Aprendizaje Automático para la Priorización de Candidatos a Análisis de Pedigrí Forense

Juan José H. Beltrán[1], Alejandro Martínez Rivera[2], Cristopher Eduardo Ascencio Cruz[2], Miranda Fabiola Córdova Mercado[3], Mayra Edduardoff[4], Ángel David Reyes Figueroa[3,5].

[1]Tecnológico de Monterrey; [2]Tecnológico Nacional de México, Campus Veracruz; [3]Centro de Investigación en Matemáticas, A.C.; [4]Sam Houston State University; [5]Secretaría de Ciencia, Humanidades, Tecnología e Innovación.

I. Introducción

Año con año, las **desapariciones en México** siguen en aumento, con un total de **128,064 personas desaparecidas** hasta mayo de 2025[1]. Ante esta realidad, resulta fundamental **desarrollar estrategias** que permitan *identificar y distinguir* relaciones de parentesco en las búsquedas en bases de datos genéticos.

El **objetivo** es proponer modelos de **aprendizaje automático** para identificar verdaderas **relaciones de paternidad** entre individuos no relacionados en análisis por pares. Esta aproximación permitirá **priorizar candidatos para estudios de pedigrí**, los cuales son procesos complejos, lentos y con alta demanda computacional, contribuyendo así al fortalecimiento de las capacidades en la identificación de personas desaparecidas.

II. Marco Teórico

Los análisis por pares (**pairwise comparisons**) consisten en comparar, uno a uno, los perfiles genéticos de **individuos no identificados** —como restos humanos— con los perfiles genéticos de **posibles familiares**, tales como padres, madres o hermanos, con el fin de evaluar la evidencia genética de una relación biológica. Cada comparación requiere calcular un cociente de verosimilitud (**likelihood ratio**, LR) específico para el tipo de parentesco evaluado. Este enfoque se utiliza como etapa **preliminar al análisis de pedigrí**, que permite reconstruir relaciones familiares más complejas.

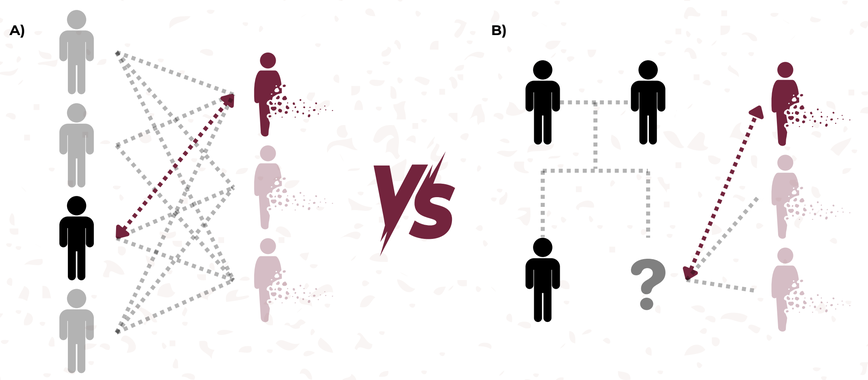

Figura 1. El análisis por pares (A) permite obtener resultados preliminares rápidos sobre grandes bases de datos, mientras que el análisis de pedigrí (B) permite obtener resultados mucho más precisos y determinantes, aunque en tiempos exponencialmente mayores. Habitualmente se utilizan complementariamente.

III. Metodología

Los datos analizados consistieron en los valores de **likelihood ratio** (LR) obtenidos bajo la hipótesis de paternidad, considerando una **tasa de mutación del 1%** entre perfiles genéticos simulados y reales. Los pares fueron etiquetados como verdaderos o falsos según la relación biológica conocida. Además, se incluyeron características adicionales como el número de **marcadores compartidos**, el número de **alelos compartidos**, y el número de marcadores en los que ambos individuos compartían uno o dos alelos. Se obtuvieron 763 pares verdaderos y 16,287 pares falsos en los datos simulados, y 778 verdaderos frente a 16,821 falsos en los datos reales

Input $\{\text{LR}, \text{Marcadores}, \text{Alelos}, \text{Loci}\}$ —— Modelo de M.L. ⟶ **Output** $\{\text{Paternidad}\}$

III.1. Generación de Conjuntos de Entrenamiento y Prueba

Con los datos simulados se generaron conjuntos de entrenamiento y prueba para **clasificación binaria** entre paternidad y no paternidad, con el objetivo de comparar el desempeño de distintos modelos bajo condiciones controladas. Se realizó una partición aleatoria y **balanceada** del conjunto de entrenamiento mediante **submuestreo**. En contraste, los datos reales se utilizaron para evaluar la aplicabilidad de los modelos en un entorno forense, ya que reflejan condiciones de incertidumbre propias de los casos reales.

III.2. Generar Tercera Etiqueta Mediante Vecinos Cercanos

Los datos mostraron **baja separabilidad** en el espacio de características (ver Figura 3-A), lo que limita el desempeño de los modelos. Ante esta situación de decidió agregar una **tercera etiqueta** para clasificar ciertos casos como **indeterminados** (ver Figura 3-B.), para los puntos en zonas donde no existe clara distinción entre clases. Se clasificó como indeterminado a todo punto que, en su ε-vecindad, tiene **vecinos de la clase opuesta**.

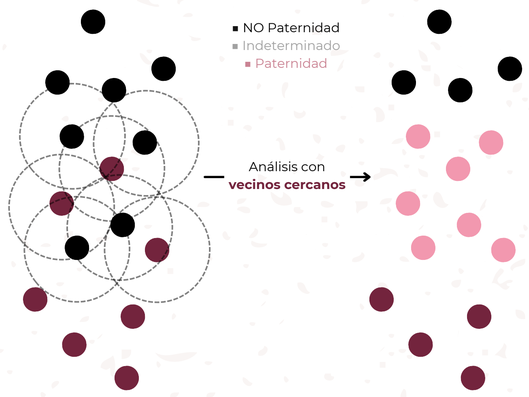

Figura 2. Ejemplo teórico del algoritmo de vecinos cercanos. Los puntos en áreas con distribución homogénea formaran parte de la nueva clase.

III.3. Entrenamiento y Prueba de Modelos

Utilizando los conjuntos tanto de clasificación binaria como ternaria, se experimentó con modelos de M.L. y se evaluó su desempeño. Los modelos utilizados fueron **regresión logística**, ***random forest*** y ***XGBoost***. Se consideran como métricas relevantes para este problema **f1-score, recall** y **precisión** sobre paternidad.

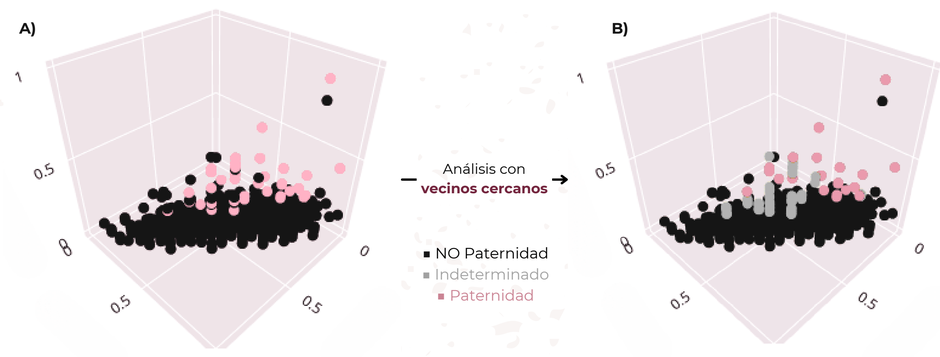

Figura 3. Distribución de los datos en una proyección del espacio de características a 3 dimensiones (LR, loci 1 y loci 2). A la izquierda se muestran las dos clases originales. A la derecha se incluye también la tercera clase "indeterminada" que se calculó después de un análisis de los vecinos cercanos de cada punto.

IV. Resultados y Discusión

Tras las pruebas realizadas con los modelos utilizando un conjunto de datos simulados para clasificación binaria, se obtuvieron **métricas notablemente buenas** en todos los casos. Sin embargo, se identificó un umbral de desempeño que los modelos no lograban superar, incluso después de ajustar los **hiperparámetros**. Este límite se atribuyó a la **baja separabilidad** de los datos, incluyendo una alta tasa de falsas paternidades marcadas como verdaderas.

Reformular la tarea como una **clasificación ternaria**, permitió reducir significativamente la tasa de error y mejorar las métricas generales. Entre los modelos evaluados, *random forest* (ver Tabla 2) fue el que alcanzó los mejores resultados.

Clasificación Binaria			
Modelo	**Precision**	**Recall**	**F1**
Logistic Regression	0.84	0.92	0.88
Random Forest	0.85	0.93	0.88
XGBoost	0.84	0.94	0.87
Support Vector Machine (SVM)	0.85	0.93	0.88

Tabla 1. Métricas obtenidas de los modelos a través del entrenamiento con el ajuste de hiperparámetros mediante *grid search* con 2 clases (Parentesco, No parentesco) evaluados con datos simulados.

Clasificación Ternaria			
Modelo	**Precision**	**Recall**	**F1**
Logistic Regression	0.47	0.47	0.46
Random Forest	0.97	0.98	0.98
XGBoost	0.9	0.93	0.91
Support Vector Machine (SVM)	0.9	0.92	0.91

Tabla 2. Métricas obtenidas de los modelos a través del entrenamiento con el ajuste de hiperparámetros mediante *grid search* con 3 clases (Parentesco, No parentesco e indeterminados) evaluados con datos simulados.

Una vez identificado ***random forest*** como el modelo con el mejor desempeño, se evaluó utilizando el conjunto de datos reales. Para ello, se comparó su **matriz de confusión** obtenida con datos reales frente a la generada con datos simulados, con el objetivo de analizar su comportamiento en un entorno más cercano a la realidad.

Se observa que, en los datos simulados, el modelo logra una clasificación efectiva, evitando en gran medida confundir parentescos con no parentescos y viceversa. Además, la tasa de error en la clasificación es aceptable, manteniéndose una buena precisión en la identificación de relaciones de parentesco.

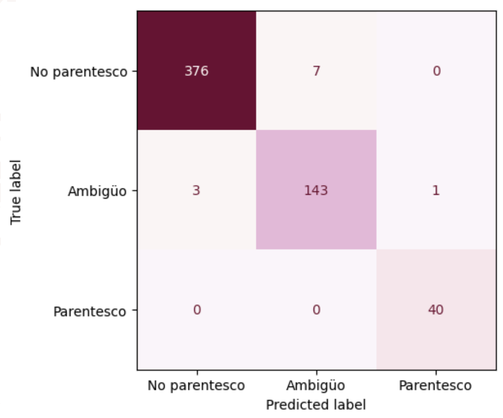

Figura 4. Matriz de confusión para el modelo de *random forest* con el ajuste de hiperparámetros utilizando datos simulados.

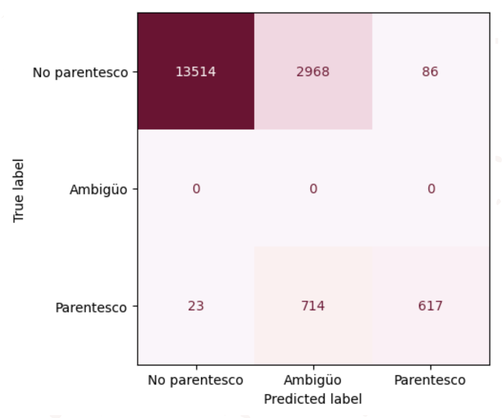

Figura 5. Matriz de confusión para el modelo de *random forest* con el ajuste de hiperparámetros utilizando datos reales.

V. Conclusión

De acuerdo con los resultados obtenidos, se concluye que, en el caso de los datos etiquetados con dos clases (*Parentesco y No parentesco*), los modelos de aprendizaje automático presentaron métricas similares, alcanzando un límite común en su desempeño. Sin embargo, al agregar una clase extra (*Indeterminados*) a los datos, se observó una mejora significativa en el modelo de ***random forest***, el cual logró una **puntuación casi perfecta** en la resolución de esta problemática (ver Tabla 4). En términos de rendimiento, se logró una **mejora de hasta un 22.5%**, además de un avance notable en cuanto al tiempo de procesamiento de los datos debido a la correcta clasificación, **disminuyendo el trabajo en un *66.6%*.** A pesar de la disparidad existente en los datos reales, el modelo de Random Forest destacó por encima de los demás (ver Figuras 4 y 5). Se mostró la capacidad de estos modelos para aprender a identificar parentescos, lo cual, dada su naturaleza, abre las puertas a **focalizar** los esfuerzos en la revisión de la clase indeterminada, y **agilizando la priorización** de candidatos para el análisis de pedigrí forense.

Referencias

Aprendizaje Automático para la Priorización de Candidatos a Análisis de Pedigrí Forense

Juan José H. Beltrán[1], Alejandro Martínez Rivera[2], Cristopher Eduardo Ascencio Cruz[2], Miranda Fabiola Córdova Mercado[3], Mayra Edduardoff[4], Ángel David Reyes Figueroa[3,5].

[1]Tecnológico de Monterrey; [2]Tecnológico Nacional de México, Campus Veracruz; [3]Centro de Investigación en Matemáticas, A.C.; [4]Sam Houston State University; [5]Secretaría de Ciencia, Humanidades, Tecnología e Innovación.

Referencias

1. IMDHD. (2025). *Informe Nacional de Personas Desaparecidas 2025*. Instituto Mexicano de Derechos Humanos y Democracia. Disponible en: https://imdhd.org/redlupa/informes-y-analisis/informes-nacionales/informe-nacional-de-personas-desaparecidas-2025/
2. *Strategies for pairwise searches in forensic kinship analysis*. Forensic Science International: Genetics, 54 (2021).
3. *Population data of 24 strs in mexican-mestizo population from Monterrey, Nuevo Leon (northeast, Mexico) based on powerplex® fusion and globalfiler® kits*. Forensic Science International: Genetics, 21 (2016).

Anexos

1. Repositorio de Git Hub. Contiene los datos procesados, notebooks y modelos entrenados. Disponible en https://github.com/JuanjoBelt/cimat-pedigri-forense
2. Información de contacto. Juan José H. Beltrán; *juanjobelt@outlook.com*.