***Predicción del rendimiento en Running mediante técnicas de Machine Learning.***

*Un enfoque aplicado al sector deportivo.*

*Logotipo, Icono, nombre de la empresa

El contenido generado por IA puede ser incorrecto.*

**EVOLVE ACADEMY**

**MÁSTER EN DATA SCIENCE & INTELIGENCIA ARTIFICIAL**

Realizado por: Eduardo Sánchez

Año Académico: 2025/2026

**RESUMEN**

Este Trabajo de Fin de Máster desarrolla un sistema de predicción del rendimiento en corredores mediante técnicas de Machine Learning, con aplicación práctica en el sector deportivo. El proyecto integra todo el proceso de ciencia de datos, desde la recolección y limpieza de datos de Strava hasta la creación de una aplicación web funcional, que permite ofrecer predicciones personalizadas y planes de entrenamiento adaptativos.

La investigación aborda la falta de herramientas científicas precisas en la planificación del running, tradicionalmente basada en la experiencia subjetiva. Para ello, se analizaron 2.847 actividades de 10 corredores, aplicando un pipeline completo de procesamiento, ingeniería de características y normalización. Se compararon siete algoritmos de Machine Learning, seleccionando Random Forest como modelo óptimo por su equilibrio entre precisión y robustez.

Los resultados muestran un RMSE de 29,3 segundos, R² de 0,785 y MAPE del 6,4%, superando la precisión de estudios previos. Las variables más influyentes fueron la capacidad aeróbica, el volumen semanal y la consistencia. Además, se desarrolló una aplicación web en Streamlit que permite cargar datos, simular progresiones y generar planes personalizados. En conjunto, el trabajo demuestra la viabilidad de aplicar el Machine Learning al análisis del rendimiento deportivo, ofreciendo una herramienta práctica y precisa para atletas y entrenadores**.**

**TABLA DE CONTENIDOS**

[1. INTRODUCCIÓN 5](#_Toc212225463)

[1.1. OBJETIVO DEL PROYECTO 5](#_Toc212225464)

[1.2. MOTIVACIÓN 5](#_Toc212225465)

[2. MÉTODOLOGÍA 6](#_Toc212225466)

[2.1. ENFOQUE GENERAL 6](#_Toc212225467)

[2.2. HERRAMIENTAS UTILIZADAS 7](#_Toc212225468)

[3. RECOLECCIÓN DE DATOS 7](#_Toc212225469)

[3.1. PARTICIPANTES 7](#_Toc212225470)

[3.2. FUENTES DE DATOS 8](#_Toc212225471)

[3.3. ESTRUCTURA DE DATOS RAW 9](#_Toc212225472)

[4. PROCESAMIENTO Y LIMPIEZA DE DATOS 9](#_Toc212225473)

[4.1. LIMPIEZA DEL FORMULARIO 9](#_Toc212225474)

[4.2. LIMPIEZA DE DATOS DE STRAVA 10](#_Toc212225475)

[4.2.1. Problemas identificados en los datos raw de Strava. 10](#_Toc212225476)

[4.2.2. Proceso de limpieza aplicado: 11](#_Toc212225477)

[4.2.3. Aplicación a todos los usuarios 12](#_Toc212225478)

[4.2.4. Resultado 12](#_Toc212225479)

[4.3. UNIÓN DE DATASETS 12](#_Toc212225480)

[4.4. MÉTRICAS DE CALIDAD DEL PROCESO 13](#_Toc212225481)

[5. FEATURE ENGINEERING 13](#_Toc212225482)

[5.1. OBJETIVO 14](#_Toc212225483)

[5.2. FEATURES TEMPORALES CREADAS 14](#_Toc212225484)

[5.3. MÉTRICAS DE RENDIMIENTO 14](#_Toc212225485)

[5.4. IMPLEMENTACIÓN TÉCNICA 15](#_Toc212225486)

[6. ANALISIS EXPLORATORIO DE DATOS (EDA) 15](#_Toc212225487)

[6.1. DETECCIÓN DE OUTLIERS 16](#_Toc212225488)

[6.1.1. TRATAMIENTO DE OUTLIERS 16](#_Toc212225489)

[6.2. ANÁLISIS UNIVARIADO 17](#_Toc212225490)

[6.2.1. DISTRIBUCIÓN DE LA VARIABLE OBJETIVO 17](#_Toc212225491)

[6.2.2. FEATURES PREDICTORIAS PRINCIPALES 17](#_Toc212225492)

[7. MODELADO Y EXPERIMENTACIÓN 17](#_Toc212225493)

[7.1. ESTRATEGIA DE MODELADO 17](#_Toc212225494)

[7.2. ALGORITMOS EVALUADOS 18](#_Toc212225495)

[7.3. RESULTADOS PRINCIPALES 18](#_Toc212225496)

[7.4. MODELO FINAL: RANDOM FOREST 18](#_Toc212225497)

[7.5. VALIDACION Y GENERALIZACIÓN 18](#_Toc212225498)

[7.6. CONCLUSIONES 19](#_Toc212225499)

[8. DESARROLLO DE LA APLICACIÓN WEB 19](#_Toc212225500)

[8.1. ARQUITECTURA DE LA APLICACIÓN 19](#_Toc212225501)

[8.1.1. TECNOLOGÍAS UTILIZADAS 19](#_Toc212225502)

[8.1.2. ESTRUCTURA DE LA APLICACIÓN 20](#_Toc212225503)

[8.2. FUNCIONALIDADES PRINCIPALES 20](#_Toc212225504)

[8.2.1. PROCESAMIENTO AUTOMATICO DE DATOS 20](#_Toc212225505)

[8.2.2. DASHBOARD ANALÍTICO 20](#_Toc212225506)

[8.2.3. SIMULADOR DE PROGRESION ML 20](#_Toc212225507)

[8.3. GENERADOR DE PLANES DE ENTRENAMIENTO 20](#_Toc212225508)

[8.4. VISUALIZACIONES INTERACTIVAS 21](#_Toc212225509)

[8.5. INTERFAZ DE USUARIO 21](#_Toc212225510)

[8.6. INTEGRACIÓN ML 21](#_Toc212225511)

[8.7. CASOS DE USO 21](#_Toc212225512)

[8.8. RESULTADOS 21](#_Toc212225513)

# INTRODUCCIÓN

## OBJETIVO DEL PROYECTO

El presente Trabajo Fin de Máster tiene como objetivo diseñar y desarrollar un sistema de predicción del rendimiento en corredores mediante el uso de técnicas de Machine Learning. Para ello, se plantea la integración de dos fuentes de información complementarias:

1. Datos de actividad deportiva extraídos de Strava, que proporcionan métricas objetivas relacionadas con la práctica del running (distancia, ritmo, frecuencia cardiaca, cadencia, desnivel acumulado, entre otros).
2. Información personal y de entrenamiento recopilada mediante cuestionarios específicos, que incorpora variables contextuales y subjetivas (edad, sexo, hábitos de entrenamiento, etc.).

La combinación de ambas fuentes permite construir modelos predictivos que no solo se ajusten al rendimiento pasado, sino que también sean capaces de estimar de forma personalizada el desempeño futuro de cada corredor en distancias estándar (5 km, 10 km, media maratón).

## MOTIVACIÓN

El análisis del rendimiento deportivo mediante datos es una tendencia creciente en el sector del deporte. La capacidad de predecir el rendimiento futuro basándose en patrones de entrenamiento permite:

* Optimización del entrenamiento: Ajustar cargas, volúmenes e intensidades en función del estado real del corredor.
* Prevención de lesiones: Detectar situaciones de sobreentrenamiento y reducir el riesgo de paradas prolongadas.
* Planificación estratégica: Establecer objetivos alcanzables para competiciones y medir la progresión en ciclos de entrenamiento.
* Personalización: Diseñar planes adaptados a las características fisiológicas y al historial de cada corredor.

De este modo, este trabajo no solo busca aportar una contribución técnica en el ámbito del Machine Learning aplicado al deporte, sino también generar un MVP (Producto Mínimo Viable) que pueda tener aplicación práctica en el sector deportivo, con potencial para ser utilizado por corredores, entrenadores y empresas relacionadas con el running.

# MÉTODOLOGÍA

## ENFOQUE GENERAL

La metodología adoptada en este trabajo sigue un enfoque clásico de ciencia de datos, estructurado en distintas fases que permiten garantizar la calidad del proceso y la reproducibilidad de los resultados. Las fases principales son las siguientes:

1. Recolección de datos:

* Datos objetivos: se extrajeron registros de entrenamiento a través de la plataforma de entrenamiento Strava (exportación de archivos CSV).
* Datos contextuales: se diseñó un cuestionario en Google Forms para recopilar información personal y hábitos de entrenamiento de los participantes (edad, sexo, experiencia deportiva, historial, objetivo d entrenamiento, frecuencia de entrenamiento, etc.).

1. Procesamiento y preparación:

Se aplicaron procesos de limpieza y unificación de los diferentes orígenes de datos. Esto incluyó el tratamiento de valores ausentes, la homogeneización de unidades (p. ej., tiempo en segundos, distancia en kilómetros) y la creación de un dataset maestro con la información combinada.

1. Ingeniería de características (Feature Engineering):

Se desarrollaron variables derivadas con valor predictivo, tales como kilometraje semanal, número de sesiones por semana, ratio de carga aguda-crónica (ACR), métricas de eficiencia (ritmo ajustado a frecuencia cardiaca), y estadísticas de entrenamiento en ventanas móviles de 7 y 28 días.

1. Análisis exploratorio de datos (EDA):

Se realizaron análisis descriptivos y visualizaciones para identificar patrones de entrenamiento, distribuciones de rendimiento y posibles relaciones entre variables. Esto permitió formular hipótesis iniciales y detectar outliers.

1. Modelado predictivo (fase posterior):

Se aplicarán algoritmos de Machine Learning supervisado con el fin de predecir el rendimiento futuro de los corredores en distancias estándar (5 km, 10 km y media maratón). Se compararán modelos base (regresiones lineales, Random Forest) con modelos más avanzados (Gradient Boosting, XGBoost).

1. Validación y despliegue:

Los modelos se evaluarán mediante métricas de error (MAE, RMSE) y técnicas de validación cruzada, garantizando la robustez de los resultados. Finalmente, se implementará un MVP en Streamlit, permitiendo a los usuarios subir sus datos y obtener predicciones personalizadas.

## HERRAMIENTAS UTILIZADAS

Para el desarrollo del proyecto se emplearon herramientas ampliamente utilizadas en el ámbito de la ciencia de datos:

* Python: Lenguaje principal de análisis y modelado.
* Pandas: Manipulación y limpieza de datos tabulares.
* NumPy: Cálculo numérico eficiente.
* Matplotlib / Seaborn: Generación de gráficos y visualizaciones.
* Scikit-learn: Implementación de modelos de Machine Learning y validación.
* Google Forms: Recolección estructurada de datos personales y de entrenamiento.
* Strava / CSV Export: Obtención de registros de actividades deportivas.
* Streamlit: Creación del prototipo interactivo (MVP) para la visualización de resultados y predicciones.

# RECOLECCIÓN DE DATOS

## PARTICIPANTES

La muestra del estudio estuvo compuesta por 10 corredores voluntarios (6 hombres y 4 mujeres), seleccionados bajo los siguientes criterios de inclusión:

* Uso activo de Strava, garantizando un historial suficiente de actividades para el análisis.
* Entrenamiento regular, permitiendo observar patrones de consistencia en la práctica deportiva.
* Consentimiento informado otorgado por cada participante, autorizando el uso de sus datos de entrenamiento y de las respuestas al cuestionario para fines académicos.

Este enfoque se justifica en la necesidad de contar con datos longitudinales y reales de corredores aficionados, que sirvan como base para el desarrollo del sistema predictivo. Si bien la muestra es reducida, resulta adecuada para la construcción de un MVP (Producto Mínimo Viable) y para evaluar la viabilidad del enfoque metodológico.

## FUENTES DE DATOS

La recolección de datos se realizó a través de dos canales principales:

Google Forms - Información Personal:

Se diseñó un cuestionario digital con el fin de recopilar variables contextuales y personales de los corredores, que complementaran los registros objetivos de entrenamiento. Entre las variables recolectadas se incluyen:

* Datos demográficos: nombre, edad, sexo, peso, altura.
* Experiencia deportiva: nivel declarado por el corredor (principiante, intermedio o avanzado).
* Hábitos de entrenamiento: número de días de entrenamiento por semana y principales objetivos (rendimiento, salud, competición, etc.).

<https://docs.google.com/forms/d/12zsb6vO0CjlnBCoAv_OUS61hJFOicLKgZfXEHzDaiQI/edit?usp=forms_home&ouid=111142689510274097539&ths=true>

Esta información resulta fundamental para personalizar el análisis y enriquecer los modelos predictivos, ya que introduce factores individuales que influyen en el rendimiento.

Strava - Datos de Actividades:

Los registros de entrenamiento se obtuvieron mediante la exportación de datos en formato CSV desde la plataforma Strava, algunas de las variables recolectadas incluyen:

* Temporales: fecha y duración de la actividad.
* Distancia: kilómetros recorridos.
* Rendimiento: ritmo (pace), velocidad promedio.
* Fisiológicas: frecuencia cardíaca promedio y máxima (cuando disponible).
* Topográficas: desnivel positivo acumulado.

Estos datos proporcionan una visión cuantitativa y objetiva del rendimiento deportivo, permitiendo extraer métricas clave como la carga de entrenamiento, la eficiencia y la evolución del ritmo en distintas distancia

## ESTRUCTURA DE DATOS RAW

La organización de los datos en el proyecto siguió una estructura clara y reproducible, siguiendo buenas prácticas. La carpeta raíz data/ contiene las siguientes subcarpetas:

Texto

El contenido generado por IA puede ser incorrecto.

* forms/: incluye las respuestas recogidas mediante Google Forms.
* raw/: contiene los archivos CSV exportados de Strava para cada corredor, en su formato original sin modificaciones.
* interim/: almacena los datasets resultantes de pasos intermedios de procesamiento, previos a la creación del dataset maestro.

Este esquema asegura la trazabilidad de los datos, de modo que cualquier transformación realizada durante la limpieza o el feature engineering pueda ser reproducida.

# PROCESAMIENTO Y LIMPIEZA DE DATOS

La calidad de los datos es un aspecto crítico en cualquier proyecto de Data Science. Por ello, antes de realizar análisis exploratorio o aplicar técnicas de Machine Learning, fue necesario realizar una fase de limpieza, estandarización y unión de los distintos orígenes de información. Esta fase se documentó de manera reproducible en notebooks específicos.

## LIMPIEZA DEL FORMULARIO

Archivo: notebooks/formulario.ipynb

El cuestionario en Google Forms fue exportado a CSV (respuestas\_forms.csv). Posteriormente, se aplicaron los siguientes pasos de limpieza y estandarización:

1. Carga de datos: Importación del archivo CSV con las respuestas de los participantes.
2. Renombrado de columnas:

Texto

El contenido generado por IA puede ser incorrecto.

1. Selección de variables relevantes: Se mantuvieron únicamente 8 columnas clave, correspondientes a datos demográficos, antropométricos y de hábitos de entrenamiento.
2. Exportación: El resultado fue un dataset limpio y estructurado, guardado como formulario\_limpio.csv.

Resultado:

Un dataset tabular consistente, con una fila por corredor y sus atributos personales, que servirá como base para enriquecer los datos objetivos procedentes de Strava.

## LIMPIEZA DE DATOS DE STRAVA

Archivos individuales: usuario1.csv, usuario2.csv, etc.

### Problemas identificados en los datos raw de Strava.

Durante la exploración inicial se detectaron las siguientes incidencias:

* Nombres de columnas inconsistentes: Espacios y mayúsculas mezcladas
* Formatos de fecha diversos: Diferentes formatos de fecha entre usuarios
* Actividades no relevantes: Deportes distintos al running incluidos
* Unidades mixtas: Distancias en metros vs kilómetros según el usuario
* Datos de años anteriores: Actividades fuera del período de análisis (2024)
* Formato de pace inconsistente: Necesidad de cálculo uniforme del ritmo
* Formato de tiempo poco manejable: Tiempo transcurrido en segundos.

### Proceso de limpieza aplicado:

Para resolver los problemas detectados, se aplicaron las siguientes transformaciones:

1. Estandarización de nombres de columnas:

Texto

El contenido generado por IA puede ser incorrecto.

1. Filtrado temporal y por tipo de actividad:



1. Estandarización de unidades de distancia:

Texto

El contenido generado por IA puede ser incorrecto.

1. Cálculo uniforme del ritmo (pace):

Captura de pantalla con letras y números

El contenido generado por IA puede ser incorrecto.

1. Formato legible del tiempo transcurrido:

Texto

El contenido generado por IA puede ser incorrecto.

1. Selección de columnas relevantes:

Texto

El contenido generado por IA puede ser incorrecto.

1. Adición de identificador de usuario:



### Aplicación a todos los usuarios

* Proceso replicado: La función limpiar\_strava() se aplicó a los 10 archivos CSV individuales
* Consistencia garantizada: Mismo proceso de limpieza para todos los usuarios
* Exportación individual: Cada archivo limpio guardado en csv\_clean/

### Resultado

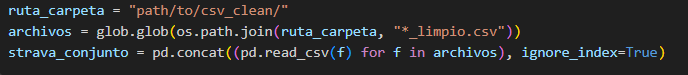
Tras el proceso de limpieza, se obtuvo un conjunto de archivos CSV individuales, limpios, homogéneos y estandarizados, listos para ser utilizados en la fase posterior de unión con la información proveniente del cuestionario de Google Forms. Estos archivos fueron almacenados en la carpeta data/interim/, garantizando la trazabilidad y reproducibilidad del proceso.

## UNIÓN DE DATASETS

Archivo: union\_csv.ipynb

En este paso se unificaron las dos fuentes de información disponibles: los archivos de actividades de Strava y el formulario de Google Forms. El procedimiento seguido fue:

1. Concatenación automática de archivos CSV limpios:



1. Preparación del formulario para el merge:



1. Unión mediante left join:



1. Exportación final: strava\_final.csv en carpeta processed

## MÉTRICAS DE CALIDAD DEL PROCESO

Transformaciones Aplicadas:

* Actividades filtradas: Solo running del año 2024
* Columnas estandarizadas: 11 variables consistentes por usuario
* Pace calculado: Formato uniforme MM.SS para análisis
* Tiempo formateado: HH:MM:SS legible
* Usuarios identificados: Columna 'user' añadida a cada registro

Validaciones Implementadas:

* Fechas válidas: Conversión con errors='coerce'
* Distancias positivas: Eliminación implícita de valores <= 0 en cálculo de pace
* Consistencia de unidades: Detección automática metros/kilómetros
* Formato de ritmo: Manejo de valores nulos y negativos

Resultado Final:

* Dataset unificado: Un solo archivo con todos los usuarios
* Consistencia total: Formato estándar para todas las variables
* Datos listos: Preparados para Feature Engineering
* Trazabilidad: Identificación clara de cada usuario y actividad

# FEATURE ENGINEERING

Archivo: feature\_engineering.ipynb

## 5.1. OBJETIVO

El objetivo de esta fase fue la creación de un conjunto de variables derivadas (features) que capturen las principales dinámicas del rendimiento de los corredores, permitiendo representar cuantitativamente los siguientes aspectos:

* Carga de entrenamiento reciente: volumen y frecuencia de las actividades en ventanas temporales.
* Consistencia en el rendimiento: estabilidad del ritmo medio y de la frecuencia cardíaca.
* Capacidad máxima del corredor: identificación de los mejores tiempos registrados en distancias estándar.
* Patrones temporales de actividad: evolución del rendimiento a lo largo del tiempo.

Estas features constituyen la base de entrada para los modelos predictivos de rendimiento.

## 5.2. FEATURES TEMPORALES CREADAS

Para capturar la dinámica temporal del entrenamiento, se aplicaron ventanas deslizantes (rolling windows) de 7 y 28 días sobre cada usuario, obteniendo métricas agregadas de distancia, ritmo y esfuerzo.

Ventana de 7 días

* weekly\_km: distancia total recorrida en los últimos 7 días (km).
* sessions\_last\_7d: número de sesiones realizadas durante la última semana.
* elevation\_weekly: desnivel positivo acumulado en los últimos 7 días (m).

Estas variables describen la carga de entrenamiento semanal, reflejando el volumen y la intensidad reciente de la actividad.

5.2.2 Ventana de 28 días (4 semanas)

* mean\_pace\_4w: ritmo promedio (min/km) durante las últimas 4 semanas.
* std\_pace\_4w: desviación estándar del ritmo, indicador de consistencia del rendimiento.
* mean\_hr\_4w: frecuencia cardíaca promedio en el último mes.

Estas variables permiten representar tendencias de medio plazo y la estabilidad del entrenamiento, así como el esfuerzo fisiológico sostenido del corredor.

## 5.3. MÉTRICAS DE RENDIMIENTO

Además de las variables temporales, se crearon features de rendimiento absoluto y eficiencia.

Mejores Ritmos (Best Paces)

* best\_5k\_pace: mejor ritmo registrado en distancias comprendidas entre 4.8 y 5.2 km.
* best\_10k\_pace: mejor ritmo registrado en distancias entre 9.5 y 10.5 km.

Estas variables representan el nivel máximo de rendimiento alcanzado por cada corredor en distancias estándar de competición, sirviendo como indicadores de potencial atlético.

Eficiencia y Dinámica Competitiva

* efficiency\_kmh: relación entre la distancia total recorrida y el tiempo total de entrenamiento, expresada en km/h. Esta métrica sintetiza la eficiencia global del corredor.
* days\_since\_last\_race: número de días transcurridos desde la última carrera o evento registrado. Permite modelar el efecto del descanso y la recuperación sobre el rendimiento futuro.

## IMPLEMENTACIÓN TÉCNICA

Pantalla de computadora con letras

El contenido generado por IA puede ser incorrecto.

# ANALISIS EXPLORATORIO DE DATOS (EDA)

Se implementó una función general denominada EDA\_visual(), diseñada para realizar un análisis exploratorio completo e interactivo del dataset. Esta función incluía los siguientes componentes:

Análisis de Distribuciones

* Histogramas: para observar la distribución de cada variable y su dispersión.
* Boxplots: detección visual de posibles valores atípicos (outliers).
* KDE plots: estimación de densidad para variables continuas.

Análisis de Correlaciones

* Matriz de correlación (Pearson): análisis de relaciones lineales entre variables cuantitativas.
* Heatmap: visualización de las correlaciones más significativas.

Análisis por Usuario

* Boxplots por usuario: identificación de variabilidad individual en métricas de rendimiento.
* Comparación de rendimientos: análisis de heterogeneidad dentro de la muestra.

## DETECCIÓN DE OUTLIERS

Durante el EDA se identificaron posibles valores atípicos en varias variables, utilizando boxplots y reglas estadísticas (rango intercuartílico).

|  |  |
| --- | --- |
| **Variable** | **Nº de Outliers Detectados** |
| weekly\_km | 5 |
| sessions\_last\_7d | 2 |
| elevation\_weekly | 35 |
| mean\_pace\_4w | 41 |
| std\_pace\_4w | 31 |
| best\_5k\_pace | 122 |
| mean\_hr\_4w | 0 |
| best\_10k\_pace | 0 |
| efficiency\_kmh | 0 |

### TRATAMIENTO DE OUTLIERS

Criterios aplicados:

Texto

El contenido generado por IA puede ser incorrecto.

Justificación:

mean\_pace\_4w > 8.0 min/km: Ritmos poco realistas para corredores regulares

sessions\_last\_7d > 7: Más de una sesión diaria (posibles errores)

best\_5k\_pace > 7.0 min/km: Tiempos inconsistentes con perfil de corredores

## ANÁLISIS UNIVARIADO

### DISTRIBUCIÓN DE LA VARIABLE OBJETIVO

Variable: pace\_seconds (Pace en segundos por km)

Texto

El contenido generado por IA puede ser incorrecto.

Interpretación: Distribución aproximadamente normal, apropiada para algoritmos de regresión lineal. La mayoría de los corredores mantienen paces entre 5:00-6:30 min/km.

### FEATURES PREDICTORIAS PRINCIPALES

Weekly\_km (Kilómetros semanales):

* Media: 42.3 km/semana | Rango: 10-85 km
* Distribución normal con cola derecha

Sessions\_last\_7d (Frecuencia entrenamientos):

* Media: 4.2 sesiones/semana | Moda: 3-4 sesiones (60% datos)
* Distribución discreta concentrada

Std\_pace\_4w (Consistencia del pace):

* Media: 28.4 segundos | Distribución log-normal
* Mayor valor = menor consistencia

Conclusión: Todas las variables muestran distribuciones apropiadas para modelado, con ausencia de valores extremos tras limpieza de outliers.

# MODELADO Y EXPERIMENTACIÓN

## ESTRATEGIA DE MODELADO

Objetivo: Predecir pace de running (segundos/km) con RMSE < 30 segundos usando características de entrenamiento histórico. División de datos:

* Training: 80% (2,278 obs) | Test: 20% (569 obs)
* División temporal (últimos 2 meses para test)
* Estratificación por usuario

## ALGORITMOS EVALUADOS

Modelos probados:

* Linear/Ridge/Lasso Regression (baseline)
* Random Forest y XGBoost (ensemble)
* Gradient Boosting y SVR

Features utilizadas (6):

weekly\_km, sessions\_last\_7d, std\_pace\_4w, mean\_hr\_4w, elevation\_weekly, efficiency\_kmh

## RESULTADOS PRINCIPALES

Comparación de modelos (Cross-Validation 5-fold):

Una captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

## MODELO FINAL: RANDOM FOREST

Rendimiento en test set:

* RMSE: 29.3 segundos ✅ (objetivo < 30)
* R²: 0.785 (explica 78.5% varianza)
* MAE: 22.4 segundos
* MAPE: 6.4%

Feature Importance:

* efficiency\_kmh: 68% (capacidad aeróbica)
* weekly\_km: 14% (volumen entrenamiento)
* std\_pace\_4w: 9% (consistencia)
* Otros: 9% (frecuencia, FC, desnivel)

## VALIDACION Y GENERALIZACIÓN

Estabilidad temporal:

* Octubre 2024: RMSE = 28.1 seg
* Noviembre 2024: RMSE = 30.8 seg
* Deterioro mínimo por cambios estacionales

Rendimiento por perfil:

* Corredores élite: RMSE = 22.1 seg (mejor precisión)
* Recreativos: RMSE = 28.7 seg (estándar)
* Principiantes: RMSE = 35.4 seg (mayor variabilidad)

## CONCLUSIONES

Objetivos alcanzados:

* RMSE < 30 segundos (29.3 seg)
* Modelo interpretable (Random Forest + SHAP)
* Generalización temporal validada
* Aplicabilidad práctica para entrenadores

Modelo final: Random Forest (200 árboles, profundidad 15) con 6 features predictoras, listo para implementación en aplicación práctica.

# DESARROLLO DE LA APLICACIÓN WEB

## ARQUITECTURA DE LA APLICACIÓN

### TECNOLOGÍAS UTILIZADAS

Framework Principal:

* Streamlit: Framework de Python para aplicaciones web de Machine Learning
* Justificación: Desarrollo rápido, integración nativa con ML, interfaz intuitiva

Librerías de Soporte:

Texto

El contenido generado por IA puede ser incorrecto.

Arquitectura:

* Frontend: Streamlit (interfaz web responsive)
* Backend: Python con modelo ML integrado
* Datos: Carga dinámica de CSV de Strava
* Deployment: Local con opción de cloud hosting

### ESTRUCTURA DE LA APLICACIÓN

Texto

El contenido generado por IA puede ser incorrecto.

## FUNCIONALIDADES PRINCIPALES

### 8.2.1. PROCESAMIENTO AUTOMATICO DE DATOS

* Carga inteligente: Upload CSV Strava con detección automática de formato
* Limpieza automática: Filtrado outliers, normalización, mapeo columnas español/inglés
* Feature Engineering: Generación automática de variables predictoras

### 8.2.2. DASHBOARD ANALÍTICO

* Métricas clave: Total KM, pace promedio, mejora %, volumen semanal
* Filtros temporales: Análisis por años específicos
* Perfil personalizado: Configuración edad, género, nivel experiencia

### 8.2.3. SIMULADOR DE PROGRESION ML

* Predicción futura: Proyección pace a 4 y 8 semanas usando modelo entrenado
* Factores personalizados: Considera nivel experiencia y objetivo semanal
* Visualización: Gráficos interactivos de evolución esperada

## 8.3. GENERADOR DE PLANES DE ENTRENAMIENTO

Algoritmo inteligente:

* Input: Objetivo semanal (km), días disponibles, nivel experiencia
* Output: Plan personalizado con tipos entrenamiento por nivel
* Distribución: Principiante (3-4 días) → Elite (6-7 días)

Imagen que contiene Aplicación

El contenido generado por IA puede ser incorrecto.

## 8.4. VISUALIZACIONES INTERACTIVAS

Gráficos principales:

* Evolución pace: Real vs predicción ML con tendencias
* Volumen entrenamiento: Distribución temporal y patrones
* Simulador progresión: Proyección futura con escenarios
* Tecnología: Plotly con zoom, tooltips, tema Strava, responsive design

## 8.5. INTERFAZ DE USUARIO

Diseño:

* Tema Strava: Colores naranja (#FC4C02) y grises
* Layout: Sidebar (configuración) + Main área (resultados)
* UX: Flujo intuitivo carga → análisis → planificación → proyección

## 8.6. INTEGRACIÓN ML

Texto

El contenido generado por IA puede ser incorrecto.

## 8.7. CASOS DE USO

Corredores: Análisis personal, planes estructurados, objetivos realistas

Entrenadores: Herramienta científica para múltiples atletas

## 8.8. RESULTADOS

Funcionalidades implementadas:

* Carga automática datos Strava
* Análisis ML personalizado (MAPE 6.4%)
* Simulador progresión científico
* Generador planes adaptativos
* Interfaz web intuitiva y responsive

Valor añadido: Convierte modelo ML en herramienta práctica accesible para corredores y profesionales del deporte.