Informe sobre la Selección de Arquitecturas para la Clasificación de Enfermedades Oculares del Ciclo 1

# 1. Planteamiento del Problema

Las discapacidades visuales son una preocupación global significativa. En Colombia, alrededor de 4000 personas por millón requieren servicios de baja visión, con el glaucoma y la retinopatía diabética siendo las principales causas. La detección temprana es crucial, y este proyecto busca mejorar la precisión del diagnóstico mediante el uso de redes neuronales convolucionales pre-entrenadas.

# 2. Proceso de Evaluación y Selección de Arquitecturas

## 2.1. Metodología

Se evaluaron cinco arquitecturas (VGG19, ResNet50, Xception, EfficientNet e InceptionV3) utilizando un dataset con imágenes de fondo de ojo, probando diferentes tasas de aprendizaje (0.01, 0.001, 0.0001) y repitiendo los experimentos diez veces.

## 2.2. Hiperparámetros y Configuraciones

* **Learning Rates:** 0.01, 0.001, 0.0001
* **Batch Size:** 16
* **Número de Épocas:** 40
* **Repeticiones:** 10 por learning rate
* **Función de Pérdida:** Entropía cruzada
* **Función de Activación:** Softmax
* **Optimización:** Adam
* **Validación Cruzada:** Implementada en este ciclo 1

## 2.3 Dataset utilizado

El conjunto de datos utilizado en este estudio es el "Clasificación de las enfermedades oculares", compilado por Guna Venkat Doddi en 2022. Este dataset contiene alrededor de 1,000 imágenes diagnósticas de retina para cada clase, incluyendo retinopatía diabética, cataratas, glaucoma y ojos normales. Con el fin de simplificar el problema y evaluar la capacidad de los modelos en la distinción básica entre ojos sanos y enfermos, se decidió realizar la clasificación en dos categorías: sano y enfermo, en lugar de las cuatro clases iniciales.

## 2.4 Ajuste de paciencia

En este estudio, se utilizó una configuración de "paciencia" (patience) que permite reducir la tasa de aprendizaje si la precisión de validación no mejora después de 5 épocas. Si la precisión de validación no mejora después de 10 épocas, el entrenamiento se detiene. Esta técnica ayudó a prevenir el sobreajuste y a mantener la eficiencia del modelo.

# 3. Análisis de Resultados

## 3.1. VGG19

* **Varianza General:**

| learning\_rate | varianza\_val\_loss | varianza\_val\_accuracy |
| --- | --- | --- |
| 0.0001 | 0.05101759482 | 7.14E-05 |
| 0.001 | 0.000689103722 | 0.0002064738184 |
| 0.01 | 0.01381640637 | 0.005167020677 |

* **Resumen General:**

| learning\_rate | repeticion | mejor\_val\_accuracy | mejor\_val\_loss |
| --- | --- | --- | --- |
| 0.0001 | 6 | 0.957 | 0.9441 |
| 0.001 | 3 | 0.937 | 0.2675 |
| 0.01 | 6 | 0.928 | 0.2323 |

**Discusión:**

* VGG19 mostró un rendimiento sólido con tasas de aprendizaje menores, destacándose por una alta precisión y baja pérdida de validación. Su capacidad para capturar características es notable, aunque la varianza aumenta con tasas de aprendizaje mayores.
* **VGG19** es conocida por su simplicidad en diseño, con capas convolucionales apiladas de manera uniforme (Simonyan & Zisserman, 2014). En los experimentos se evidencian resultados significativos como:
* **Val\_accuracy:** Consistentemente alta, alcanzando hasta un 0.957 en las mejores repeticiones, siendo el tercer mejor valor entre las cinco arquitecturas.
* **Val\_loss:** Mostró una baja pérdida de validación, esto indica una buena capacidad para generalizar sobre el conjunto de prueba.

**Conclusión**: VGG19 fue seleccionada debido a su **simplicidad** y **rendimiento constante**, además de ser más fácil de ajustar y tener una facilidad de adaptación más notable que otras arquitecturas más complejas.

## 3.2. ResNet50

* **Varianza General:**

| learning\_rate | varianza\_val\_loss | varianza\_val\_accuracy |
| --- | --- | --- |
| 0.0001 | 0.002705031821 | 2.00E-05 |
| 0.001 | 0.00651681603 | 5.74E-05 |
| 0.01 | 0.01331717177 | 0.004911075686 |

* **Resumen General:**

| learning\_rate | repeticion | mejor\_val\_accuracy | mejor\_val\_loss |
| --- | --- | --- | --- |
| 0.0001 | 7 | 0.964 | 0.3002 |
| 0.001 | 3 | 0.966 | 0.2126 |
| 0.01 | 1 | 0.882 | 0.3625 |

**Discusión:**

* ResNet50 presentó un rendimiento superior con tasas de aprendizaje más bajas, mostrando una alta precisión y baja pérdida de validación. Su arquitectura permite el manejo eficiente de características profundas, esto se traduce en un buen desempeño en tareas de clasificación (S. Ortiz & M. A. Goenaga Jimenez, 2023).
* **ResNet50** es conocida por sus bloques residuales, que permiten el flujo de gradiente a través de capas más profundas (S. Ortiz & M. A. Goenaga Jimenez, 2023). Se evidencia lo siguiente:
* **Val\_accuracy:** Alcanzó consistentemente valores altos, hasta 0.966, siendo el segundo mejor.
* **Val\_loss:** Se mantuvo baja, mostrando una gran capacidad para generalizar sin caer en problemas de sobreajuste.

**Conclusión**: ResNet50 fue seleccionada por su **balance entre profundidad y precisión**, aprovechando sus bloques residuales para entrenar redes más profundas sin comprometer el rendimiento.

## 3.3. EfficientNet (Versión B3)

* **Varianza General:**

| learning\_rate | varianza\_val\_loss | varianza\_val\_accuracy |
| --- | --- | --- |
| 0.01 | 0.0006407968539 | 0.0001732614944 |
| 0.001 | 0.002088421583 | 7.32E-05 |
| 0.0001 | 0.9934003025 | 0.0001138630617 |

* **Resumen General:**

| learning\_rate | repeticion | mejor\_val\_accuracy | mejor\_val\_loss |
| --- | --- | --- | --- |
| 0.01 | 3 | 0.942 | 0.1917 |
| 0.001 | 3 | 0.969 | 0.239 |
| 0.0001 | 2 | 0.959 | 0.6561 |

**Discusión:**

* EfficientNet mostró una excelente precisión con tasas de aprendizaje de 0.001 y 0.0001, su versión 3 (B3) utiliza un enfoque de "compound scaling" que ajusta simultáneamente la profundidad, el ancho y la resolución de la red, esto resulta en una precisión superior (Elmoufidi & Amoun, 2021). A pesar de la mayor pérdida de validación con la tasa más baja, su eficiencia en términos de parámetros y rendimiento es notable.
* **EfficientNet** es una arquitectura más reciente que ajusta automáticamente la resolución de entrada, la profundidad y el número de filtros para optimizar el rendimiento, además de destacarse por su buen rendimiento en sistemas con limitaciones computacionales (Elmoufidi & Amoun, 2021). En los experimentos se demostró que:
* **Val\_accuracy:** EfficientNet fue la más consistente, con valores que llegaron hasta 0.969. siendo el mejor.
* **Val\_loss:** Fue el modelo con la menor pérdida en validación, mostrando un excelente equilibrio entre capacidad de generalización y precisión.

**Conclusión**:EfficientNet fue seleccionada por su **eficiencia generalizada**, logrando un rendimiento excelente con una arquitectura más optimizada, lo que la hace ideal para su despliegue en sistemas con recursos computacionales limitados.

## 3.4. Xception

* **Varianza General:**

| learning\_rate | varianza\_val\_loss | varianza\_val\_accuracy |
| --- | --- | --- |
| 0.01 | 0.0235267522 | 0.009047988737 |
| 0.0001 | 0.006131442435 | 2.26E-05 |
| 0.001 | 0.0002303632532 | 2.61E-05 |

* **Resumen General:**

| learning\_rate | repeticion | mejor\_val\_accuracy | mejor\_val\_loss |
| --- | --- | --- | --- |
| 0.01 | 2 | 0.94 | 0.2086 |
| 0.0001 | 6 | 0.949 | 0.5279 |
| 0.001 | 8 | 0.947 | 0.2614 |

**Discusión:**

* Xception es una arquitectura que extiende la idea de las convoluciones separables en profundidad, optimizando el flujo de información a través de un diseño más eficiente en términos de computación (Lim et al., 2022). A pesar de esto, en los experimentos realizados con las imágenes de retinoscopía, los resultados obtenidos no fueron los mejores comparados con otras arquitecturas. En los experimentos se evidenció que:
* **Val\_accuracy**: Los valores de accuracy en Xception se mostraron inconsistentes. Aunque en algunas repeticiones lograba valores superiores a 0.80, había una alta variabilidad entre los experimentos. En general, no superó el rendimiento de EfficientNet o ResNet50, y su estabilidad fue inferior.
* **Val\_loss**: El valor de la pérdida en validación también fluctuaba demasiado, lo que indicaba que el modelo tenía problemas de sobreajuste y generalización con los datos de validación, presentando una pérdida significativamente más alta que otros modelos después de 40 épocas.

**Conclusión**: Aunque es un modelo potente y ha demostrado eficacia en otros dominios, en esta tarea específica no mostró un rendimiento lo suficientemente competitivo, esto justifica su exclusión.

## 3.5. InceptionV3

* **Varianza General:**

| learning\_rate | varianza\_val\_loss | varianza\_val\_accuracy |
| --- | --- | --- |
| 0.01 | 3.61E-05 | 5.23E-07 |
| 0.001 | 0.04614476938 | 4.55E-05 |
| 0.0001 | 1.07494822 | 8.65E-05 |

* **Resumen General:**

| learning\_rate | repeticion | mejor\_val\_accuracy | mejor\_val\_loss |
| --- | --- | --- | --- |
| 0.01 | 7 | 0.74 | 0.5913 |
| 0.001 | 6 | 0.949 | 0.3377 |
| 0.0001 | 9 | 0.933 | 0.7647 |

**Discusión:**

* **InceptionV3** es una de las versiones más avanzadas de la arquitectura Inception, que utiliza una mezcla de convoluciones de diferentes tamaños para capturar características a varias escalas (Lim et al., 2022). No obstante, en esta aplicación específica de clasificación de imágenes de retinoscopía, los resultados fueron inferiores comparados con VGG19, ResNet50 y EfficientNet. Los experimentos mostraron que:
* **Val\_accuracy:** InceptionV3 mostró una precisión moderada, con valores que oscilaban alrededor de 0.74 en su tasa de aprendizaje más alta, lo que lo colocaba por debajo de las arquitecturas elegidas para los experimentos finales. Aunque en ciertas pruebas se observó una convergencia rápida, no logró superar a EfficientNet o ResNet50 en consistencia o precisión final.
* **Val\_loss:** La pérdida en validación fue más estable que en Xception, pero tampoco mostró una mejora significativa respecto a VGG19 o ResNet50. En algunos casos, el modelo parecía estancarse, lo que sugiere que no estaba aprovechando por completo la capacidad de representación de los datos de retinoscopía.

**Conclusión**: Aunque el rendimiento fue relativamente bueno, no logró superar las tres arquitecturas seleccionadas en términos de precisión o estabilidad, lo que justificó su exclusión.

# 4. Conclusiones y Justificación de Selección

## 4.1 Conclusiones con respecto a las tablas de varianza y tablas generales

A partir lo anterior concluimos que:

* **ResNet50** ofrece un manejo eficiente de características profundas y logra una alta precisión, especialmente con tasas de aprendizaje más bajas. Durante los experimentos, mostró una baja varianza en los resultados de validación, lo que indica una buena capacidad para generalizar. Los bloques residuales de su arquitectura permiten entrenar redes profundas sin comprometer el rendimiento, lo que la hace una excelente opción para la clasificación de imágenes complejas como las de retinoscopía (S. Ortiz & M. A. Goenaga Jimenez, 2023). ResNet50 fue seleccionada por su equilibrio entre precisión, estabilidad y capacidad para manejar características profundas, destacando su bajo riesgo de sobreajuste en comparación con otras arquitecturas.
* **EfficientNet** se distingue por su enfoque en la eficiencia, logrando un rendimiento competitivo con menos parámetros. Su alta precisión y buena estabilidad con tasas de aprendizaje más bajas refuerzan su idoneidad para la clasificación de imágenes de retinoscopía. Adicionalmente, EfficientNet optimiza la resolución de entrada, la profundidad y los filtros, lo que permite obtener un gran rendimiento incluso en sistemas con limitados recursos computacionales (Elmoufidi & Amoun, 2021). EfficientNet fue seleccionada por su excelente equilibrio entre eficiencia y rendimiento, sobresaliendo por su capacidad de ofrecer alta precisión con una arquitectura más ligera.
* **VGG19** mostró una alta precisión con tasas de aprendizaje bajas, aunque presentó una mayor varianza en la pérdida con tasas de aprendizaje más altas. A pesar de esta variabilidad, sigue siendo una opción sólida gracias a su consistencia en tasas más bajas y su habilidad para capturar características profundas. La simplicidad de su arquitectura facilita su ajuste, lo que reduce el riesgo de sobreajuste en tareas de clasificación complejas.VGG19 fue seleccionada por su rendimiento constante y su capacidad para captar características profundas, siendo una opción confiable para este tipo de problemas, pese a la mayor variabilidad observada en condiciones más exigentes.
* **Xception** e **InceptionV3** también lograron buenos resultados, pero mostraron una mayor variabilidad en sus métricas de desempeño. Su sensibilidad a las tasas de aprendizaje; en particular, afectó su consistencia en comparación con las arquitecturas seleccionadas. Aunque tienen ventajas en otros dominios, para esta tarea específica no ofrecieron la misma estabilidad ni precisión que **ResNet50**, **EfficientNet** y **VGG19**.

## 4.2 Conclusiones con respecto a las tablas de métricas de cada arquitectura y matrices de confusión:

Adicionalmente y a partir de las matrices de confusión y las tablas de métricas para los modelos ResNet50, VGG19 y EfficientNet (las cuales se encontrarán en la hoja de cálculo adjunta en Anexos) , se puede realizar el siguiente análisis comparativo:

### 4.2.1 Precisión

**ResNet50:** La precisión del modelo es del 93% en general. El modelo es capaz de identificar correctamente a los pacientes con enfermedades en el 95% de los casos, pero para los pacientes sanos, la precisión cae al 85%.

**VGG19:** La precisión mejora ligeramente en comparación con ResNet50, alcanzando un 95%. Para los pacientes enfermos, se mantiene alta en 97%, y para los pacientes sanos es del 90%.

**EfficientNet**: Es el modelo con la mayor precisión, logrando un 99%. Tanto para pacientes enfermos como sanos, la precisión es sobresaliente con un 99% y 98% respectivamente.

### 4.2.2 Recall (Sensibilidad)

**ResNet50:** El recall del modelo es 0.95 para los pacientes enfermos, pero es algo más bajo para los pacientes sanos, con un valor de 0.87. Esto sugiere que aunque ResNet50 es muy bueno identificando a pacientes enfermos, tiene más dificultades para detectar correctamente a los pacientes sanos.

**VGG19:** El recall general es 0.95. Para los pacientes enfermos, es 0.96 y para los pacientes sanos es 0.91. Esto indica que VGG19 es más equilibrado, logrando mejorar la detección de pacientes sanos en comparación con ResNet50.

**EfficientNet:** El mejor rendimiento en términos de recall también es para EfficientNet, con un 0.99 para pacientes enfermos y un 0.98 para pacientes sanos, lo que refleja su capacidad de generalización y su equilibrio al detectar ambas clases.

### 4.2.3 F1-Score

**ResNet50:** El f1-score global del modelo es 0.93, con valores de 0.95 para pacientes enfermos y 0.86 para sanos. Aunque el modelo tiene un rendimiento sólido, presenta una ligera caída al tratar de clasificar pacientes sanos.

**VGG19:** Este modelo tiene un f1-score general de 0.95. Al igual que con la precisión y el recall, VGG19 logra mejores resultados que ResNet50, con un f1-score de 0.97 para los enfermos y 0.90 para los sanos.

**EfficientNet:** EfficientNet es el que mejor se desempeña, con un f1-score global de 0.99. Esto se debe a su excelente equilibrio entre precisión y recall, con 0.99 para pacientes enfermos y 0.98 para sanos.

### 4.2.4 Exactitud General (Accuracy)

**ResNet50:** La exactitud global es del 93%, lo que muestra un buen rendimiento, aunque no el mejor entre los modelos evaluados.

**VGG19:** Mejora en comparación con ResNet50, logrando una exactitud del 95%.

**EfficientNet:** Este modelo destaca con una exactitud del 99%, lo que lo convierte en el más robusto y eficiente en la clasificación de las imágenes de retinoscopía.

### 4.2.5 Comparación basada en las matrices de confusión:

**ResNet50:**

* Clasifica correctamente 297 pacientes enfermos, con solo 11 clasificaciones erróneas.
* Para los pacientes sanos, clasifica correctamente 97, con 10 falsos negativos, lo que indica cierta dificultad al detectar sanos.

**VGG19:**

* Clasifica correctamente 292 pacientes enfermos y comete 16 errores, ligeramente peor que ResNet50 en esta clase.
* Sin embargo, clasifica 93 pacientes sanos correctamente, con 14 falsos negativos.

**EfficientNet:**

* Clasifica correctamente a 306 pacientes enfermos, con solo 2 errores, y es el que mejor rendimiento tiene en esta clase.
* Clasifica correctamente a 105 pacientes sanos, con solo 2 falsos negativos, superando tanto a ResNet50 como a VGG19 en ambas clases.

# 5. Anexos

En el siguiente enlace de hojas de cálculo se encontrará todos los resultados (Tablas generales, específicas y de varianza) de todas las arquitecturas, así como las tablas de métricas y mapas de calor (matrices de confusión) de VGG19, Efficientnet y Resnet50: https://docs.google.com/spreadsheets/d/18GqFupcKL0uT1mN5Ji31Urf7yENDeNsU5bf5WnM5sPQ/edit?gid=0#gid=0

# 6. Referencias

Elmoufidi, Abdelali & Amoun, Hind. (2021). EfficientNetB3 Architecture for Diabetic Retinopathy Assessment using Fundus Images. 10.21203/rs.3.rs-609899/v1.

Lim, Wee & Ho, Heng-Yen & Ho, Heng-Chen & Chen, Yan-Wu & Lee, Chih-Kuo & Chen, Pao-Ju & Lai, Feipei & Jang, Jyh-Shing & Ko, Mei-Lan. (2022). Use of multimodal dataset in AI for detecting glaucoma based on fundus photographs assessed with OCT: focus group study on high prevalence of myopia. BMC Medical Imaging. 22. 10.1186/s12880-022-00933-z.   
S. Ortiz & M. A. Goenaga Jimenez. (2023, 2 de noviembre). Deep Learning-Based Ocular Disease Classification in Fundus Images. IEEE Colombian Caribbean Conference (C3), Barranquilla, Colombia, 2023. pp. 1-6. doi: 10.1109/C358072.2023.10436234.  
Simonyan, K., & Zisserman, A. (2014, 4 septiembre). Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv.org. <https://arxiv.org/abs/1409.1556>

Vipas.Ai. (2024, 8 agosto). RESNet-50: A robust Image Classification Model - VIPAS.AI - Medium. Medium. <https://medium.com/@vipas.ai/resnet-50-a-robust-image-classification-model-9bd3fffbc3e3>