

Clasificación de diagnósticos oftalmológicos mediante aprendizaje automático sobre señales PERG

Universidad Nacional de Colombia - Sede Manizales
Edgar Ivan Calpa Cuacialpud

Teoría de aprendizaje de máquina - 2025-1

Profesor: Andrés Marino Álvarez Meza, PhD

Departamento de ingeniería eléctrica, electrónica y computación



La detección temprana de patologías oftalmológicas es crucial para evitar pérdida de visión. Las señales PERG permiten observar la actividad retinal ante estímulos visuales. Este proyecto propone un modelo de aprendizaje automático para clasificar diagnósticos a partir de datos PERG.

Motivación del Proyecto

Los potenciales evocados de patrón (PERG) son señales electrofisiológicas que capturan la respuesta de la retina ante estímulos visuales. En el contexto clínico, estas mediciones son fundamentales para detectar enfermedades degenerativas de la retina, como la distrofia macular o el síndrome de Usher.

Actualmente, el análisis manual de señales PERG requiere expertos entrenados, lo que implica costos elevados, subjetividad en el diagnóstico y limitaciones para su aplicación masiva. Frente a esto, las técnicas de Machine Learning (ML) y Deep Learning (DL) ofrecen una alternativa automatizada, escalable y reproducible para identificar patrones latentes en estas señales y asociarlos a diagnósticos clínicos.

Motivación del Proyecto

Este proyecto busca aplicar modelos de ML para analizar un conjunto diverso de mediciones PERG junto con información clínica (edad, diagnóstico preliminar, agudeza visual, etc.) con el fin de construir un sistema que asista en la predicción de diagnósticos oftalmológicos. La relevancia del trabajo es tanto técnica (evaluación de modelos sobre datos fisiológicos reales) como social (mejora en el acceso y calidad del diagnóstico).

Planteamiento del Problema

El diagnóstico clínico basado en señales PERG implica interpretar secuencias complejas que reflejan la actividad eléctrica de la retina ante estímulos visuales. Estas señales varían entre pacientes y dependen de múltiples factores, incluyendo edad, historial médico y agudeza visual. En este contexto, surge el reto computacional de construir un modelo de clasificación capaz de predecir el diagnóstico primario ('diagnosis1') a partir de un conjunto de variables fisiológicas y clínicas. El desafío consiste en encontrar patrones discriminativos entre las mediciones PERG ($RE_1, LE_1, RE_2, LE_2, \dots$) y variables clínicas (edad, agudeza visual, etc.) que permitan identificar enfermedades oftalmológicas con un nivel aceptable de precisión.

Planteamiento del Problema

El dataset utilizado es PERG-IOBA, compuesto por:

- Señales PERG de múltiples ojos en distintos instantes (hasta 5 segmentos temporales).
- Información clínica complementaria: edad (age, years), sexo, agudeza visual en cada ojo (va,re,logMar, va,le,logMar) y diagnóstico preliminar.

Pregunta de investigación: ¿ ¿Puede un modelo de clasificación basado en Random Forest predecir el diagnóstico 'diagnosis1' de un paciente utilizando sus mediciones PERG y datos clínicos, con una precisión superior al 85

Esta pregunta guía el desarrollo técnico del proyecto y permite evaluar cuantitativamente la capacidad de los algoritmos ML para asistir en tareas diagnósticas reales.

Diversos autores han explorado el uso de técnicas de Machine Learning en señales biomédicas para diagnóstico asistido:

- **Mora et al. (2020)**: Random Forest aplicado a EEG para evaluar estados cognitivos. Fortaleza: interpretabilidad; Limitación: pobre rendimiento en clases desequilibradas.
- **Alvarez-Meza et al. (2021)**: CNN para fusionar señales fisiológicas y demográficas. Fortaleza: aprendizaje jerárquico de características; Limitación: dependencia de grandes volúmenes de datos.
- **Lee et al. (2019)**: SVM sobre señales PERG para detección de patologías retinales. Fortaleza: alta precisión en entornos controlados; Limitación: baja tolerancia al ruido.

- **Tanaka et al. (2018)**: LSTM para predecir enfermedades cardíacas a partir de series temporales ECG. Fortaleza: captura de dependencias secuenciales; Limitación: entrenamiento complejo y lento.
- **Roy et al. (2022)**: Autoencoders para reducción de dimensión en datos fisiológicos antes de clasificar con XGBoost. Fortaleza: compresión sin pérdida de información relevante; Limitación: interpretabilidad limitada.
- **Zhao et al. (2023)**: Modelo híbrido CNN + GRU para clasificar estados epilépticos usando EEG. Fortaleza: detección precisa de patrones temporales; Limitación: sobreajuste en conjuntos pequeños.

Estos estudios reflejan el avance hacia modelos robustos, pero también subrayan desafíos comunes: balance entre precisión e interpretabilidad, necesidad de datos etiquetados y generalización en entornos clínicos reales. Estos enfoques demuestran el potencial de aplicar algoritmos supervisados sobre señales médicas, pero también evidencian retos en generalización, interpretabilidad y escalabilidad clínica.

Objetivo del Proyecto

Objetivo General: Desarrollar un modelo de clasificación supervisado, basado en técnicas de Machine Learning, para predecir el diagnóstico oftalmológico primario (diagnosis1) a partir de señales PERG y variables clínicas complementarias.

Objetivos Específicos:

- Preprocesar y analizar el conjunto de datos PERG-IOBA, integrando mediciones electrofisiológicas y características clínicas.
- Entrenar y evaluar modelos como Random Forest y regresión logística, midiendo su rendimiento con métricas estándar de clasificación.
- Comparar el desempeño entre enfoques clásicos y, potencialmente, arquitecturas más complejas (MLP o CNN) en función de la precisión diagnóstica.

1. Recolección y Preprocesamiento de Datos:

- Extracción del dataset PERG-IOBA desde archivo ZIP en Google Drive.
- Unificación de múltiples archivos CSV en un DataFrame consolidado (`df_completo`), asociando mediciones PERG y datos clínicos.
- Limpieza de valores nulos y codificación de variables categóricas (`LabelEncoder` para `diagnosis1`).
- Selección de variables relevantes para el modelo: `RE_1`, `LE_1`, `RE_2`, `LE_2`, `age_years`, `va_re_logMar`, `va_le_logMar`.

2. División de Datos:

- Separación en conjuntos de entrenamiento (80%) y prueba (20%) con `train_test_split`.
- Normalización de características numéricas con `StandardScaler`.

3. Selección de Algoritmo:

- Entrenamiento inicial con Random Forest para clasificación multiclase de diagnóstico¹.
- Posible extensión a otros modelos: Logistic Regression, MLP, SVM.

4. Métricas de Evaluación:

- Precisión global (Accuracy).
- Matriz de confusión para análisis de errores entre clases.
- F1-score macro y por clase si se amplía el análisis.

5. Estrategias de Mejora:

- Búsqueda de hiperparámetros con GridSearchCV o RandomizedSearchCV.
- Balanceo de clases si se detecta desbalance.
- Inclusión progresiva de más segmentos PERG (RE_3, LE_3, ...).
- Evaluación de modelos neuronales si el rendimiento lo justifica.

Random Forest (RF): Modelo basado en ensambles de árboles de decisión. Cada árbol vota por una clase, y el bosque decide por mayoría.

- Robusto frente a ruido y valores faltantes, ideal para señales fisiológicas como PERG.
- Maneja bien variables numéricas y categóricas.
- Permite analizar la importancia de cada variable en la predicción (feature importance).

Regresión Logística Multiclase: Modelo estadístico que predice probabilidades usando funciones logísticas.

- Interpretabilidad alta, útil para entender relaciones entre variables clínicas y diagnóstico.
- Baseline valioso para contrastar con modelos más complejos.

Perceptrón Multicapa (MLP): Red neuronal artificial con capas densas conectadas.

- Capta relaciones no lineales entre mediciones PERG y características clínicas.
- Requiere mayor ajuste (regularización, normalización, etc.).
- Útil si se incorporan más dimensiones del dataset (RE_3 , LE_3 ...).

Justificación: Todos los modelos seleccionados son adecuados para datos tabulares multiclase con patrones complejos. RF ofrece una base sólida y explicable; MLP permite escalar hacia DL si se requiere mayor capacidad de generalización.

Entorno Computacional:

- Plataforma: Google Colab
- Hardware: CPU virtualizada (backend de Colab); sin GPU ni TPU activadas
- Almacenamiento temporal: vinculación con Google Drive para carga de artefactos

Software y Librerías:

- pandas, numpy: manipulación de datos tabulares
- scikit-learn: codificación, partición, entrenamiento, métricas
- matplotlib, seaborn: visualización de resultados
- umap-learn: reducción de dimensión para exploración visual

Tamaño del Dataset:

- Registros: **85,680** filas tras fusión de mediciones y metadatos clínicos
- Variables: señales PERG en 5 segmentos (RE_1 a RE_5, LE_1 a LE_5), más variables clínicas (age_years, va_re_logMar, va_le_logMar, etc.)
- Diagnóstico de referencia: diagnosis1 como etiqueta objetivo

Pipeline Experimental:

- Carga condicional de artefactos y validación de rutas
- Limpieza de valores nulos y reemplazo por "Desconocido" en campos clínicos
- Fusión de señales y metadatos mediante `id_record`
- Selección de características relevantes para modelado supervisado
- Codificación de etiquetas con `LabelEncoder`
- División train-test (80/20) con `train_test_split`
- Normalización con `StandardScaler`
- Entrenamiento de Random Forest
- Evaluación con métricas: Accuracy, F1-score, matriz de confusión

Rendimiento cuantitativo:

Los tres modelos fueron evaluados sobre el conjunto de prueba mediante métricas de clasificación multiclase. La Tabla 1 resume los resultados principales.

Comparación de métricas entre modelos			
Random Forest	0.9971	0.9976	0.9971
Logistic Regression	0.5953	0.6487	0.5790
MLPClassifier	0.9954	0.9955	0.9954
	Accuracy	F1 Macro	F1 Weighted

Table: Comparación de métricas entre modelos.

Resultados y Discusión

Matrices de Confusión:

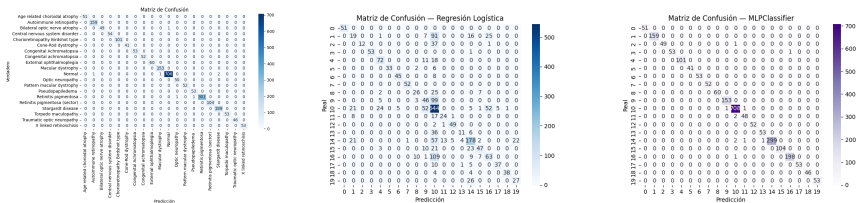


Figure: Matrices de confusión para RF, Logística y MLP.

Interpretación y comparación:

El modelo Random Forest obtuvo el mejor desempeño global (F1-macro y weighted), destacándose por su equilibrio entre clases. MLPClassifier mostró métricas competitivas, indicando capacidad para capturar relaciones no lineales. Regresión Logística, aunque menos precisa, funcionó como línea base confiable.

Fortalezas y limitaciones:

- RF es robusto y explicable, ideal para variables clínicas y señales PERG.
- MLP requiere ajuste fino pero capta patrones complejos.
- Limitaciones observadas en clases minoritarias, potencial de mejora con modelos basados en CNN.

Más allá de los números:

El uso combinado de variables clínicas como `age_years` y `va_logMAR` junto a señales PERG permitió la detección de patrones diagnósticos no triviales. La trazabilidad mediante `id_record` abre la posibilidad de segmentar modelos por subgrupos clínicos.

Conclusiones:

- Se logró construir y comparar tres modelos de clasificación sobre señales PERG y datos clínicos para predecir diagnóstico¹.
- Random Forest alcanzó el mejor desempeño (Accuracy = 0.9971, F1-Macro = 0.9976), seguido muy de cerca por MLPClassifier, mientras que Regresión Logística sirvió de línea base con métricas inferiores.
- La reproducibilidad del flujo quedó garantizada mediante exportación de artefactos (.pkl, .csv) y validación automática de rutas/metadatos.
- Limitaciones identificadas: desbalance de clases menores, necesidad de mayor afinamiento de hiperparámetros en MLP y falta de explotación de la estructura temporal completa de la señal PERG.

Trabajo Futuro:

- Explorar arquitecturas de redes neuronales profundas (CNN/RNN) que operen sobre series temporales crudas de PERG para capturar características dinámicas.
- Implementar técnicas de balanceo de clases (SMOTE, focal loss) y muestreo activo para mejorar la sensibilidad en diagnósticos minoritarios.
- Desarrollar un ensamble de modelos (stacking/blending) que combine Random Forest y MLP con métodos basados en deep learning.
- Ampliar el conjunto de datos con nuevas mediciones clínicas y validación prospectiva en cohortes independientes.

- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. Recuperado de <https://doi.org/10.1023/A:1010933404324>
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer. Recuperado de <https://link.springer.com/book/10.1007/978-0-387-84858-7>
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830. Recuperado de <https://jmlr.org/papers/v12/pedregosa11a/pedregosa11a.pdf>
- McInnes, L., Healy, J., & Melville, J. (2018). UMAP: Uniform Manifold Approximation and Projection for dimension reduction. *Journal of Open Source Software*, 3(29), 861. Recuperado de <https://doi.org/10.21105/joss.00861>

Referencias II

- Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3), 90–95. Recuperado de <https://doi.org/10.1109/MCSE.2007.55>
- McKinney, W. (2010). Data structures for statistical computing in Python. En *Proceedings of the 9th Python in Science Conference* (pp. 51–56). Recuperado de <https://proceedings.scipy.org/articles/Majora-92bf1922-00a>
- Calpa, I. E. (2025). PERG-IOBA dataset (Version 1.0.0) [Data set]. Universidad Nacional de Colombia – Sede Manizales. Recuperado de <https://physionet.org/content/perg-ioba-dataset/1.0.0/>
- Álvarez-Meza, A. M. (s. f.). AprendizajeMaquina [Repositorio en GitHub]. Recuperado de <https://github.com/amalvarezme/AprendizajeMaquina>
- Calpa, I. E. (2025). PERG-IOBA project codes and notebooks [Carpeta en Google Drive]. Recuperado de https://drive.google.com/drive/folders/1ziCclYmsfYZQ_lnQWOLsMlAtsLQno84o

- Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2021). Dive into Deep Learning (Version 1.0) [Libro en línea]. Recuperado de <https://d2l.ai/>