Machine Learning y su vinculación con Algoritmos Genéticos para la planificación óptima de cultivos

Resumen

La planificación espacial de cultivos presenta alta incertidumbre por variabilidad climática, heterogeneidad edáfica y volatilidad de precios. Este trabajo propone un enfoque híbrido que integra aprendizaje automático (ML) en la función objetivo de un algoritmo genético (AG) para asignar superficies entre siete cultivos (girasol, soja, maíz, trigo, sorgo, cebada, maní), maximizando el ingreso esperado. El pipeline combina: series históricas y proyecciones a 14 meses de variables climáticas (temperatura, humedad, viento, precipitación), atributos de suelo por departamento, y precios por tonelada por cultivo. A partir de estas fuentes, se entrena un modelo supervisado para estimar rendimiento (ton/ha) y se incorpora esa señal económica (rendimiento por precio) en el fitness del AG. Presentamos la arquitectura del sistema, decisiones de diseño y operadores evolutivos (selección por ruleta/torneo, cruce de 1 punto, mutación del tipo swap y elitismo). Los resultados preliminares muestran convergencia estable y asignaciones coherentes con el escenario agroclimático proyectado, evidenciando la utilidad de la sinergia ML-AG para decisiones agrícolas robustas.

Palabras Clave

Algoritmos genéticos - Aprendizaje automático - Predicción climática - Optimización agrícola - Rendimiento de cultivos

1. Introducción

En los últimos años, la evolución de la capacidad de cómputo ha permitido la modelización de diferentes escenarios del mundo físico. En este trabajo se busca aprovechar dicha evolución para encontrar soluciones óptimas al evaluar múltiples características de la planificación agrícola en el contexto de la Argentina.

En este marco, los **algoritmos genéticos (AG)**, por su capacidad para abordar problemas de optimización con múltiples variables y restricciones, permiten explorar configuraciones eficientes en sistemas agrícolas diversificados. Por su parte, con **técnicas de *machine learning*** se pueden obtener modelos predictivos entrenados sobre datos históricos, útiles para anticipar el comportamiento de los sistemas bajo las diferentes situaciones posibles en la agricultura.

La articulación entre ambos enfoques —optimización evolutiva y predicción basada en datos— permite avanzar hacia modelos de planificación agrícola más adaptables y contextualizados. Este trabajo explora dicha integración, con el objetivo de contribuir al desarrollo de soluciones computacionales que apoyen la toma de decisiones en unidades productivas reales, contemplando tanto la eficiencia como la sustentabilidad del sistema.

Cabe señalar que, en el presente trabajo, se ha decidido acotar deliberadamente el alcance del modelo propuesto. En particular, se omiten variables relacionadas con la incidencia de enfermedades, la presencia de plagas y los niveles de contaminación ambiental o química. Esta decisión metodológica responde a la necesidad de abordar inicialmente la problemática desde una perspectiva simplificada, que permita focalizar el análisis en aspectos estructurales de la planificación agrícola, tales como la selección de cultivos, la distribución espacial y temporal, y la consideración de factores geográficos y climáticos.

Situación problemática

En el contexto actual de la agricultura argentina, la planificación eficiente de la producción se ha vuelto una tarea cada vez más compleja. Esta planificación no solo abarca la selección de cultivos, sino también su disposición espacial dentro de las parcelas. Las decisiones en torno a estos aspectos deben considerar simultáneamente múltiples factores: las características del suelo, la variabilidad climática, la disponibilidad de agua y las condiciones económicas del entorno productivo.

Pese a esta complejidad, en muchos casos las

decisiones agronómicas aún se toman con base en la experiencia previa del productor o en recomendaciones generales que no captan las particularidades de cada parcela ni se adaptan dinámicamente a condiciones cambiantes. Esto puede derivar en subutilización del potencial productivo, manejo ineficiente de recursos (especialmente agua y nutrientes del suelo) y pérdida de sustentabilidad del sistema agropecuario en el largo plazo.

Problema

¿Cómo aplicar modelos de machine learning combinados con algoritmos genéticos para optimizar la planificación agrícola, considerando variables interdependientes como clima, suelo y siembra de diferentes semillas en simultáneo?

Objetivo General

Desarrollar un modelo de optimización para la planificación espacial de cultivos en parcelas agrícolas, utilizando algoritmos genéticos y técnicas de machine learning.

Objetivos Específicos

- Construir un pipeline de datos: clima mensual histórico y proyectado; suelo por departamento; precios por tonelada.
- Definir la función objetivo económica (ingreso esperado) e integrarla en el AG.
- Desarrollar una interfaz GIS para capturar área y ubicación del campo.
- Realizar el ajuste de hiperparámetros de cada modelo de red neuronal para que el entrenamiento se ajuste a los datos de la mejor manera.
- Entrenar modelos de ML para estimar rendimiento (ton/ha) por cultivo bajo clima proyectado.
- Combinar el algoritmo genético con las redes neuronales para que, mediante las predicciones de éstas últimas, el AG pueda optimizar la utilización del área seleccionada.

2. Marco Teórico

Para el desarrollo del prototipo se utiliza una lista de siete cultivos (soja, maíz, trigo, maní, sorgo, cebada y girasol). La elección se basó en la disponibilidad y calidad de datos provenientes de fuentes confiables como la Bolsa de Cereales y Productos de Bahía Blanca [2] y la Bolsa de Cereales de Córdoba [1], que ofrecen series históricas y actualizaciones consistentes de precios por tonelada y reportes de mercado. Además, para la recopilación de estos datos se implementó web scraping; por lo tanto, se necesitaban fuentes que publicaran tablas consolidadas con varios cultivos en una misma página para automatizar la extracción y la normalización diaria. En consecuencia, no fue posible relevar cualquier cultivo arbitrario: restringimos el análisis a aquellos con presencia estable y formateo homogéneo en dichas fuentes, garantizando consistencia y trazabilidad.

Para la selección del área de estudio, los puntos clave fueron la utilización de GIS (Geographic Information Systems). Con esta interfaz web, el usuario dibuja la parcela; luego se procesa la geometría (validación del polígono,cálculo de área en m^2 o ha) y se determina el departamento/provincia, para unir las propiedades de suelo correspondientes. Este contexto espacial asegura que las variables climáticas y de la composición del suelo utilizadas por los modelos reflejen la ubicación real del lote.

Para la proyección climática se utiliza dos enfoques complementarios: LSTM[10][11] para variables con mayor componente secuencial (temperatura y humedad) y Gradient Boosting Regressor (GBR)[6][7] para variables con patrones no lineales en formato tabular (viento y precipitación). Ambos generan pronósticos de 14 meses a futuro, que se combinan con los atributos de suelo por departamento para construir el dataset de rendimiento esperado por cultivo. Finalmente, esas estimaciones de rendimiento se multiplican con los precios obtenidos mediante web scraping para definir la función objetivo económica del algoritmo genético, que optimiza la asignación de hectáreas entre los cultivos considerados. Los da-

tos utilizados para la proyección del clima se obtienen de la API NASA POWER[3].

Web scraping

El web scraping es la extracción automatizada de información publicada en sitios web para convertirla en datos estructurados. En nuestro caso, se lo utiliza para obtener precios por tonelada de los cultivos analizados desde páginas que publican varios cultivos en una misma vista. Esta condición es clave para que la extracción sea estable, rápida y normalizable de forma diaria. Por este motivo, no se recuperarn precios de cualquier cultivo arbitrario ni de páginas con formatos cambiantes: se restringe a fuentes confiables con estructura uniforme (p. ej., pizarras/reportes de bolsas regionales).

LSTM

Las redes neuronales recurrentes (RNN)[11] están diseñadas para modelar secuencias (series temporales, lenguaje, señales). Las RNN "simples" suelen sufrir desvanecimiento/explosión del gradiente al intentar capturar dependencias de largo plazo. Para mitigar este problema, Hochreiter y Schmidhuber[10] introdujeron las Long Short-Term Memory (LSTM): celdas con estado interno y tres compuertas (entrada, olvido, salida) que regulan qué información se incorpora, qué se olvida y qué se expone en cada paso. Matemáticamente, estas compuertas se implementan con activaciones sigmoide (para filtrar la información importante) y tangentes hiperbólicas (para obtener un resumen de la información). Por su capacidad de retener contexto, las LSTM son ampliamente usadas en predicción de series temporales con estacionalidad y dependencias de media y larga duración.

Generar proyecciones climáticas de hasta 14 meses a futuro. Estas proyecciones alimentan el dataset de rendimiento por cultivo y, en consecuencia, la función objetivo del algoritmo genético.

Formación de secuencias. A partir de la serie mensual, se construyen ventanas deslizantes: un bloque histórico reciente (longitud ω) se usa como entrada y la red aprende a predecir $H = 14$ meses hacia adelante. Una vez predicho el valor del mes próximo se lo agrega al bloque histórico de longitud ω como el valor del último mes, y se elimina el valor del primer mes; de esta forma es como se va "deslizando"la ventana de valores y se puede predecir el valor correspondiente al mes siguiente del próximo. Este método se repite hasta obtener los valores de los 14 meses siguientes al actual.

Arquitectura y entrenamiento

- **Arquitectura**: 1–2 capas LSTM seguidas de una capa densa con H neuronas (una por mes futuro).
- **Pérdida y métricas**: entrenamiento con el promedio del cuadrado del error (MSE)[4][5].
- **Early stopping**: es una técnica de regularización que interrumpe el entrenamiento de una red neuronal cuando el error en un conjunto de validación deja de mejorar, para controlar sobreajuste.
- **Reproducibilidad**: semillas fijas, registro de versiones y esquema de columnas persistente.

Salida e integración. El modelo devuelve, para cada variable (temperatura y humedad), un vector de 14 meses proyectados. Estas proyecciones se ensamblan con:

- Datos del suelo por departamento.
- Los datos del cultivo.
- Los climáticos restantes que predicen dos modelos diferentes de GBR (explicados más adelante).

Con ello se forma el conjunto de características que consume el modelo de rendimiento (ton/ha) por cultivo. Luego, el algoritmo genético combina esas tasas con la superficie propuesta para predecir las toneladas que se esperan cosechar. Estas toneladas se multiplican por los precios para calcular el ingreso esperado y optimizar la asignación de hectáreas.

GBR

El Gradient Boosting Regressor (GBR)[7] es un método de aprendizaje supervisado basado en boosting[6]. La idea central es construir un modelo aditivo fuerte sumando, de manera secuencial, varios modelos de redes neuronales débiles, pero, a su vez, más flexibles. En este caso se utilizan árboles de decisión poco profundos[8][9] (aprendices débiles). En cada iteración, el nuevo árbol se ajusta para corregir los errores del ensamble: formalmente, se avanza en la dirección del gradiente negativo de la función de pérdida (p. ej., MSE[4][5] en regresión). Con una tasa de aprendizaje pequeña (es el tamaño del paso con que se actualizan los parámetros en cada iteración del entrenamiento) y varios estimadores (`n_estimators`), el modelo captura no linealidades e interacciones de forma efectiva, controlando el sobreajuste con hiperparámetros como la cantidad máxima de ramas que tendrá cada árbol (`max_depth`) y la cantidad mínima de hojas que debe tener (`min_samples_leaf`).

En la proyección climática para variables como la velocidad del viento y precipitaciones, al menos en período mensuales, tienden a perder la secuencialidad y volverse variables aleatorias. Se empieza de la serie diaria obtenida mediante API NASA POWER[3], se transforma dichos datos a meses y se construye una columna para representar el valor obtenido en el mes anterior, indicando que el valor de tal mes depende de cuánto se obtuvo en el mes previo (se le simula secuencialidad). Con ese set el GBR produce un vector de 14 meses futuros por variable.

También, el modelo principal de predicción de toneladas por cultivo es GBR, pero para su entrenamiento se utiliza un dataset con los valores historicos de cada semilla, con los datos del suelo de cada departamente y los datos climáticos previos. El GBR aprende la relación no lineal para que cuando se le pase los datos de las semillas a predecir más los datos del clima a futuro, pueda predecir las toneladas esperadas por semilla.

Validación y métricas

Durante el entrenamiente se monitoriza el MSE[5] y se aplica early stopping sobre validación para que no haya un sobreajuste.

Integración de RN en AG

El GBR climático produce datos de viento y precipitaciones para los 14 meses proyectados. Esas proyecciones se ensamblan con temperatura y humedad proyectadas (LSTM), datos del suelo por departamento y cultivo codificado, para construir el feature set del modelo de rendimiento. El GBR de rendimiento devuelve las toneladas cosechadas por cultivo. Luego, el AG calcula la FO económica con $toneladas_i \times precio_i$ y optimiza la asignación de hectáreas.

3. Metodología

El proyecto busca determinar la distribución óptima de siembra en una parcela dada, maximizando la rentabilidad del productor a partir de precios de mercado y rendimiento esperado bajo clima proyectado. A continuación se detalla la obtención de datos, con el nivel de información necesario para su reproducción.

Utilización del GIS

Al iniciar el sistema, el usuario delimita la parcela sobre un mapa interactivo (GIS) dibujando un polígono. Con la selección confirmada, el sistema calcula el área en hectáreas. Luego, se determina departamento mediante el cálculo del centroide (centro geométrico de un objeto) del polígono.

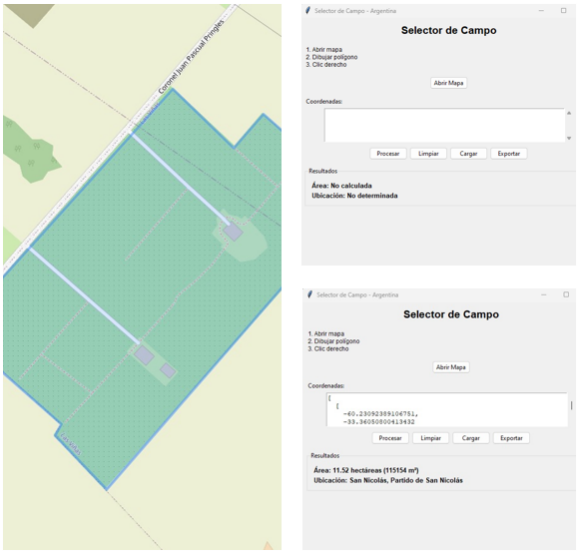

Figura 1: Pantalla de interacción con el usuario del selector de campo.

Los datos recuperados por la aplicación de la Figura 1 permiten unir atributos de suelo por departamento y parametrizar la descarga climática por las coordenadas (latitud y longitud) del centroide.

Recuperación y procesamiento de datos

Con el centroide, se consultan series mediante la API NASA POWER [3]. Se descargan registros diarios y se los tranforma a promedios mensuales para construir el histórico en el sitio de estudio. Las variables estudiadas son temperatura, humedad, viento y precipitaciones.
Los datos de las semillas y los suelos son recuperados de las APIs proporcionadas por el gobierno de la nación[4], luego los archivos fueron trabajados mediantes merges e inner joins (formas de unir las tuplas de entre tablas de archivos), por departamento, nombre de cultivo o año, para tener toda la información necesaria para entrenar el modelo. Luego se realiza una depuración de las columnas con más del 85 porciento de los datos faltantes y las tuplas (filas) que contengan datos faltantes, ya que interfieren en la utilización de redes neuronales. Esto se debe a que éstas, de forma resumida, son relaciones entre parámetros, relaciones que no se pueden formar si es que hay faltante de uno de ellos.
Además, es necesario transformar el nombre de los cultivos a valores numéricos enteros, ya que el GBR no interpreta cadenas de caracteres. La transformación fue posible mediante la creación de un diccionario, el cual traduce cada nombre de cultivo a un valor entero.
Los precios son recuperados diariamente mediante web scraping. Gracias a esto se puede trabajar con valores contemporáneos de precios por toneladas.

Modelado y proyección climática

Se emplean dos familias de modelos, según la naturaleza de la variable:

- **LSTM** para temperatura y humedad.
- **GBR** para viento y precipitaciones, se toma el dato del mes anterior para predecir el del actual.

Ambos modelos generan vectores de 14 elementos, uno por cada mes predicho, por variable y se persisten como artefactos para reutilización (evitando reentrenar).

Ejecución del algoritmo genético

El AG opera sobre siete cultivos. Cada individuo es un vector real de longitud 7, donde cada elemento del vector (o gen) corresponde a las hectáreas a sembrar por cultivo, sujeto a:

$$\sum_{i=1}^{7} x_i = A \quad \text{y} \quad x_i \geq 0, \tag{1}$$

donde A es el área total de la parcela recuperada por el GIS.
Operadores y parámetros:

- **Inicialización**: población aleatoria.
- **Selección**: torneo o ruleta.
- **Cruce**: 1 punto.
- **Mutación**: del tipo swap, es decir, se intercambian dos genes al azar.
- **Elitismo**: conservación de las mejores soluciones.

Función objetivo
Para cada individuo se arma, por cultivo, una fila de inferencia con cultivo codificado, datos del suelo y clima proyectado. El modelo de rendimiento devuelve las toneladas de esa semilla que se esperarían cosechar. Esas toneladas ($\hat{r}_i$) se multiplican por el precio por tonelada para cada semilla (p_i). Ahora se tiene un vector con la ganancia que produce cada semilla para ese individuo, por lo tanto, se suman las ganancias y se obtiene una ganancia total, que se utiliza como valor objetivo del individuo. Dicho valor es el que se busca maximizar a lo largo de las generaciones.

$$\text{FO} = \sum_{i=1}^{7} (\hat{r}_i\,[\text{ton}] \times p_i\,[\$/\text{ton}]) \tag{2}$$

4. Resultados

En esta sección se reportan los resultados de la evaluación predictiva (clima y rendimiento) y de la optimización con el algoritmo genético.

Modelos climáticos

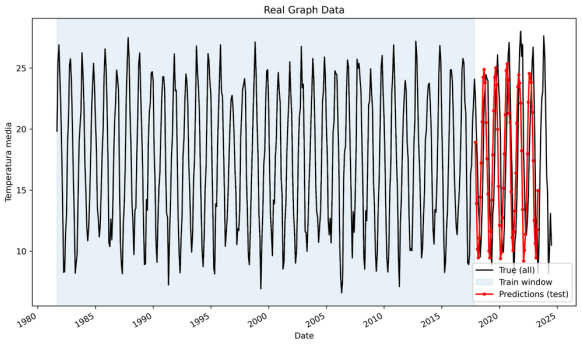

Figura 2: Temperatura media: real vs. predicho (test), $H = 14$.

En la Figura 2 se representa con un fondo sombreado los datos que son utilizados para el entrenamiento del modelo LSTM que predice la temperatura media en grados celsius. Luego, en rojo, están graficados los valores obtenidos mediante predicciones realizadas con los valores reales (gráfica con línea negra) que ”no habia visto nunca.[es] decir con valores que no fueron utilizados para entrenarlo. Como se puede observar las predicciones son bastante fieles a los valores reales.

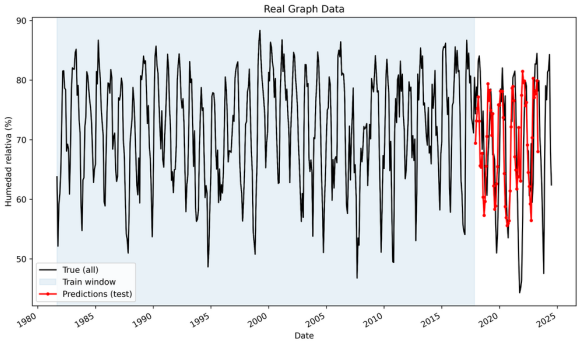

Figura 3: Humedad relativa: real vs. predicho (test), $H = 14$.

Al igual que en la Figura 2, en la Figura 3 se puede apreciar que se utiliza la primer gran parte de datos para el entreno de este modelo, y el último 15 porciento se lo destina al testeo del mismo. Para la misma forma de entrenamiento, ahora, los valores predichos no son tan fieles a los valores reales, esto se da que por más que los hiperparámetros hayan sido de mucha ayuda para el primer modelo, para éste puede que se necesite otra combinación de valores.

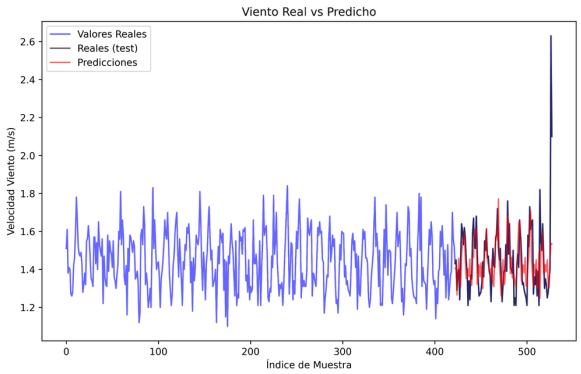

Figura 4: Viento: real vs. predicho (test), $H = 14$.

En la Figura 4, ya se observa la predicción con menos error entre las realizadas (las lineas rojas son las predicciones y la azul los datos reales). Ésta corresponde a la red neuronal GBR, la cual no se basaba en la secuencia de datos para predecir.

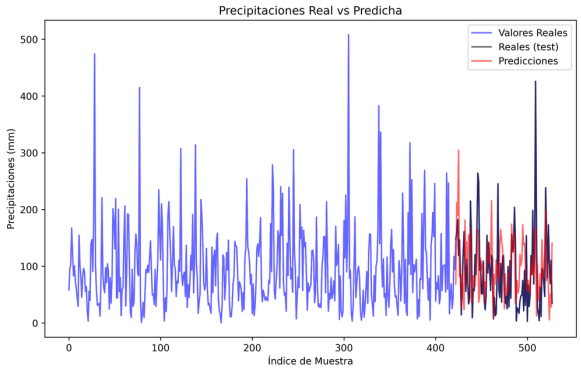

Figura 5: Precipitación: real vs. predicho (test), $H = 14$.

La Figura 5 es representación de que por más que se hayan utilizado los mismos hiperparámetros que para el modelo GBR mencionado en la Figura 4, ya sea por la unidad de medición o por una diferente distribución de datos, se realizó una predicción no tan convincente como las anteriores. Esto se soluciona con varias horas de procesamiento realizando un Hyperparameter Tunning.

Modelo de rendimiento

Anteriormente se realizó hincapié en la importancia de la correcta creación de los archivos con datos de entrada y salida para que el modelo principal sea entrenado una sola vez. Esto es lo que se busca para acortar los tiempos de uso del sistema, si no tardaría horas por cada corrida.

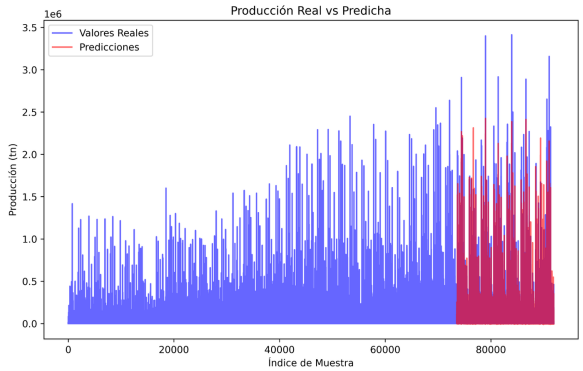

Figura 6: Producción: real vs. predicha en el conjunto de test.

La Figura 6 contrasta la producción estimada con la serie real. El modelo reproduce las variaciones de escala y la tendencia general; las discrepancias locales se asocian a picos de precipitación y a la variabilidad interanual.

Optimización con AG

A continuación se pordrán observar los resultados obtenidos luego de 1000 corridas del AG.

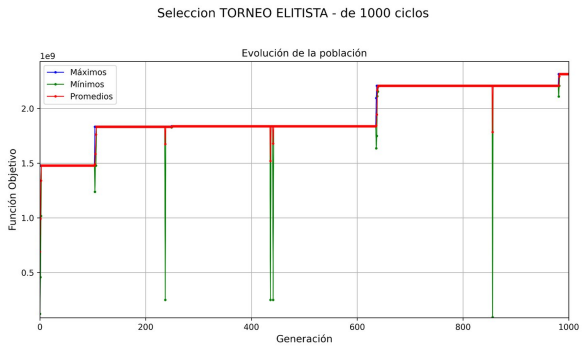

Figura 7: Convergencia del algoritmo genético: mejor/promedio/peor por generación.

La Figura 7 muestra la evolución de la población de individuos y cómo ésta va mejorando, es decir un mayor valor de la función objetivo, a medida que pasan más iteraciones. Como es una corrida elitista, donde los mejores individuos se saltean el proceso de combinación y el de mutación, éstos no se pierden. Los demás individuos si sufren estos cambios y es por ello que se observan picos descendentes verdes. La linea verde corresponde a los individuos con menor valor de FO, la azul a los máximos y la roja al promedio de la población.

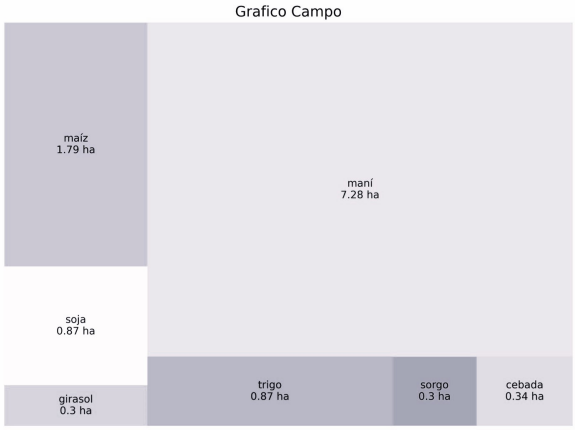

Figura 8: Distribución de hectáreas por cultivo en la solución final (treemap).

La solución final se resume en la Figura 8, que presenta la distribución de hectáreas por cultivo del mejor individuo alcanzado en la última iteración. En la corrida base, el maní domina la asignación, seguido por maíz, con menores áreas para trigo, soja, sorgo, cebada y girasol, consistente con el ingreso esperado dado el rendimiento estimado y los precios vigentes.

5. Conclusión y Trabajos Futuros

En este trabajo se abordó la problemática de la planificación agrícola en un contexto de alta complejidad, integrando predicciones climáticas, modelos de aprendizaje automático y algoritmos genéticos para la optimización espacial de cultivos. La propuesta permite capturar la interacción entre factores edáficos, climáticos y económicos de manera más realista que con enfoques tradicionales, y sentó las bases para un esquema de planificación flexible y adaptativo.

Los resultados muestran que la combinación de predicciones climáticas y modelos de machine learning mejoran la estimación de rendimientos, mientras que los algoritmos genéticos facilitan la búsqueda de configuraciones de cultivo más eficientes en términos de ingreso esperado y aprovechamiento de recursos. Asimismo, la interfaz GIS constituye un aporte práctico, acercando la herramienta al usuario final.

Sin embargo, el estudio enfrentó una limitación importante: la escasez de datos económicos detallados sobre algunas semillas (por ejemplo, cultivos menos difundidos o con escasa trazabilidad en mercados locales). Esta carencia restringió la capacidad del modelo para evaluar alternativas productivas de manera integral, ya que el ingreso esperado no pudo calcularse con la misma precisión para todos los cultivos considerados. De este modo, el análisis económico resultó más robusto en los granos principales que en las opciones secundarias.

En futuras líneas de investigación será esencial ampliar la base de datos incorporando precios históricos y proyecciones más completas de todos los cultivos relevantes, junto con indicadores de costos asociados. Además, deberían explorarse enfoques multiobjetivo que ponderen no sólo los beneficios económicos, sino también criterios ambientales y sociales.

En síntesis, el trabajo confirma el potencial de las herramientas de inteligencia artificial y optimización para transformar la toma de decisiones en la producción agropecuaria, aunque también deja en claro que la disponibilidad y calidad de los datos —especialmente los económicos— constituye un factor crítico para alcanzar recomendaciones más equitativas, precisas y sustentables.

Referencias

[1] Bolsa de Cereales de Córdoba. (s. f.). *Todas las pizarras*. Recuperado de https://www.bccba.org.ar/todas-las-pizarras/

[2] Bolsa de Cereales y Productos de Bahía Blanca. (s. f.). *Informes*. Recuperado de https://www.bcp.org.ar/informes.asp

[3] NASA. (2025). *POWER Data Access Viewer (API)*. Recuperado de https://power.larc.nasa.gov/

[4] Datos Argentina. (2025). *Datasets de semillas, departamentos y suelos argentinos*. Recuperado de https://datos.gob.ar/dataset

[5] Wikipedia. (2025). *Gradient boosting*. Recuperado de https://en.wikipedia.org/wiki/Gradient_boosting

[6] scikit-learn. (2025). *Ensembles: Gradient boosting, random forests, bagging, voting, stacking*. Recuperado de https://scikit-learn.org/stable/modules/ensemble.html

[7] Neptune.ai. (2025). *Gradient Boosted Decision Trees (Guide): A Conceptual Explanation*. Recuperado de https://neptune.ai/blog/gradient-boosted-decision-trees-guide

[8] Toprak, M. (2020). *Gradient Boosting and Weak Learners*. Recuperado de https://medium.com/@toprak.mhmt/gradient-boosting-and-weak-learners-1f93726b6fbd

[9] Machine Learning Mastery. (2021). *Strong Learners vs. Weak Learners in Ensemble Learning*. Recuperado de https://machinelearningmastery.com/strong-learners-vs-weak-learners-for-ensemble-learning/

[10] Hochreiter, S., y Schmidhuber, J. (1997). *Long Short-Term Memory*. Neural Computation, 9(8), 1735–1780. Recuperado de https://doi.org/10.1162/neco.1997.9.8.1735

[11] Le, X.-H., Ho, H. V., Lee, G., y Jung, S. (2019). *Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting*. Water, 11(7), 1387. Recuperado de https://doi.org/10.3390/w11071387