

# Estimación de la Edad Legal en Antropología Forense Utilizando Técnicas de Deep Learning

## Titulación

Grado en Ingeniería Informática

## Asignatura

Trabajo Fin de Grado - 2023/24  
Convocatoria de Julio

## Autor

José Antonio López Palenzuela

## Directores

Óscar Cordón García  
Pablo Mesejo Santiago



UNIVERSIDAD  
DE GRANADA

---

# ETSIT

Escuela Técnica Superior  
de Ingenierías Informática  
y de Telecomunicación



# Índice

- Introducción.
- Estado del Arte.
- Materiales y Métodos.
- Experimentación.
- Resultados.

# INTRODUCCIÓN

# Motivación I. Estimación de la Edad Legal.

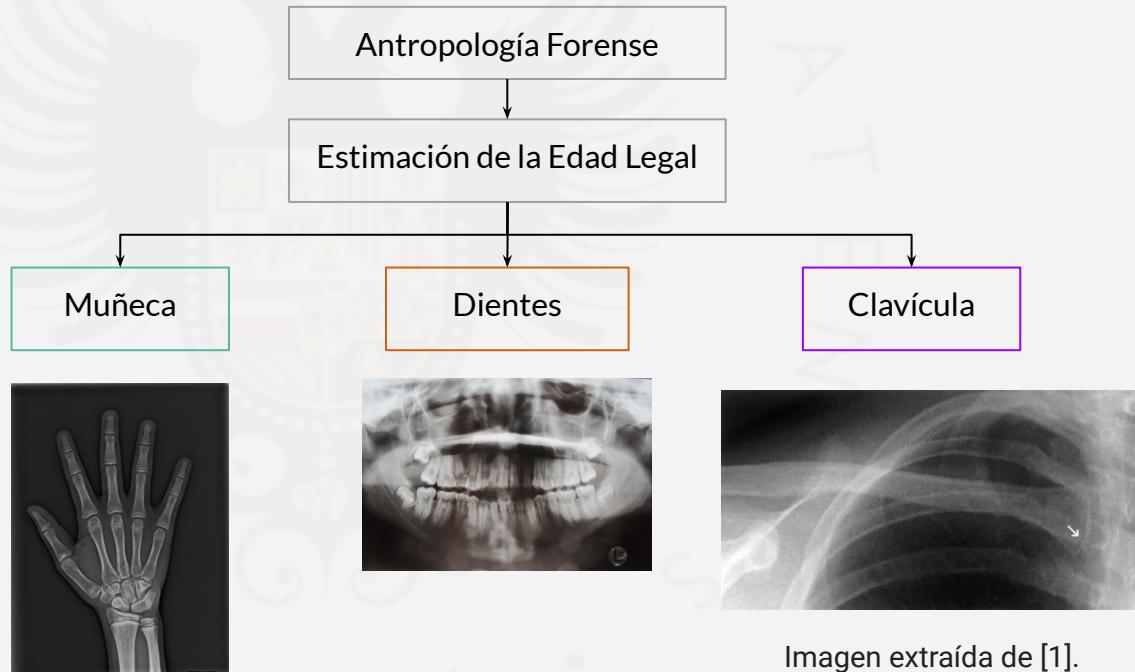
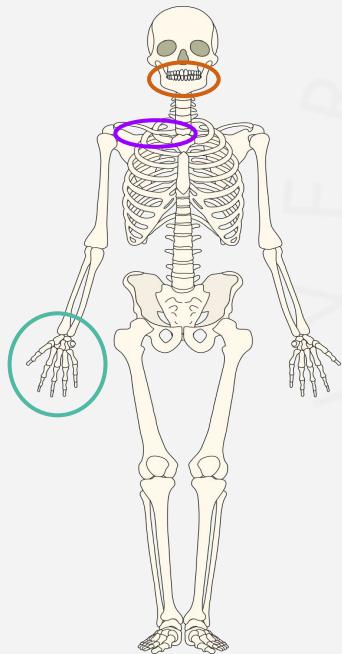
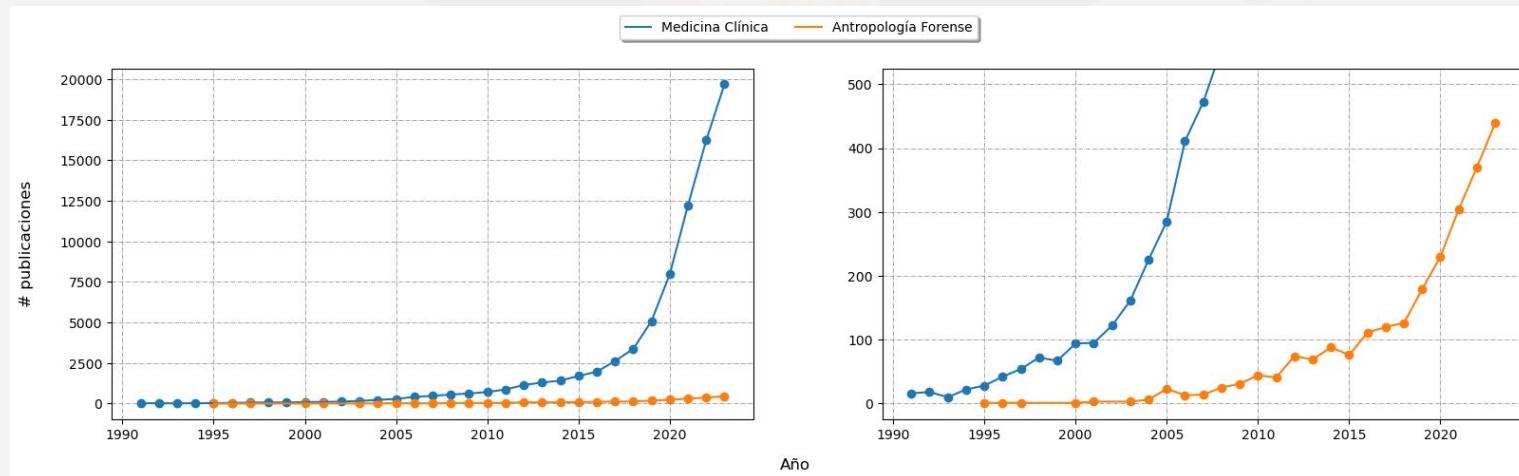


Imagen extraída de [1].

# Motivación II. Uso ML clásico y DL.

- Praxis habitual **manual, imprecisa y subjetiva**.
- La antropología forense presenta un **limitado desarrollo tecnológico**.



Búsqueda en Scopus.

Naranja: (forensic anthropology) AND (machine learning OR deep learning)  
Azul: (clinic medicine) AND (machine learning OR deep learning)

# Objetivo Principal.

Implementar un **método automatizado, basado en técnicas de ML clásico y DL**, para **determinar la edad legal de un individuo a partir de una radiografía panorámica de la boca (ortopantomografía u OPG)**.



14 años y 7 meses



20 años y 3 meses

# ESTADO DEL ARTE

# Estado del Arte. Métodos Automatizados.

- Estudios: Guo et al. (2021) y Murray et al. (2024).
- Mismo problema, “mismos” métodos pero diferentes materiales .
  - **OPG de individuos de distintas poblacionales: chinos y brasileños.**
  - Uso de modelos DL y técnicas de experimentación similares a los empleados en este proyecto.
  - Superan (chinos) o rivalizan (brasileños) con los resultados del Estado del Arte.

[2] Murray et al. (2024). Applying artificial intelligence to determination of legal age of majority from radiographic. Morphologie 108.360 (2024): 100723.

[3] Guo et al. (2021). Accurate age classification using manual method and deep convolutional neural network based on orthopantomogram images. International journal of legal medicine, 135, 1589-1597.

# Estado del Arte. Expertos Humanos.

- Estudio: Angelakopoulos et al. (2018).
- Test de Cameriere.
  - **Tercer Molar = Muela del Juicio.**
  - $(a+b)/c < 0.08 \rightarrow$  Menor
  - $(a+b)/c \geq 0.08 \rightarrow$  Mayor
- Mismo problema, “mismos” materiales y diferente método.
  - **OPG de individuo de la misma población (sudafricanos negros).**

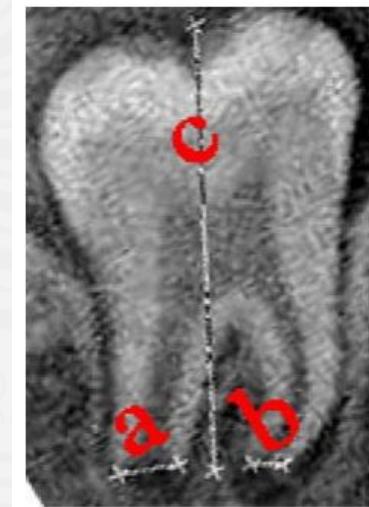


Imagen extraída de [5].

[4] Angelakopoulos et al. (2018). Third molar maturity index ( $I_3M$ ) for assessing age of majority: study of a black South African sample. International journal of legal medicine, 132, 1457-1464.

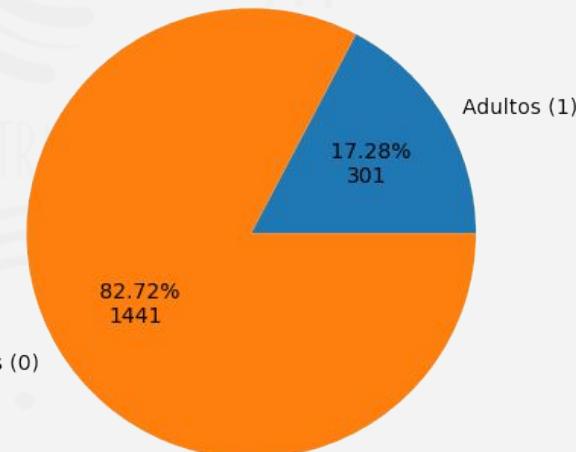
[5] Cameriere et al. (2008). The comparison between measurement of open apices of third molars and Demirjian stages to test chronological age of over 18 year olds in living subjects. International journal of legal medicine, 122, 493-497.

# MATERIALES Y MÉTODOS

# Datos. Exploración Inicial.

- Imágenes facilitadas por **Stefano De Luca** ([Panacea](#)) y **Javier Irurita Olivares** (UGR).
- Población de origen: **sudafricanos negros**.
- Conjunto de datos gravemente **DESBALANCEADO**.
- Test → 20 % balanceado.
- Validación cruzada ( $k=5$ ) → 80 % desbalanceado.

N = 1742



# Candidatos Propuestos.

- Se presentaron dos tipos de soluciones:
  - 2 candidatos basados en DL (ResNet [6]).
    - **ResNet18**.
    - **ResNet50**.
  - 3 candidatos basados en técnicas clásicas ML (HOG [7] + Clasificador).
    - **HOG + LR** (Regresión Logística).
    - **HOG + SVM** (Support Vector Machine).
    - **HOG + RF** (Random Forest).
- Diferencia fundamental: extracción de características manual o automática.

[6] Kaiming et al. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

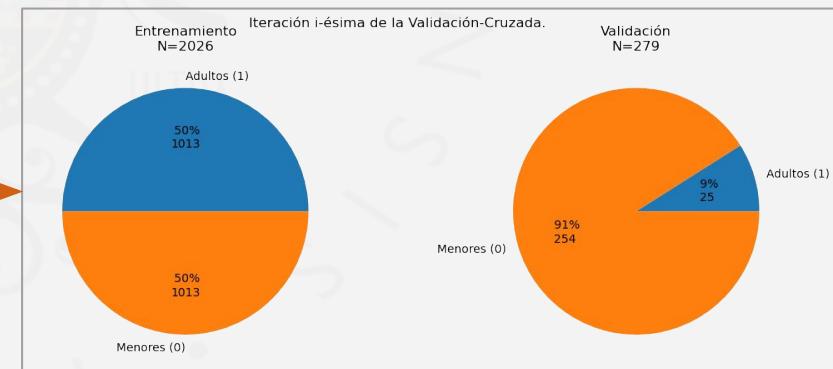
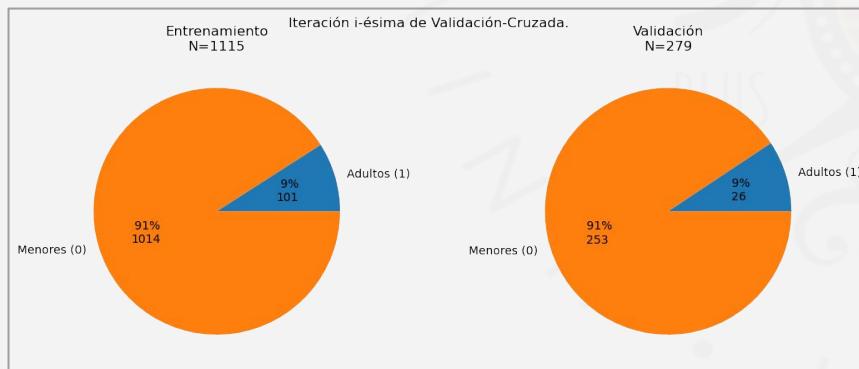
[7] Navneet Dalal & Bill Triggs (2005). Histograms of oriented gradients for human detection. In 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05) (Vol. 1, pp. 886-893). ieee.

# Desbalanceo.

- “Un modelo es tan bueno como lo son los datos con los que entrena”.
- El grave desbalance de los datos de entrenamiento provocaba que **todos los candidatos propuestos interiorizaran el sesgo de los datos**: todos aprendían a predecir siempre la clase mayoritaria.
- Para solucionarlo, se probaron dos técnicas habituales en este contexto:
  - **Sobremuestreo**.
  - **Aumento de Datos**.

# Desbalanceo. Sobre muestreo.

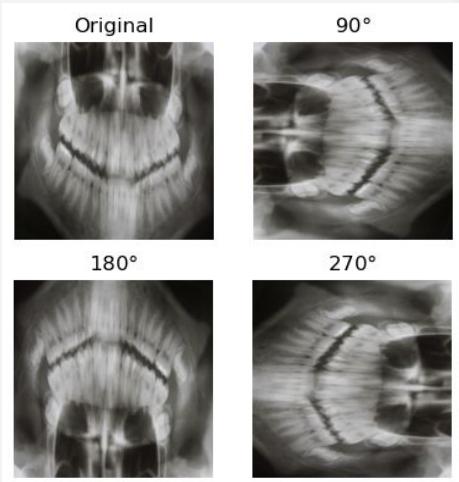
- **Duplicar imágenes de la clase minoritaria** hasta que el número de imágenes de ambas clases sea el mismo.
- Maximizar diversidad de las imágenes duplicadas.
- **Aplicado para ML clásico y DL** sobre el conjunto de entrenamiento de cada iteración de la validación cruzada.



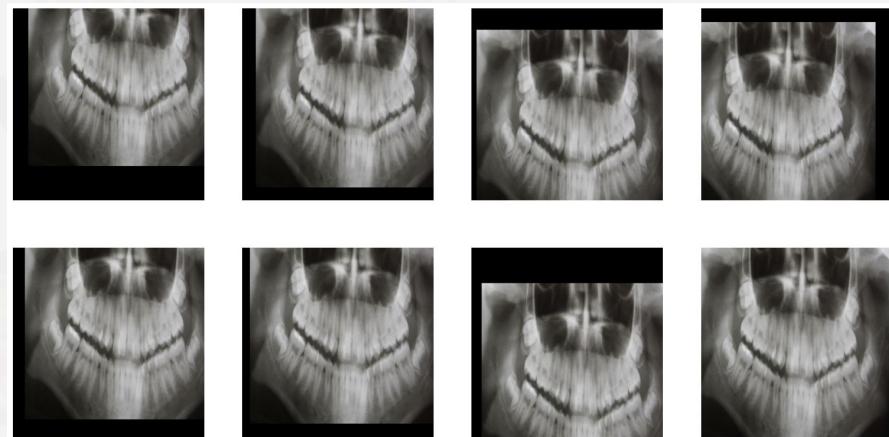
# Desbalanceo. Aumento de Datos.

- Generar nuevas instancias (imágenes y etiquetas) para entrenar fruto de transformar instancias pre-existentes del conjunto de entrenamiento.
- Aumentar la diversidad de las imágenes duplicadas vía sobremuestreo.
- Intuitivas: **Rotaciones** ( $90^\circ$ ,  $80^\circ$  y  $270^\circ$ ) y **traslaciones** (20%).
- Contra-intuitivas: **MixUp** [8].
- Pueden **combinarse**.
- **Aplicado únicamente a modelos de DL.**

## Rotaciones



## Traslaciones



## MixUp

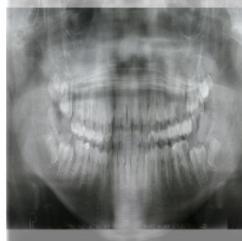
$\lambda_{MU} = 0.0$ .  
Etiqueta = [0.0, 1.0]



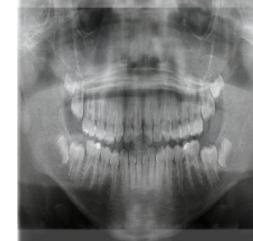
$\lambda_{MU} = 0.25$ .  
Etiqueta = [0.25, 0.75]



$\lambda_{MU} = 0.5$ .  
Etiqueta = [0.5, 0.5]



$\lambda_{MU} = 0.75$ .  
Etiqueta = [0.75, 0.25]



$\lambda_{MU} = 1.0$ .  
Etiqueta = [1.0, 0.0]



# GradCAM

- Modelos “caja negra”.
- Aplicable a **candidatos DL** (CNN).
- Genera un mapa de activación que **señala las regiones que más influyen** en la predicción del modelo.

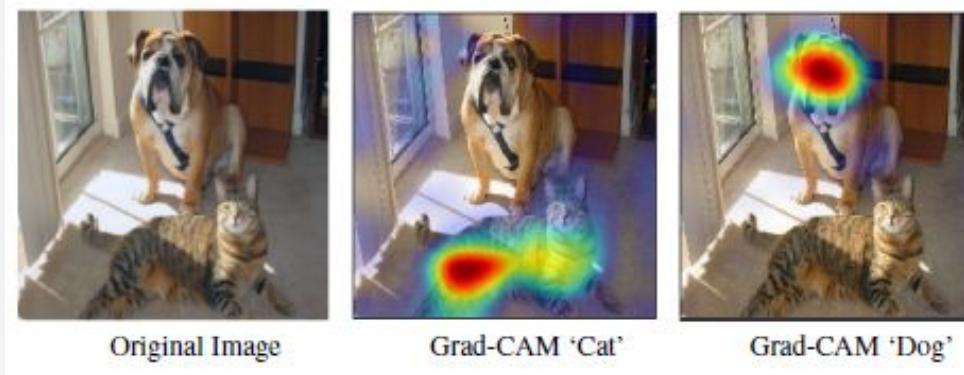


Imagen extraída de [9].

[9] Selvaraju et al. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE international conference on computer vision (pp. 618-626).

# **EXPERIMENTACIÓN**

# Experimentación. Estructura Global.

1. Encontrar la mejor combinación de hiperparámetros para cada candidato propuesto vía validación cruzada ( $k=5$ ): **ResNet18, ResNet50, LR, SVM** y **RF**.
2. Evaluar el rendimiento de los mejores candidatos sobre test (20% de los datos, balanceado).
3. Determinar mejor clasificador implementado en base a los resultados test.
4. Comparar con el Estado del Arte.

# Experimentación. Métricas de Rendimiento.

$P \rightarrow$  Imagen predicha como mayor de edad (positiva = 1).

$N \rightarrow$  Imagen predicha como menor de edad (negativa = 0).

$$ACC = \frac{TP + TN}{TP + FP + TN + FN}$$

$$SEN = \frac{TP}{TP + FN}$$

$$SPE = \frac{TN}{TN + FP}$$

# Experimentación. DL.

En el caso de los modelos DL (ResNet18 y ResNet50) se experimentó para determinar:

- Si el **sobremuestreo** era efectivo contra el desbalanceo.
- Si el **aumento de datos** era efectivo contra el desbalanceo y qué técnica aportaba mejores resultados ofrecía (así como los hiperparámetros de esta).
- Si **Fine-Tuning** aportaba algún tipo de mejora.

Todos los modelos entrenaron con los siguientes hiperparámetros:

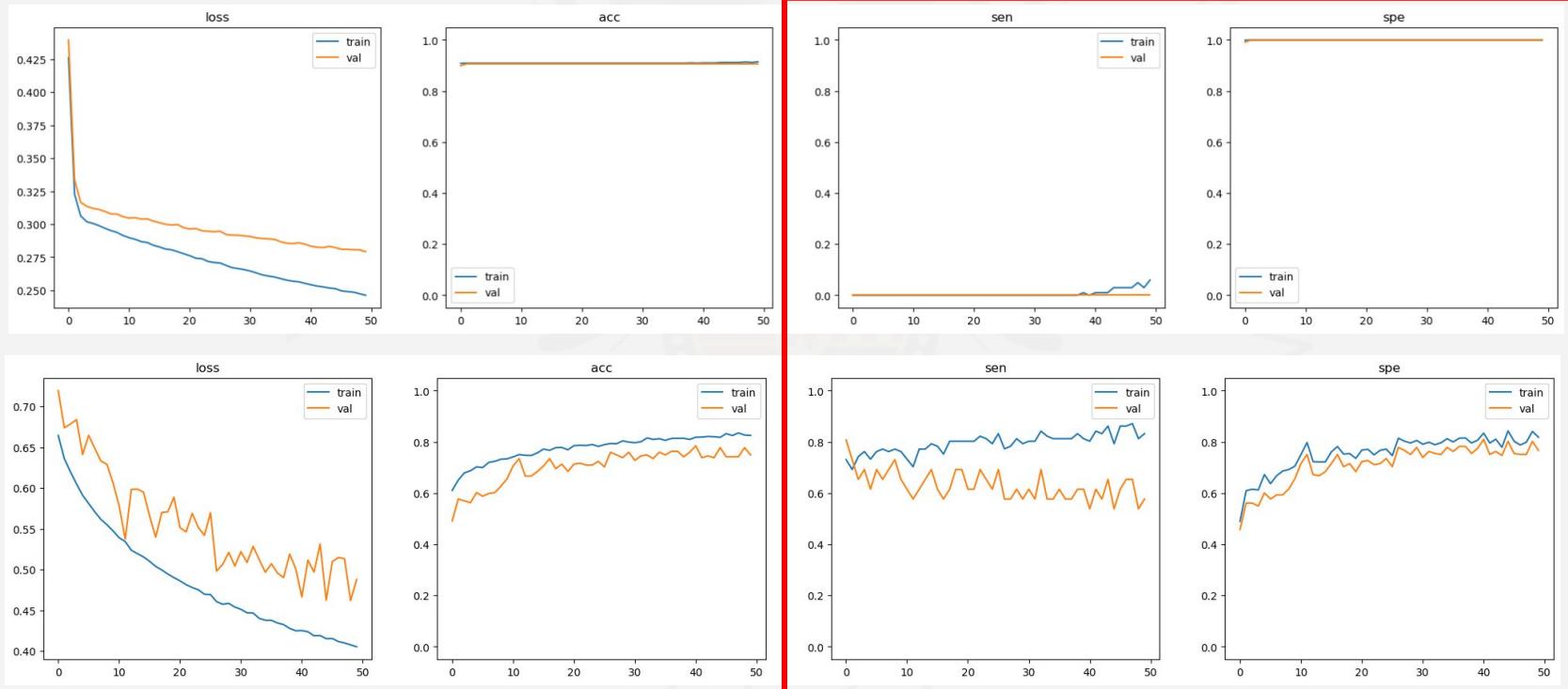
- Función Pérdida = Entropía Cruzada.
- Tamaño de Batch = 32.
- Preprocesado = Normalización con métricas de ImageNet y re-escalado  $3 \cdot 224 \cdot 224$ .
- **LRFind** [10] para determinar el valor de la tasa de aprendizaje ( $\lambda = 1 \cdot 10^{-4}$ , en la mayoría de casos).
- Se guarda el mejor modelo.

# Experimentación. DL. Sobre|muestreo.

- Tanto para ResNet18 como para ResNet50, el **sobre|muestreo evitaba que el modelo interiorizarse el sesgo de los datos** y proporcionaba una **notable mejora de rendimiento**. Obsérvese la mejora para ResNet18.

Modelo	$\alpha_M$	Loss	ACC	SEN	SPE	$M_{CV}$
Desbalanceado	0.4	0.3400	0.9103	0.0157	1.0000	0.9843
	0.5	0.3400	0.9103	0.0157	1.0000	0.9843
	0.6	0.3400	0.9103	0.0157	1.0000	0.9843
Balanceado	0.4	0.5013	0.7762	0.6852	0.7853	0.1001
	0.5	0.5068	0.7704	0.6929	0.7782	0.0853
	0.6	0.5330	0.7489	0.7169	0.7522	0.0353

# Experimentación. DL. Sobre|muestreo.



# Experimentación. DL. ResNet18. Aumento de Datos.

- La técnica de aumento de datos que mejores resultados ofreció fue aplicar rotaciones o traslaciones a un 50% de las imágenes con las que trabaja el modelo.

DA	$\alpha_M$	<i>Loss</i>	<i>ACC</i>	<i>SEN</i>	<i>SPE</i>	<i>MCV</i>	<i>O<sub>SEN</sub></i>
Ninguna (6.1)	0.6	0.5330	0.7489	0.7169	0.7522	0.0353	0.1197
R (6.3)	0.5	0.5423	0.7446	0.7326	0.7458	0.0132	0.0061
T (6.4)	0.5	0.5344	0.7382	0.7483	0.7372	0.0111	0.0565
R y T (6.5)	0.5	0.5396	0.7475	0.7486	0.7475	0.0011	0.0111
<i>MixUp</i> (6.6)	0.6	0.5499	0.7310	0.7329	0.7309	0.0020	0.1025
M* y MU (6.6)	0.5	0.5447	0.7417	0.7403	0.7419	0.0016	0.0281

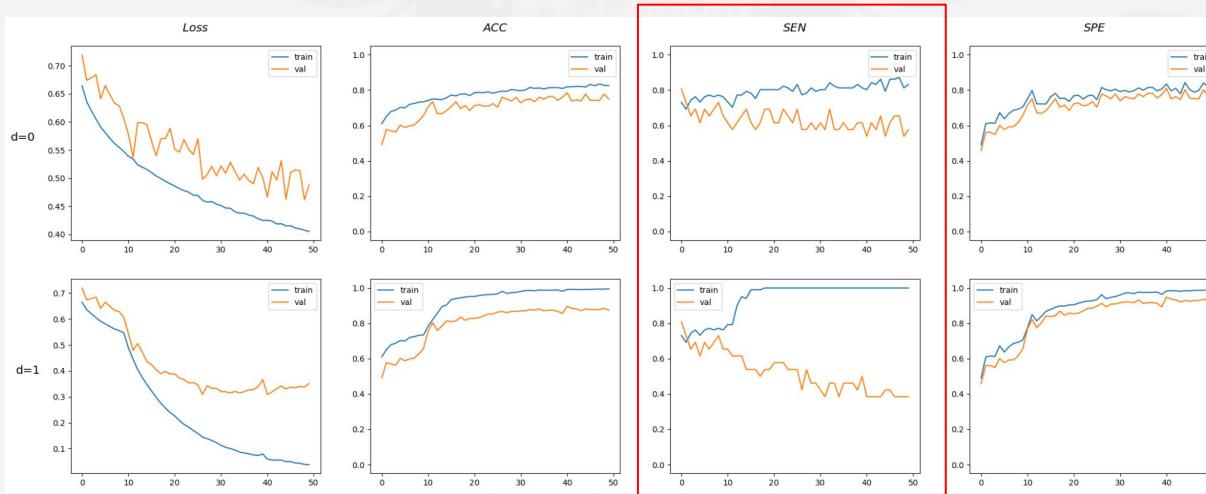
# Experimentación. DL. ResNet50. Aumento de Datos.

- La técnica de aumento de datos que mejores resultados ofreció fue aplicar: primero MixUp con  $\alpha_{MU} = 0.1$  y después, rotar o trasladar un 50% de las imágenes.

DA	$\alpha_M$	Loss	ACC	SEN	SPE	$M_{CV}$	$O_{SEN}$
Ninguna (8.4)	0.5	0.4883	0.7927	0.6852	0.8035	0.1182	0.2142
R (8.5)	0.6	0.5407	0.7590	0.7640	0.7585	0.0055	0.0857
T (8.6)	0.6	0.5098	0.7805	0.7403	0.7846	0.0442	0.0677
R or T (8.7)	0.5	0.5412	0.7611	0.7637	0.7608	0.0029	0.0944
R and T (8.8)	0.6	0.5430	0.7647	0.7560	0.7656	0.0096	0.0453
<i>MixUp</i> (8.9)	0.6	0.5371	0.7554	0.7557	0.7553	0.0004	0.1550
<i>MixUp + M*</i> (8.10)	0.5	0.5400	0.7640	0.7637	0.7640	0.0003	0.0899
<i>M* + MixUp</i> (8.11)	0.5	0.5733	0.7482	0.7486	0.7482	0.0004	0.0183

# Experimentación. DL. Fine-Tuning.

- **Fine-Tuning no ofreció una mejora** para ninguno de los modelos (aplicado sobre el mejor candidato resultante del aumento de datos).
- **Se disparaba el sobreajuste para la clase antes minoritaria** desde que se descongelaba la primera capa (época 5). Obsérvese para ResNet18.



# Experimentación. ML Clásico.

- Experimentos **poco costosos** computacionalmente (en comparación con DL).
- Filosofía **GridSearch** para cada clasificador → Definir todos los valores posibles para cada hiperparámetro y probar todas las combinaciones de estos.
- El **sobremuestreo** era **beneficioso** para todos los clasificadores.
- No se experimentó con la extracción de características → **HOG con hiperparámetros por defecto.**

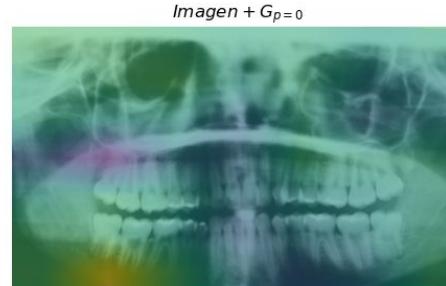
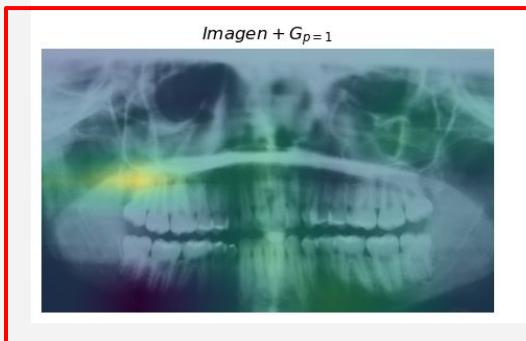
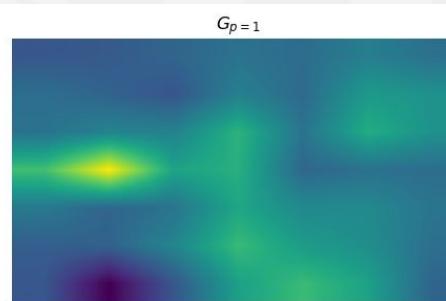
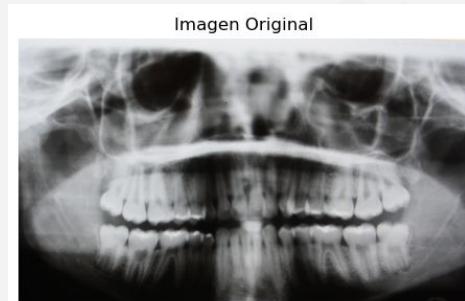
# Experimentación. Comparativa Global.

## Test

Modelo	<i>ACC</i>	<i>SEN</i>	<i>SPE</i>
<i>ResNet18</i>	0.7241	0.7126	0.7356
<i>ResNet50</i>	0.7414	0.7644	0.7184
<i>LR</i>	0.6695	0.5287	0.8103
<i>SVM</i>	0.6580	0.4828	0.8333
<i>RF</i>	0.6379	0.5345	0.7414
<i>Dummy</i>	0.4626	0.4828	0.4425
<i>Humano</i>	0.9004	0.8080	0.9588
<i>C</i>	0.9230	0.9260	0.9190
<i>B</i>	0.8750	0.8700	0.8790

- **Dummy** = Clasificador aleatorio.
- **Humano** = Angelakopoulos et al. (2018).
  - Experto humano.
  - Misma población.
- **C** = Guo et al. (2021).
  - Método automático.
  - Distinta Población → Chinos.
- **B** = Murray et al. (2024).
  - Método automático.
  - Distinta Población → Brasileños.

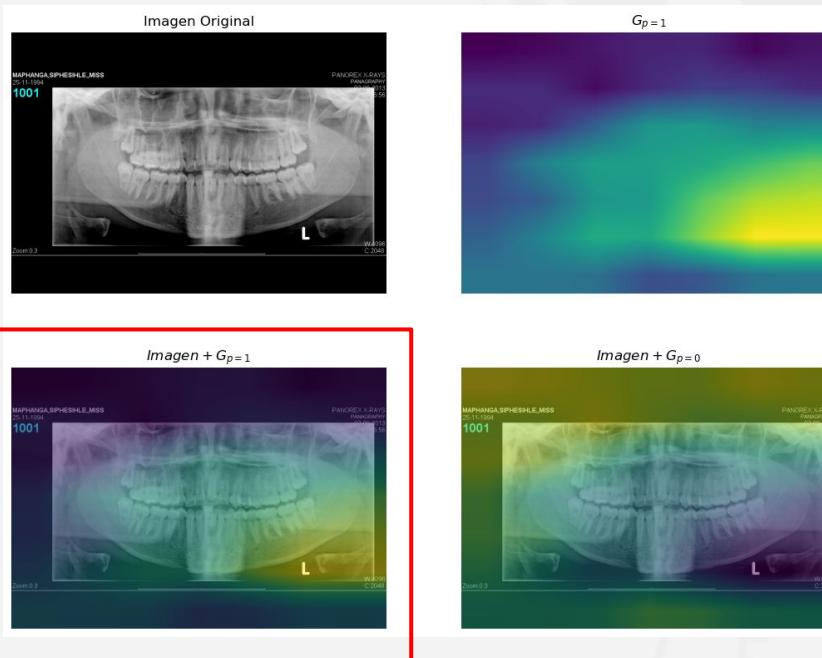
# Experimentación. Explicabilidad. Caso Prometedor.



- Edad = 19.49 años.
- Predicción = 1  
( $p=0.73$ ).

Se centra en el tercer molar superior izquierdo para predecir la mayoría de edad.

# Experimentación. Explicabilidad. Caso Problemático.



- Edad = 20.24 años.
- Predicción = 1  
( $p=0.97$ ).

Nuevamente, la región de la imagen que más influye a la predicción del individuo corresponde a una anotación externa en la radiografía. → **¿possible sesgo en los datos?**

# CONCLUSIONES

# Conclusiones.

- Método **automatizado**, basado en **DL**, para **estimar la edad legal** a través de una **OPG**.
  - CNN basada **ResNet50**.
  - Predicciones **instantáneas, reproducibles** y **explicables**.
  - ACC = 74.14 %, SEN = 76.84 % y SPE = 71.84 % sobre **sudafricanos negros**.
- El **DL** ofrece mejores resultados que el **ML clásico**.
- El **desbalanceo de los datos**, la falta de OPG de mayores de edad, es el factor más limitante del rendimiento.
- **Sobremuestreo** y **aumento de datos** son técnicas **efectivas** contra el desbalanceo pero **insuficientes** para alcanzar el Estado del Arte.

# Trabajos Futuros.

- Conjunto de imágenes mayor y/o más balanceado.
- Regresión + umbral en lugar de clasificación.
- Preprocesado más complejo para evitar el sesgo identificado vía GradCAM.

# Estimación de la Edad Legal en Antropología Forense Utilizando Técnicas de Deep Learning

## Titulación

Grado en Ingeniería Informática

## Asignatura

Trabajo Fin de Grado - 2023/24  
Convocatoria de Julio

## Autor

José Antonio López Palenzuela

## Directores

Óscar Cordón García  
Pablo Mesejo Santiago



UNIVERSIDAD  
DE GRANADA

---

# ETSIT

Escuela Técnica Superior  
de Ingenierías Informática  
y de Telecomunicación



# Motivación III. Estimación de la Edad Legal.

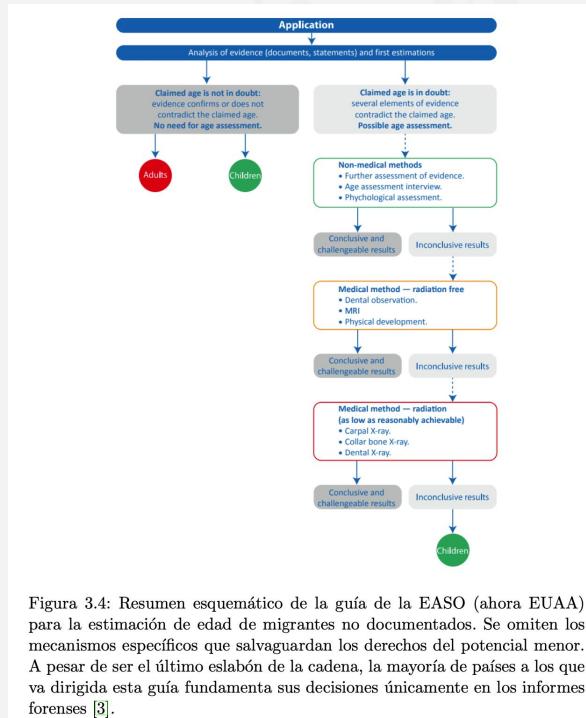


Figura 3.4: Resumen esquemático de la guía de la EUAS (ahora EUAA) para la estimación de edad de migrantes no documentados. Se omiten los mecanismos específicos que salvaguardan los derechos del potencial menor. A pesar de ser el último eslabón de la cadena, la mayoría de países a los que va dirigida esta guía fundamenta sus decisiones únicamente en los informes forenses [3].

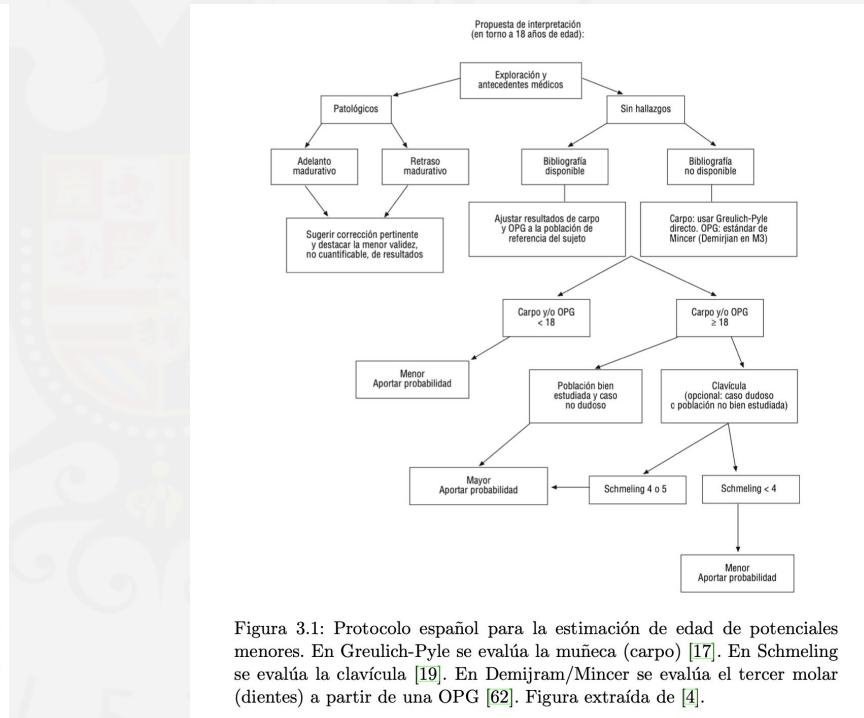
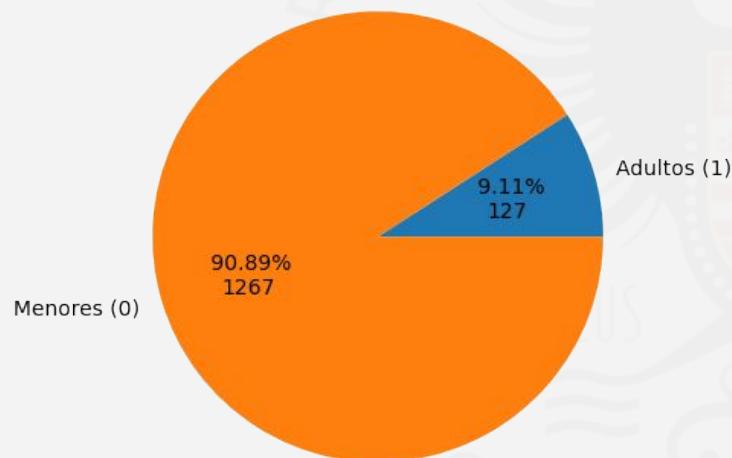


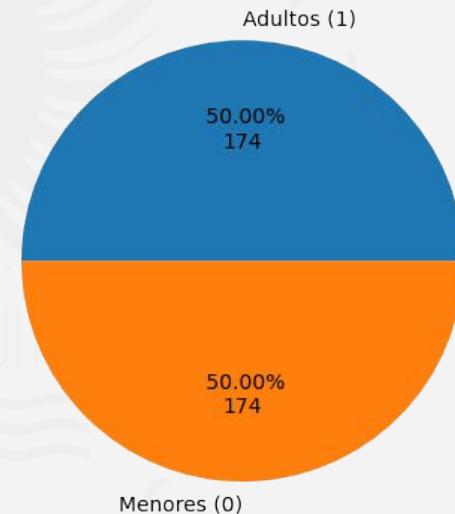
Figura 3.1: Protocolo español para la estimación de edad de potenciales menores. En Greulich-Pyle se evalúa la muñeca (carpo) [17]. En Schmeling se evalúa la clavícula [19]. En Demirjam/Mincer se evalúa el tercer molar (dientes) a partir de una OPG [62]. Figura extraída de [4].

# Datos. Partición Entrenamiento-Validación y Test.

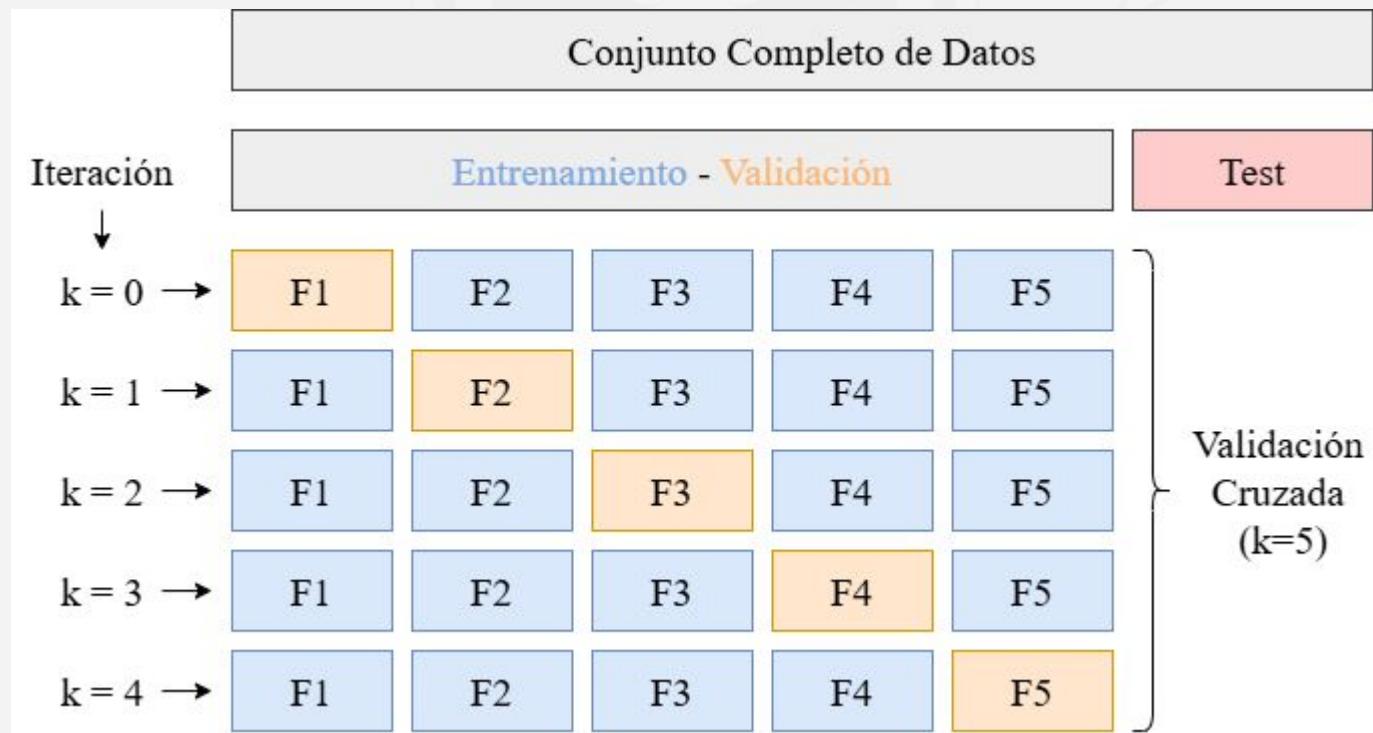
Conjunto Entrenamiento-Validation.



Conjunto de Test.



# Datos. Validación Cruzada.



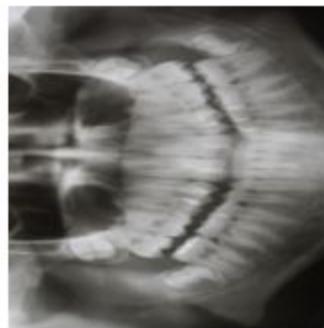
# Desbalanceo. Aumento de Datos. Rotaciones.

- Se genera una nueva imagen rotando la original, la etiqueta se mantiene.
- Tres hiperparametros: clases a las que se aplica (**I**), frecuencia con la que se aplica (**p**) y ángulo de giro (**90°, 180° o 270°**).

Original



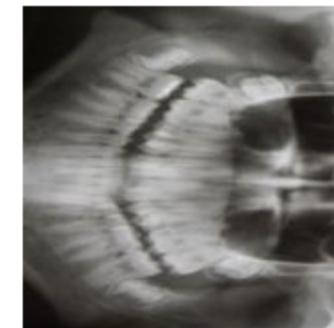
90°



180°

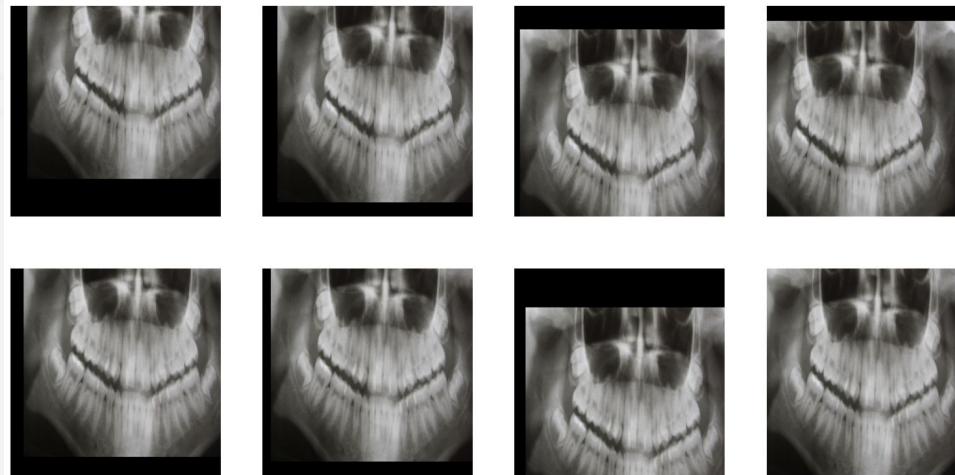


270°



# Desbalanceo. Aumento de Datos. Traslaciones.

- Se genera una nueva imagen trasladando la original, la etiqueta se mantiene.
- Tres hiperparametros: clases a las que se aplica (**I**), frecuencia con la que se aplica (**p**) y rango máximo de desplazamiento (**20% del tamaño original de la imagen**).



# Desbalanceo. Aumento de Datos. MixUp I.

- Se genera una nueva imagen y etiqueta, resultado de combinar linealmente dos imágenes y etiquetas en base a un factor  $\lambda_{MU}$  que se extrae de aleatoriamente de una “distribución beta” con  $\alpha = \beta = \alpha_{MU}$ .

$\lambda_{MU} = 0.0$ .  
Etiqueta = [0.0, 1.0]



$\lambda_{MU} = 0.25$ .  
Etiqueta = [0.25, 0.75]



$\lambda_{MU} = 0.5$ .  
Etiqueta = [0.5, 0.5]



$\lambda_{MU} = 0.75$ .  
Etiqueta = [0.75, 0.25]

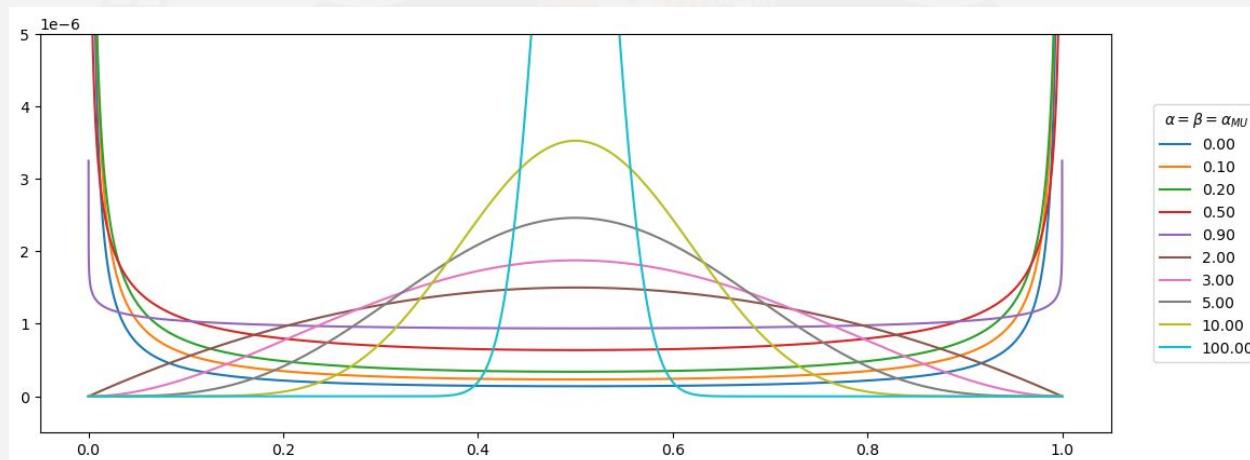


$\lambda_{MU} = 1.0$ .  
Etiqueta = [1.0, 0.0]



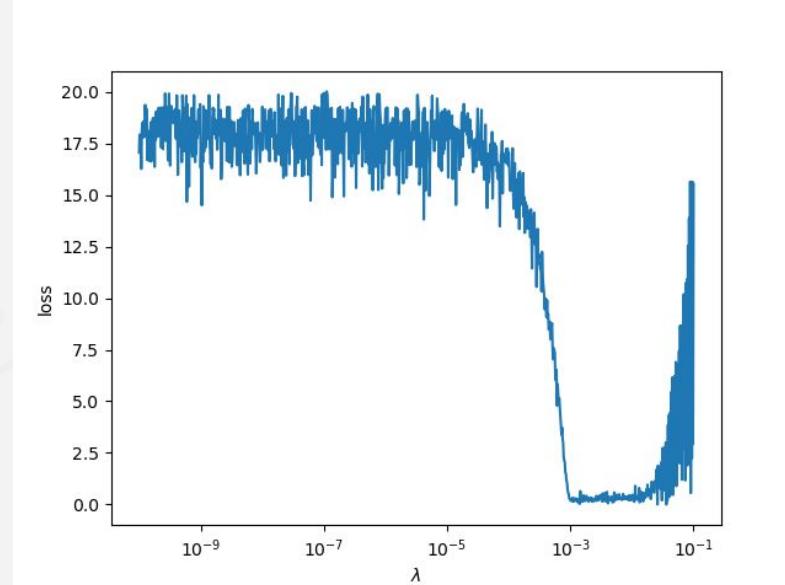
# Desbalanceo. Aumento de Datos. MixUp II.

- Se genera una nueva imagen y etiqueta, resultado de combinar linealmente dos imágenes y etiquetas en base a un factor  $\lambda_{MU}$  que se extrae de aleatoriamente de una “distribución beta” con  $\alpha = \beta = \alpha_{MU}$ .

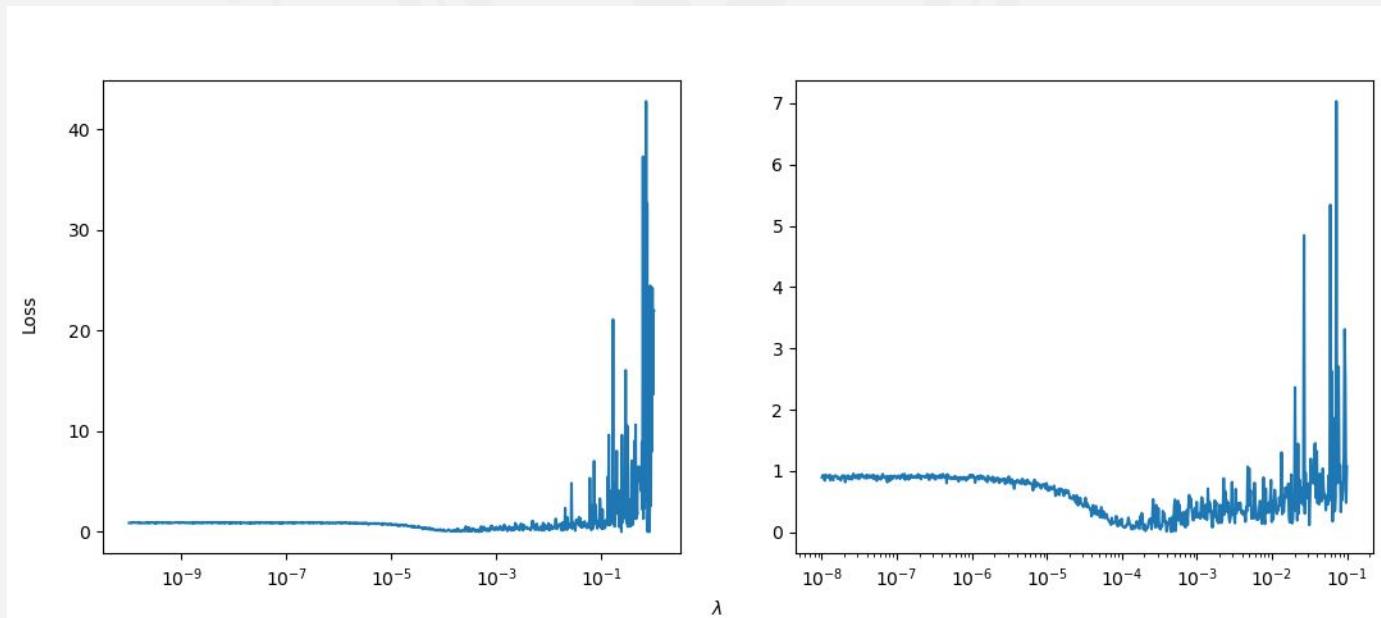


# LR-Find. Teoría.

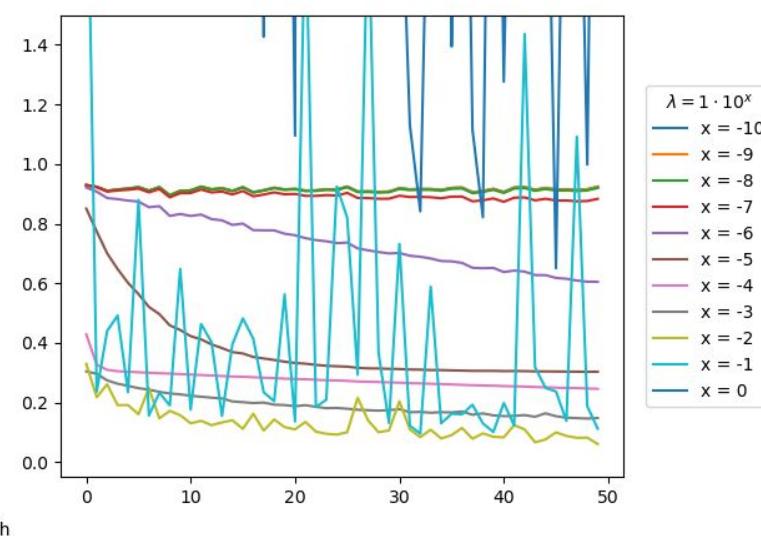
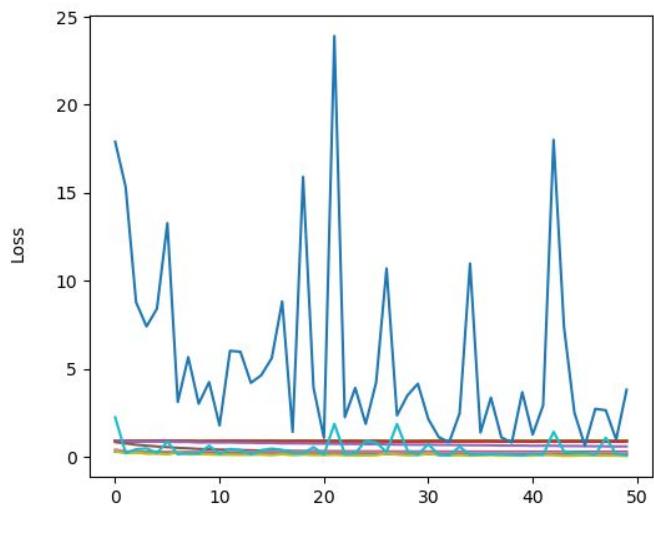
- Técnica para determinar una tasa de aprendizaje adecuada para un modelo.
- Utilizada para los modelos de DL.



# LR-Find. Práctica I.

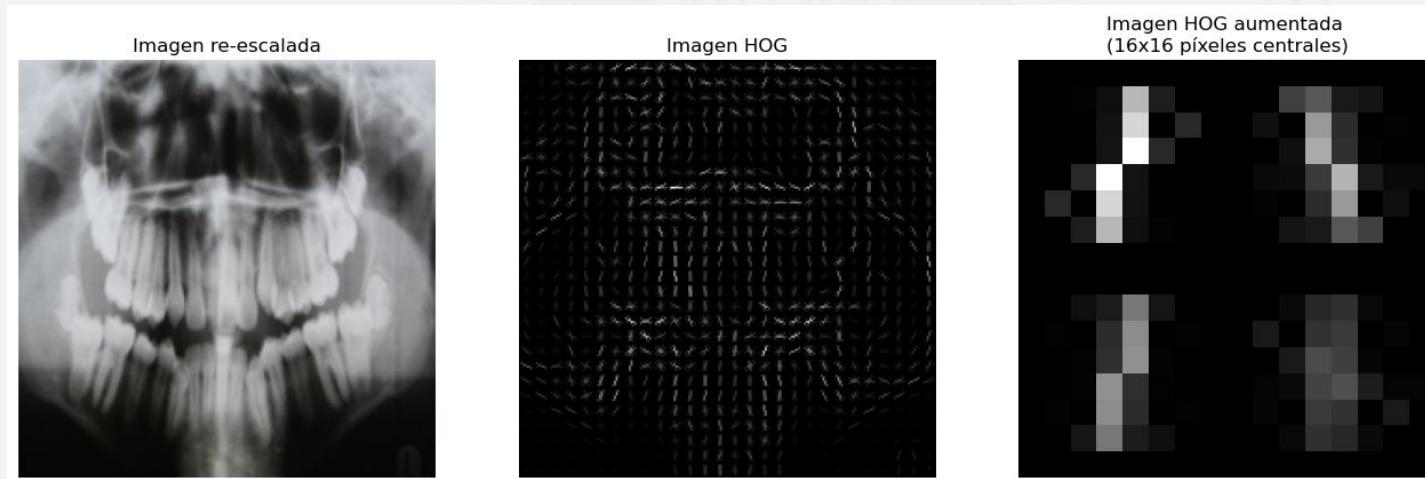


# LR-Find. Práctica II.



# Modelos Clásicos de ML. HOG.

- Técnica de **extracción de características**.
- Consiste en transformar una imagen en un vector de mucha menor dimensionalidad empleando la magnitud y la orientación del gradiente de una imagen.
- En este caso, vectores de 3.790 componentes ( $<< 224 \cdot 224 \cdot 3 = 150.528$ ).



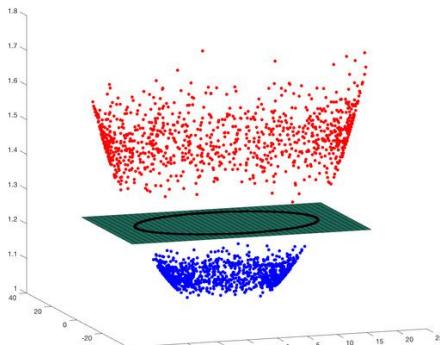
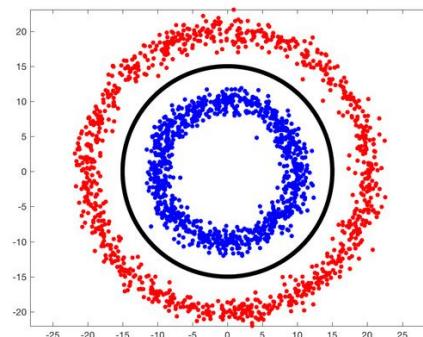
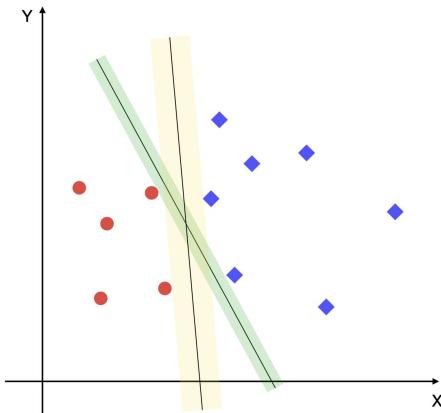
# Modelos Clásicos de ML. LR.

- Modelo lineal. Poca capacidad expresiva.
- Toma las componentes de la instancia a clasificar, las combina linealmente en base a unos parámetros entrenables y aplica una sigmoide al resultado.
- La salida del modelo, acotada entre 0 y 1, corresponde a la posibilidad de que la instancia procesada pertenezca a la clase positiva.
- Hiperparámetros a estimar.
  - Tipo de regularización ( $L$ ).
  - Factor de regularización ( $\lambda_L$ ).

# Modelos Clásicos de ML. SVM.

- Trata de encontrar un frontera (hiperplano) en el espacio dimensional en el que están definidos los datos de entrada, que no sólo separe la instancias de cada clase, si no que lo haga de forma óptima. → Maximizando el margen entre clases (distancia entre sí mismo y las instancias más cercanas de cada clase o “support vector machines”).
- Puede encontrar fronteras no lineales transformando las instancias de entrenamiento a un espacio vectorial donde sí sean linealmente separables (truco del kernel).
- Hiperparámetros a estimar.
  - Kernel.
  - Factor de regularización (regularización  $L_2$ ).
  - Grado (kernel polinómico).
  - $\gamma$  (kernel polinómico y radial): fronteras más abruptas, mayor riesgo de sobreajuste.

# Modelos Clásicos de ML. SVM.



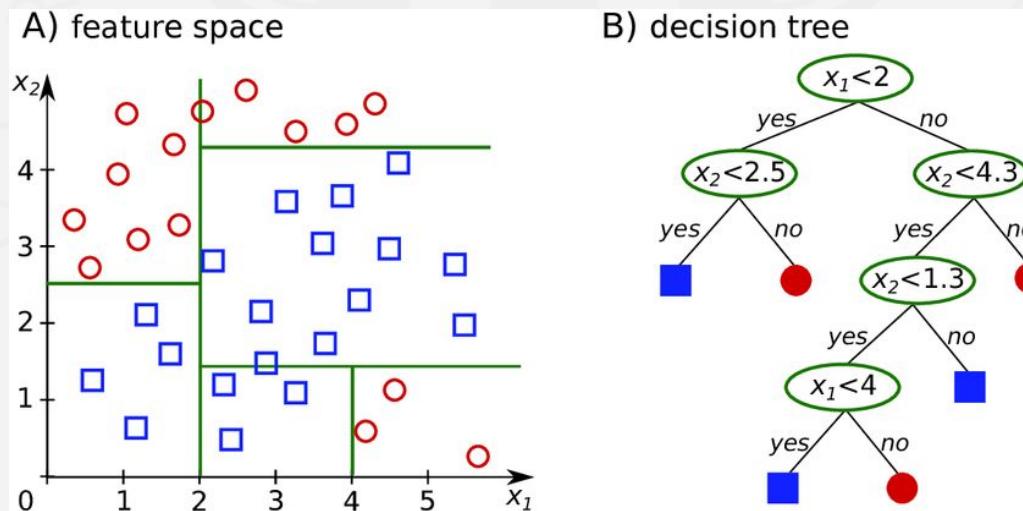
Green = Frontera no óptima.  
Yellow = Frontera óptima.

Truco del kernel.

# Modelos Clásicos de ML. RF.

- Modelo “ensemble”, formado por un conjunto de modelos → Árboles de Decisión
- Árbol de Decisión: Divide de forma jerárquica y sucesiva el espacio en el que están definidos todos los datos con los que trabaja.
- Hiperparámetros a estimar.
  - Número de estimadores.
  - Criterio de Pureza.
  - Número de características a evaluar en cada nodo.
  - Número de instancias mínimo para dividir.

# Modelos Clásicos de ML. RF.



# Experimentación. Criterio de Evaluación DL.

- Criterio para escoger el mejor candidato en base a sus resultados de validación cruzada.

$$M_{cv} = |SEN_{cv} - SPE_{cv}|$$

- Criterio para escoger el mejor candidato durante una iteración de la validación cruzada.

$$M = \alpha_M \cdot SEN + (1 - \alpha_M) \cdot SPE$$

$$\alpha_M = \{0.4, 0.5, 0.6\}$$

- Criterio evaluar/comparar varios candidatos vía validación cruzada.
  1. Para cada candidato, computar  $SEN_{cv}$  y  $SPE_{cv}$  vía M con  $\alpha_M = \{0.4, 0.5, 0.6\}$ . De cada candidato se generan 3 que difieren en  $\alpha_M$ .
  2. Escoger aquel con mejor  $M_{cv}$ .
- Maximizar el rendimiento para ambas clases (M) y que no prime una sobre otra ( $M_{cv}$  y  $\alpha_M$ ). SEN y SPE altas y similares.

# Experimentación. Criterio de Evaluación ML Clásico.

- Criterio de evaluación → menor  $M_{CV} = |SEN_{CV} - SPE_{CV}| + \text{top 5 de modelos con mejor } SEN_{cv} \text{ y } SPE_{cv} \text{ media.}$ 
  - La biblioteca utilizada no permitía guardar el mejor modelo en base a un criterio propio en cada iteración de la validación cruzada.

# Experimentación. DL.

- **ResNet18 → 31 experimentos** (ejecuciones completas de validación cruzada)
  - 2 → Con sobremuestreo y sin sobremuestreo.
  - 4 → Hiperparámetros de rotación (clases a quién aplicarlo y proporción).
  - 4 → Hiperparametros de traslación (clases a quién aplicarlo y proporción).
  - 2 → Combinación de mejor traslación y mejor rotación (juntas o por separado).
  - 5 → MixUp, alterando  $\alpha_{MU}$ .
  - 5 → MixUp (alterando  $\alpha_{MU}$ ) + M\* (técnica que mejores resultados diese de las anteriores).
  - 5 → M\* + MixUp.
  - 4 → Fine-Tuning para distintos niveles de profundidad ( $d=\{1,2,3,4\}$ ).
- **ResNet50 → 26 experimentos**
  - Fine-Tuning se detuvo para  $d=1$ . → -3 experimentos.
  - No se probaron traslaciones aplicadas sólo a la clase minoritaria. → -2 experimentos.
- Entre **10 y 60 minutos** por experimento (en función del tipo de aumento de datos).

# Experimentación. ML Clásico.

- **LR → 14 experimentos.**
  - Con sobremuestreo y sin sobremuestreo (sin regularización).
  - Todas las combinaciones posibles de:
    - $C = \{0.0001, 0.001, 0.01, 0.1, 1.0, 10.0\}$ . Mayor  $C$ , menor regularización.
    - Tipo de regularización = {Ninguna,  $L_1$ ,  $L_2$ }
- **SVM → 55 experimentos.**
  - Con sobremuestreo y sin sobremuestreo ( $C = 1.0$ , kernel radial y  $\gamma = 1/(nf * \text{Var}(X_{train}))$  ).
  - Todas las combinaciones posibles de:
    - Kernel = {Gaussiano, Radial, Polinómico, Lineal}
    - $C () = \{0.001, 0.01, 0.1, 1.0, 10.0\}$ .
    - Grado (kernel polinómico) = {2, 5, 10}
    - $\gamma$  (kernel polinómico y radial) =  $\{1/nf, 1/(nf * \text{Var}(X_{train}))\}$ .
- **Random Forest → 33 experimentos.**
  - Número de estimadores = {50, 100}.
  - Criterio de Pureza o criterio de separación = {Gini, Entropía Cruzada}.
  - Número de características a evaluar en cada nodo =  $\{\log_2(nf), \sqrt{nf}\}$ .
  - Número de instancias mínimo para dividir = {2, 10%, 20% o 30%}.

# Experimentación. Comparativa ML Clásico.

## Regression Logística

$r$	$C$	ACC	SEN	SPE	$M_{CV}$	$O_{SEN}$
-	-	0.8996	0.1655	0.9732	0.8076	0.8345
-	-	0.8773	0.2277	0.9424	0.7147	0.7723
$L_1$	0.0001	0.9089	0.0000	1.0000	1.0000	0.0000
	0.0010	0.9089	0.0000	1.0000	1.0000	0.0000
	0.0100	0.7447	0.2000	0.8000	0.6000	0.0000
	0.1000	0.7912	0.4486	0.8256	0.3770	0.2288
	1.0000	0.8680	0.3532	0.9195	0.5662	0.6468
	10.0000	0.8773	0.2588	0.9392	0.6804	0.7412
	0.0001	0.8063	0.4477	0.8422	0.3945	0.0407
	0.0010	0.8085	0.4554	0.8437	0.3883	0.1178
$L_2$	0.0100	0.8214	0.4409	0.8595	0.4186	0.4436
	0.1000	0.8601	0.3846	0.9076	0.5230	0.6154
	1.0000	0.8737	0.2665	0.9345	0.6680	0.7335
	10.0000	0.8773	0.2354	0.9416	0.7062	0.7646

## SVM

$c$	$max_f$	$n_{leaf}$	$n$	ACC	SEN	SPE	$M_{CV}$	$O_{SEN}$
Gini	$\sqrt{n_f}$	2.0000	100	0.9103	0.0625	0.9953	0.9328	0.8587
		2.0000	50	0.9082	0.1812	0.9811	0.7998	0.8188
		2.0000	100	0.9067	0.1735	0.9803	0.8067	0.8265
		0.1000	50	0.8472	0.4089	0.8911	0.4822	0.2886
		0.1000	100	0.8501	0.3775	0.8974	0.5199	0.3450
		0.2000	50	0.7841	0.4723	0.8153	0.3430	0.1400
		0.2000	100	0.7784	0.4566	0.8106	0.3540	0.1835
		0.3000	50	0.7274	0.4883	0.7514	0.2631	0.1322
		0.3000	100	0.7281	0.5037	0.7506	0.2469	0.1204
		2.0000	50	0.9110	0.2132	0.9811	0.7678	0.7868
Gini	$log_2(n_f)$	2.0000	100	0.9110	0.2286	0.9795	0.7509	0.7714
		0.1000	50	0.8321	0.4012	0.8753	0.4740	0.2071
		0.1000	100	0.8479	0.4092	0.8919	0.4826	0.2153
		0.2000	50	0.7640	0.4800	0.7924	0.3124	0.1046
		0.2000	100	0.7583	0.5037	0.7838	0.2801	0.0890
		0.3000	50	0.7425	0.5514	0.7617	0.2103	0.0587
		0.3000	100	0.7267	0.5434	0.7451	0.2017	0.0492
		2.0000	50	0.9067	0.1649	0.9811	0.8162	0.8351
		2.0000	100	0.9089	0.1575	0.9842	0.8267	0.8425
		0.1000	50	0.8450	0.3855	0.8911	0.5055	0.2999
Entropy	$\sqrt{n_f}$	0.1000	100	0.8508	0.3778	0.8982	0.5203	0.3430
		0.2000	50	0.7848	0.4646	0.8169	0.3523	0.1437
		0.2000	100	0.7848	0.4566	0.8177	0.3611	0.1855
		0.3000	50	0.7274	0.4883	0.7514	0.2631	0.1322
		0.3000	100	0.7281	0.5037	0.7506	0.2469	0.1204
		2.0000	50	0.9075	0.1889	0.9795	0.7906	0.8111
		2.0000	100	0.9096	0.2052	0.9803	0.7750	0.7948
		0.1000	50	0.8393	0.3935	0.8840	0.4904	0.2170
		0.1000	100	0.8479	0.4092	0.8919	0.4826	0.2036
		0.2000	50	0.7647	0.4800	0.7932	0.3132	0.1046
Entropy	$log_2(n_f)$	0.2000	100	0.7597	0.5037	0.7853	0.2816	0.0831
		0.3000	50	0.7425	0.5514	0.7617	0.2103	0.0587
		0.3000	100	0.7281	0.5434	0.7467	0.2033	0.0492

## Random Forest

$k$	$d$	$C$	$\gamma$	ACC	SEN	SPE	$M_{CV}$	$O_{SEN}$
$g$	-	1.0	-	0.9075	0.0080	0.9976	0.9896	0.1000
$p$	10	10.0000	auto	0.6384	0.6206	0.6401	0.0195	0.2716
$s$	-	10.0000	scale	0.6636	0.6369	0.6661	0.0292	0.0860
$p$	10	1.0000	auto	0.6779	0.5431	0.6914	0.1483	0.2999
$g$	-	10.0000	auto	0.7927	0.4717	0.8248	0.3531	0.4243
$l$	-	0.0100	-	0.8149	0.4868	0.8476	0.3609	0.4957
$s$	-	1.0000	scale	0.8006	0.4637	0.8342	0.3705	0.0979
$s$	-	10.0000	auto	0.7991	0.4560	0.8335	0.3775	0.3023
$p$	5	0.0010	scale	0.7977	0.4243	0.8350	0.4107	0.3361
$p$	2	0.1000	scale	0.8372	0.4483	0.8761	0.4278	0.5438
$g$	-	0.1000	scale	0.8329	0.3852	0.8777	0.4924	0.3829
$p$	5	10.0000	auto	0.8357	0.3622	0.8832	0.5211	0.0867
$p$	5	0.1000	auto	0.8357	0.3622	0.8832	0.5211	0.0867
$p$	5	0.0100	auto	0.8357	0.3622	0.8832	0.5211	0.0867
$p$	5	0.0010	auto	0.8357	0.3622	0.8832	0.5211	0.0867
$p$	5	1.0000	auto	0.8357	0.3622	0.8832	0.5211	0.0867
$l$	-	0.0010	-	0.8709	0.3462	0.9234	0.5773	0.0615
$p$	2	0.0100	scale	0.8766	0.3382	0.9306	0.5924	0.0616
$p$	2	10.0000	auto	0.8866	0.2911	0.9463	0.6553	0.0006
$p$	2	1.0000	auto	0.8866	0.2911	0.9463	0.6553	0.0006
$p$	2	0.0010	scale	0.8866	0.2911	0.9463	0.6553	0.0006

# Experimentación. Comparativa Global.

## Validación Cruzada

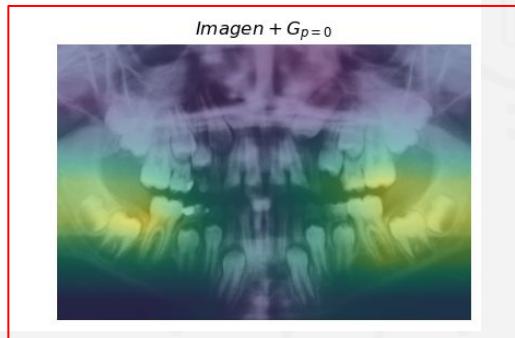
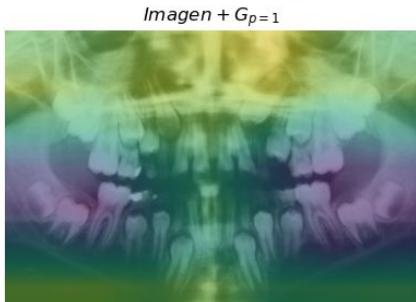
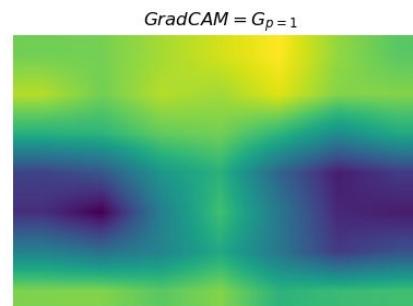
Modelo	<i>ACC</i>	<i>SEN</i>	<i>SPE</i>
<i>ResNet18</i>	0.7475	0.7486	0.7475
<i>ResNet50</i>	0.7640	0.7637	0.7640
<i>LR</i>	0.7912	0.4486	0.8256
<i>SVM</i>	0.6636	0.6369	0.6661
<i>RF</i>	0.7267	0.5434	0.7451
<i>Dummy</i>	0.5064	0.4966	0.5075

Humano = Angelakopoulos et al. (2018) = Experto humano. Misma población.  
 C = Guo et al. (2021) = Método automático. Distinta Población. Chinos.  
 B = Murray et al. (2024) = Método automático. Distinta Población. Brasileños.

## Test

Modelo	<i>ACC</i>	<i>SEN</i>	<i>SPE</i>
<i>ResNet18</i>	0.7241	0.7126	0.7356
<i>ResNet50</i>	0.7414	0.7644	0.7184
<i>LR</i>	0.6695	0.5287	0.8103
<i>SVM</i>	0.6580	0.4828	0.8333
<i>RF</i>	0.6379	0.5345	0.7414
<i>Dummy</i>	0.4626	0.4828	0.4425
<i>Humano</i>	0.9004	0.8080	0.9588
<i>C</i>	0.9230	0.9260	0.9190
<i>B</i>	0.8750	0.8700	0.8790

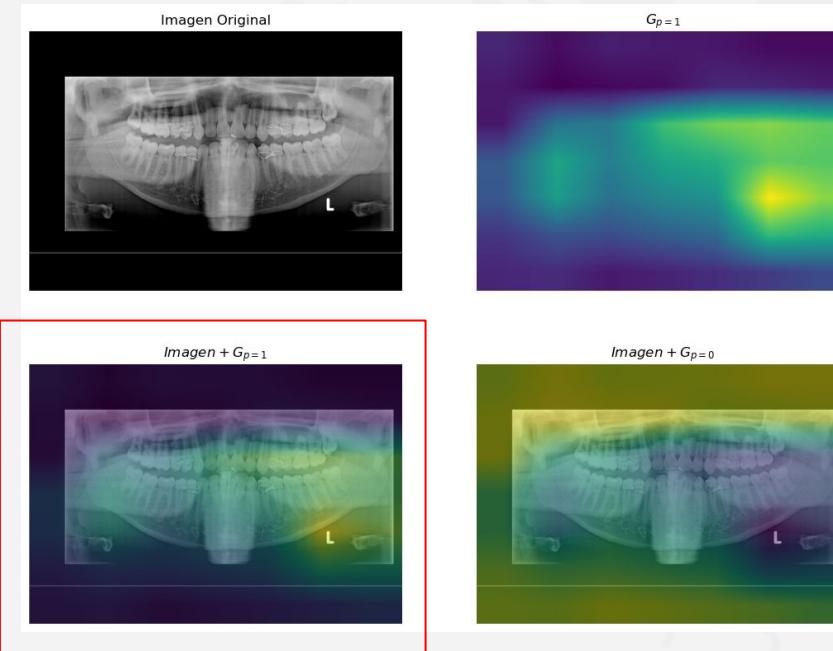
# Experimentación. Explicabilidad. Caso Prometedor II.



- Edad = 9.95 años.
- Predicción = 0  
( $p=0.90$ ).

Se centra en el prematuro estado de desarrollo de los molares de ambos lados de la boca.

# Experimentación. Explicabilidad. Caso Problemático II.



- Edad = 20.65 años.
- Predicción = 1  
( $p=0.94$ ).

La región de la imagen que más influye a la predicción del individuo corresponde a una anotación externa en la radiografía. → **¿posible sesgo en los datos?**