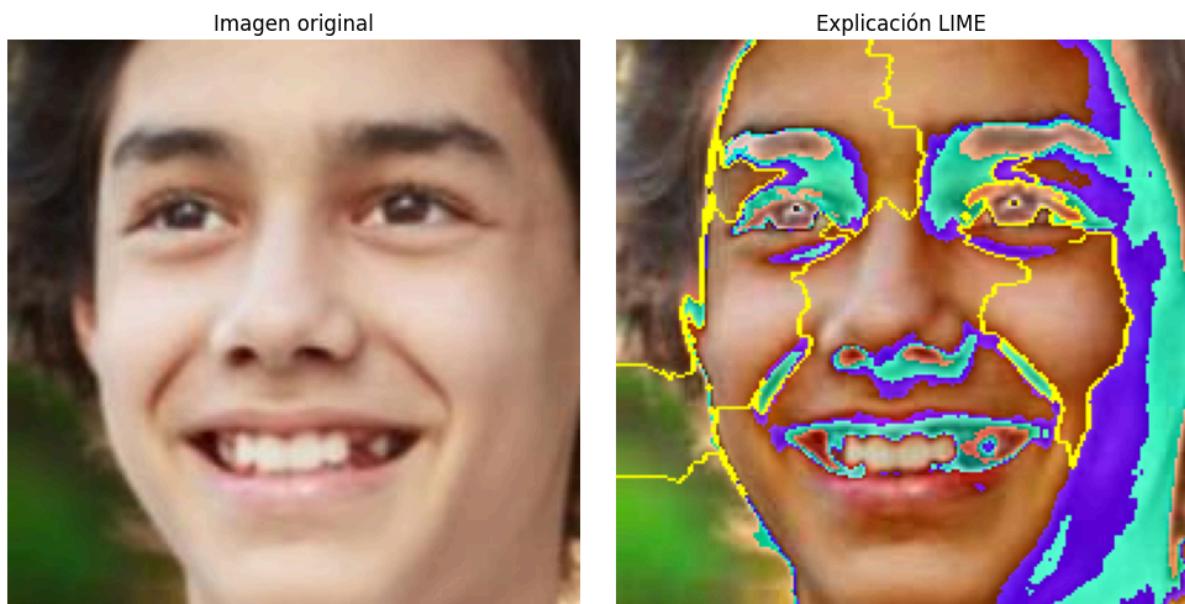
A continuación, adjuntamos la imagen destacando los rasgos más significativos a la hora de realizar la predicción:



Los rasgos más relevantes para la predicción de la raza “Otros” en este modelo se encuentran principalmente en los ojos y en la forma de la cara.

Modelo para la Edad: la predicción de este modelo es 10,68 . La predicción es correcta.

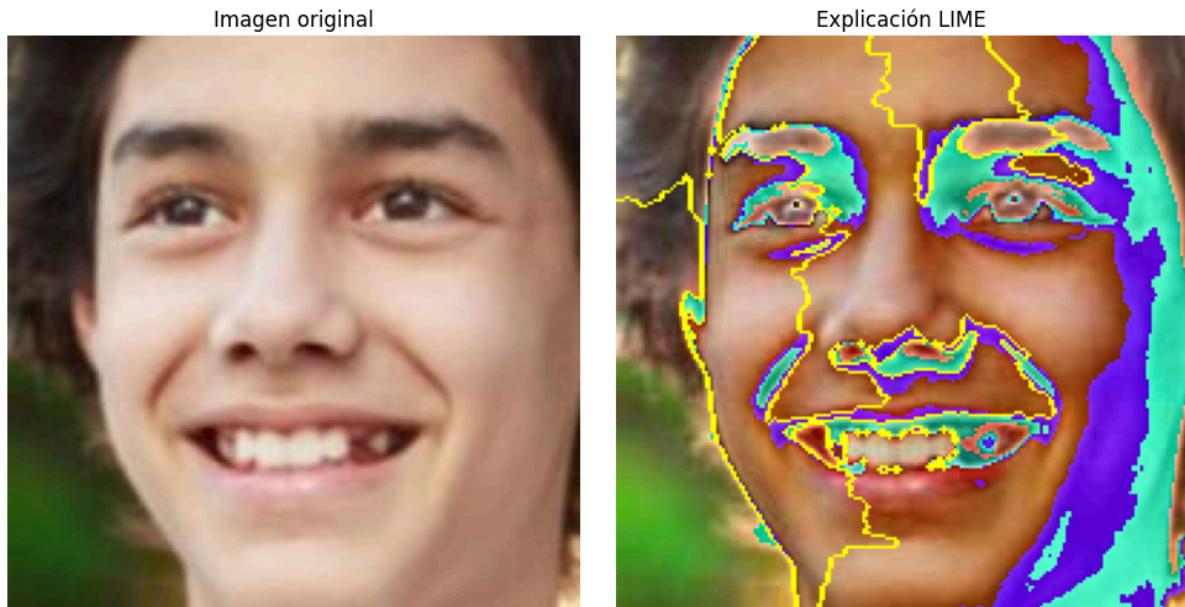
A continuación, adjuntamos la imagen destacando los rasgos más significativos a la hora de realizar la predicción:



Los rasgos más relevantes para la predicción de la edad de 10 años en este modelo se encuentran principalmente en los ojos y en la forma de la cara.

5.4. Modelo CNN- Lola Lolita

Modelo para el Género: la predicción de este modelo es la clase “Hombre” . La predicción es correcta.



Los rasgos más relevantes para la predicción del género “Hombre” en este modelo se encuentran principalmente en los ojos y en la forma de la cara.

Modelo para raza: la predicción de este modelo para la imagen es “Otros” . La predicción es correcta.

A continuación, adjuntamos la imagen destacando los rasgos más significativos a la hora de realizar la predicción:

Imagen original



Explicación LIME



Los rasgos más relevantes para la predicción de la raza “Otros” en este modelo se encuentran principalmente en los ojos y en la forma de la cara.

Modelo para la Edad: la predicción de este modelo es 11,17 . La predicción es incorrecta aunque bastante aproximada..

A continuación, adjuntamos la imagen destacando los rasgos más significativos a la hora de realizar la predicción:

Imagen original



Explicación LIME



Los rasgos más relevantes para la predicción de la edad de 10 años en este modelo se encuentran principalmente en los ojos y en la forma de la cara.

5.5. Conclusiones

Como se puede observar, en los modelos convolucionales (Lola Lolita y Denzel Washington), los rasgos más destacados para la toma de decisiones son prácticamente los mismos en ambos casos, lo que dificulta identificar con claridad qué características son realmente determinantes en cada uno. En cambio, en los modelos MLP y KNN sí se aprecian diferencias más notables: para la clasificación por género, ambos consideran relevantes el tercio medio y superior del rostro; para la raza, destaca el tercio superior; y en cuanto a la edad, el MLP curiosamente se enfoca en el tercio superior, mientras que el KNN da más importancia al tercio medio e inferior.

6.División de tareas

El proyecto entero está compuesto por los siguientes archivos:

- agente_cnn.py: contiene las funciones para los modelos de CNN
- agente_knn.py: contiene las funciones para el modelo de KNN
- agente_mlp.py: contiene las funciones para el modelo MLP
- [data.py](#) : contiene las funciones para el tratamiento de las imagens

y los siguientes notebooks:

- dataset.ipynb
- modelos.ipynb
- modelos_XAI.ipynb

Itziar López Almagro se ha encargado de implementar, entrenar y analizar los modelos de KNN y MLP y en consecuencia de los archivos agente_knn.py y agente_mlp.py

Por otro lado, Adrian Carlos Skaczylo se ha encargado de implementar, entrenar y analizar los dos modelos de CNN y en consecuencia del archivo agente_cnn.py

Además,cada uno se ha encargado de realizar las distintas configuraciones de XIA de sus correspondientes modelos, es decir, 2 configuraciones cada uno.

Por último, los notebooks se han realizado conjuntamente, al igual que de la implementación del archivo data.py y de la memoria.

7. Conclusiones

En general, obtenemos buenos resultados teniendo en cuenta que predecir la raza y la edad de una persona a veces es bastante complicado incluso para una persona. Por ello, estamos bastante contentos con los resultados a pesar de que, obviamente, no son del todo perfectos. Creemos que de los 4 modelos los mejores son los dos últimos: las redes convolucionales. No obstante, la diferencia de comportamiento entre ambos modelos es mínima (se ve claramente en los resultados) mientras que la complejidad de la implementación como la diferencia de tiempo invertida en entrenar ambos es modelos es bastante significativa, lo cual nos hace escoger el 3º modelo como el mejor en términos de resultados-tiempo de cómputo.

Además, creemos que hubiera sido interesante, para cada uno de los modelos probar distintas configuraciones en el tratamiento de las imágenes. Por ejemplo, probar distintos modelos de KNN en los que algunos reciban imágenes RGB y otras en Blanco y Negro, y aplicar esta misma idea a los demás modelos. No solo eso, pensamos también que podría resultar interesante algún modelo de regresión lineal para intentar predecir la edad pues es la categoría con "peores" resultados.

Por otro lado, de cara al futuro y teniendo ya los modelos entrenados, nos surgió la idea de combinarlos con otro modelo de reconocimiento facial en tiempo real, es decir, un modelo que mediante una cámara fuese capaz de detectar/reconocer la cara de la persona y predecir los correspondientes atributos