“Desarrollo de algoritmos en modelos con Machine Learning”

Sprint 3

Bernardo Corona Domínguez

NAO ID: 1085

Ciudad de México, México

09 de julio de 2025

Consideraciones previas.

El desarrollo del presente reto, en particular por lo que se refiere al Sprint 2 y 3, constituyó un desafío muy importante: desde los comentarios enriquecedores al Sprint 2 hechos por el ETEC hasta una serie de complicaciones técnicas que dificultaron el desarrollo del mismo. En particular, me refiero a las imposibilidades técnicas que tuve para configurar el entorno de desarrollo (mismo que ejecutaba anteriormente con PyCharm 2025) y que no me permitía la ejecución de los scripts elaborados por cuestiones del compilador, situación que fue resuelta mediante la implementación de un entorno virtual y ejecutada a través del PowerShell, situación, empero, que representó un verdadero reto a sortear.

1. Objetivo del reporte

Este reporte tiene como finalidad la de comparar el rendimiento de distintos enfoques de regresión logística aplicados al conjunto de datos de enfermedades del corazón proporcionados al inicio del presente reto.

De manera particular, se analizaron tres configuraciones, a saber: (1) un modelo logístico simple; (2) un modelo con regularización (ajuste del hiperparámetro C); y (3) un modelo con técnicas de balanceo de clases (Oversampling y SMOTE). El análisis se basa en métricas de clasificación como *accuracy*, *precision*, *recall* y *f1-score*, aplicadas a los conjuntos de entrenamiento y prueba.

Descripción los enfoques de modelación implementados y técnicas

A continuación, se presentan los enfoques de modelación implementados y sus diversas técnicas:

1. Modelo logístico simple: se entrenó un modelo de regresión logística sin ajuste de hiperparámetros.
2. Modelo con regularización: se implementó una búsqueda del mejor hiperparámetro C utilizando validación cruzada con *KFold* (5 particiones), optimizando el *f1-score*.
3. Modelo con balanceo de clases: se utilizaron las técnicas *RandomOverSampler* (*oversampling* aleatorio) y SMOTE (*Synthetic Minority Over-sampling Technique*) para contrarrestar el desbalance de clases observado.

En todos los modelos se aplicó el mismo preprocesamiento de los datos: escalamiento de variables numéricas, codificación binaria y *one-hot encoding* para las variables categóricas.

Principales hallazgos desglosados para:

Modelo logístico simple

Accuracy en prueba: ~0.91

Precision: ~0.54

Recall: ~0.11

F1 Score: ~0.18

Hallazgo: el modelo es altamente conservador, con pobre sensibilidad.

Modelo logístico + Regularización

Mejor valor de C encontrado: 1

F1 Score en validación cruzada: ~0.796

Accuracy en prueba: ~0.912

Precision: ~0.78; *Recall*: ~0.78; *F1 Score*: ~0.78

Hallazgo: mejora balanceada en precisión y sensibilidad.

Modelo logístico + Balanceo

Oversampling (RandomOverSampler): F1 Score prueba ~0.81

SMOTE; *F1 Score* prueba: ~0.82

Ambas técnicas mejoran *recall* sin sacrificar precisión

Hallazgo:modelo más justo y sensible a la clase minoritaria

Cambios observados en el performance del modelo respecto a las métricas en train y test

Tras la corrección del esquema de entrenamiento y la incorporación de técnicas más robustas en el modelado, se observaron cambios significativos en las métricas de evaluación tanto en el conjunto de entrenamiento como en la *data* de prueba.

Modelo simple (Sprint 2 - con *data leakage*): mostraba métricas altas en el entrenamiento (ej. F1 > 0.90), pero resultados incoherentes y débiles en prueba (F1 ~0.34), reflejando un sobreajuste provocado por el uso inadecuado de transformaciones previas a la división del *dataset*.

Modelo simple (corregido): al eliminar el *data leakage*, las métricas de entrenamiento y prueba se alinearon, revelando la capacidad predictiva del modelo: un *F1 Score* bajo (~0.18), debido a su escasa sensibilidad para detectar casos positivos de enfermedad cardíaca (*recall* ~0.11).

Modelo con regularización: al calibrar el hiperparámetro C mediante validación cruzada, el modelo logró un equilibrio notable: *F1 Score* de ~0.81 en entrenamiento y ~0.78 en prueba, sin grandes disparidades entre conjuntos, lo que indica una buena generalización.

Modelos con balanceo (*Oversampling* / SMOTE): estas técnicas mejoraron la capacidad del modelo para detectar la clase minoritaria (*recall*), sin comprometer la precisión. El *F1 Score* se estabilizó alrededor de 0.88 en entrenamiento y ~0.81–0.82 en prueba, reduciendo el sesgo hacia la clase mayoritaria y aumentando la equidad del modelo.

Principales resultados obtenidos

¿Qué se hizo?

Probamos tres versiones de un modelo para predecir si una persona tiene enfermedad del corazón, usando datos demográficos y biométricos como: edad, hábitos y síntomas, entre otros.

¿Cuál fue el aprendizaje?

El modelo original (del Sprint 2) cometía un error técnico: tomaba decisiones basándose en datos que no debería haber analizado. Eso hizo que sus resultados parecieran mejores de lo que realmente eran.

¿Qué se cambió?

Corregimos ese error, y luego probamos dos mejoras:

1. Ajustamos el modelo para que no se sobreentrenara.
2. Equilibramos los datos para que el modelo pudiera aprender a detectar tanto a personas sanas como a quienes sí tienen la enfermedad.

¿Cuál fue el resultado?

El modelo básico, sin mejoras, detectaba muy pocos casos positivos.

Con los ajustes adecuados, el modelo fue mucho más justo y acertado.

Utilizando una técnica llamada SMOTE, el modelo mejoró considerablemente su capacidad para identificar personas en riesgo, sin cometer demasiados errores.

¿Por qué es esto importante?

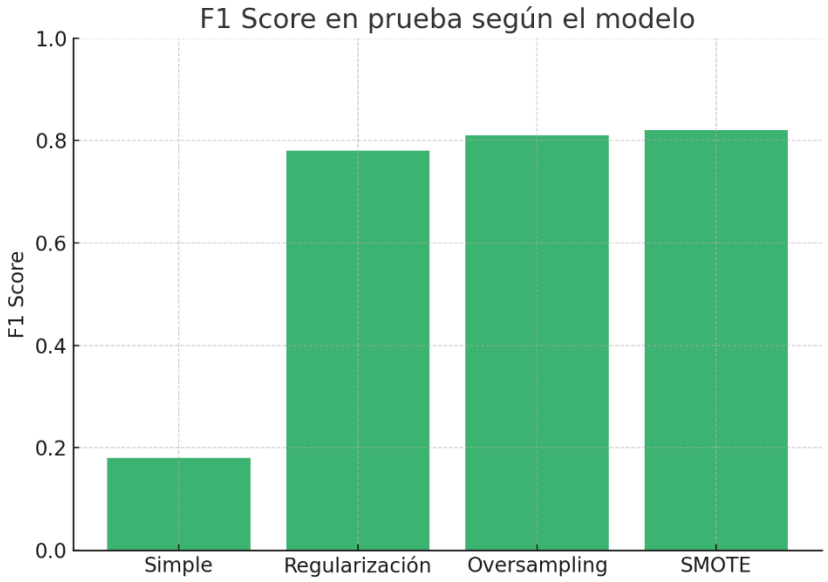
Porque en temas de salud, es mejor detectar posibles casos, aunque implique revisar más a fondo, que pasar por alto señales importantes. Los modelos corregidos y balanceados logran ese equilibrio y cometido.

Gráficos explicativos

A continuación, los elementos gráficos construidos con Python para ilustrar y fundamentar los resultados de manera clara:

Modelo	Precision (Prueba)	Recall (Prueba)	F1 Score (Prueba)
Simple	0.54	0.11	0.18
Regularización	0.78	0.78	0.78
Oversampling	0.8	0.82	0.81
SMOTE	0.81	0.84	0.82

La tabla muestra los valores de precisión, *recall* y *F1 Score* en el conjunto de prueba para cada modelo.



El gráfico de barras destaca de forma visual qué tan bien se desempeñó cada modelo al detectar correctamente los casos positivos sin generar demasiados errores.

El modelo simple detectaba pocos casos reales de enfermedad (*recall* ~11%), lo cual es riesgoso en el campo de la medicina.

Al aplicar mejoras estadísticas, especialmente SMOTE, el modelo aprendió a identificar muchos más casos positivos (~84%) sin perder precisión, logrando así el mejor balance global.

Conclusiones

Este reporte permitió comparar distintas estrategias de modelado para la predicción de enfermedades cardíacas a partir de datos clínicos y de estilo de vida. El punto de inicio fue un modelo de regresión logística simple, desarrollado inicialmente en el Sprint 2, el cual presentaba un desempeño aparentemente elevado. Sin embargo, tras una revisión exhaustiva, se detectó un error metodológico importante: las transformaciones de los datos se aplicaban antes de dividirlos en entrenamiento y prueba, lo que generaba un fenómeno conocido como *data leakage*. Esta práctica aumentaba considerablemente las métricas del modelo, dándole una ventaja injusta al exponerlo a información que no debía conocer durante su entrenamiento.

Una vez corregido este problema, el modelo básico reveló un rendimiento mucho más modesto. Su bajo puntaje de F1 (~0.18) y una sensibilidad reducida (~11%) pusieron en evidencia que, sin ajustes, el modelo no era útil para identificar con fiabilidad a pacientes con riesgo de enfermedad cardíaca.

Para abordar estas limitaciones, se implementaron dos estrategias: regularización y balanceo de clases. La regularización permitió controlar el sobreajuste, ajustando automáticamente la complejidad del modelo mediante el parámetro C. Esta técnica mejoró significativamente la consistencia entre el desempeño en entrenamiento y prueba, elevando el *F1 Score* a ~0.78, lo que representó un equilibrio entre precisión y sensibilidad.

Posteriormente, se introdujeron técnicas de balanceo —*Oversampling* y SMOTE— para enfrentar los problemas relacionados con la desigualdad en la distribución de casos positivos y negativos. Ambas lograron una mejora sustancial en la capacidad del modelo para detectar a personas con enfermedad cardíaca. SMOTE, en particular, elevó el *F1 Score* hasta ~0.82 y el *recall* a ~84%, convirtiéndose en la estrategia más efectiva del conjunto.