



PROYECTO “Patient Survival Prediction”

Miguel Angel Urueña Riobo miguel.uruena@udea.edu.co

C.C 1.006.121.797

Gaia Ramirez Hincapíe gaia.ramirez@udea.edu.co

C.C 1.005.273.358

Tutor:

Raul Ramos Pollan raul.ramos@udea.edu.co

INTRODUCCIÓN A LA INTELIGENCIA ARTIFICIAL

CÓDIGO: 2508401 - INGENIERÍA DE SISTEMAS V4

Nombre del Proyecto: Patient Survival Prediction

Links Github:

- https://github.com/uruenariobo/ai4eng_health_issues
- https://github.com/gaiamilenium99/ai4eng_health_issues1

1. Descripción del problema predictivo a resolver:

Dadas las características de un paciente y su historia clínica vamos a predecir la probabilidad de supervivencia de dicho paciente.

2. Dataset a utilizar:

Vamos a usar el dataset de kaggle “Patient Survival Prediction” ([enlace](#)), que tiene 91.713 número de muestras (pacientes) y 85 columnas, de las cuales las columnas más representativas son:

❖ Edad.



- ❖ Índice de masa corporal.

- ❖ Género.

- ❖ Peso.

- ❖ Presión sanguínea

- ❖ Ritmo cardíaco máximo.

- ❖ La ubicación del paciente antes de su ingreso en la unidad.

- ❖ El APACHE IV (es una predicción probabilística de la mortalidad hospitalaria del paciente).

3. Métricas de desempeño requeridas (de machine learning y de negocio):

Nuestro modelo de predicción de la supervivencia en pacientes debería de tener un porcentaje de acierto $>80\%$, pero también un false negative rate $<10\%$, ya que es un dictamen grave y delicado, por lo cual es preferible no fallar una predicción de un paciente que verdaderamente morirá, aunque eso implique que aumente el número de falsos positivos.

4. Criterio sobre cuál sería el desempeño deseable en producción:

Si la ocupación hospitalaria de las unidades de cuidados intensivos no aumentan más de un 10% no merece la pena poner el modelo en operatividad ya que el coste de desarrollo y mantenimiento no cubriría las ganancias adicionales de ese aumento.

5. Iteraciones de desarrollo:

Preprocesado de datos y exploración descriptiva del Dataset:

El dataset que seleccionamos para el proyecto contenía 91713 registros y 85 campos, considerando que esperamos obtener un modelo de predicción de la supervivencia de distintos pacientes ingresados a centros hospitalarios en diferentes situaciones médicas, es importante la precisión en la información por lo que no pudimos utilizar métodos para completar datos faltantes cómo se aprendió en clase.

Para realizar el filtro de esta información se eliminaron campos que no tenían importancia en lo que se desea predecir, cómo la identidad del paciente o del hospital, o incluso de la icu. También se eliminaron datos que podrían

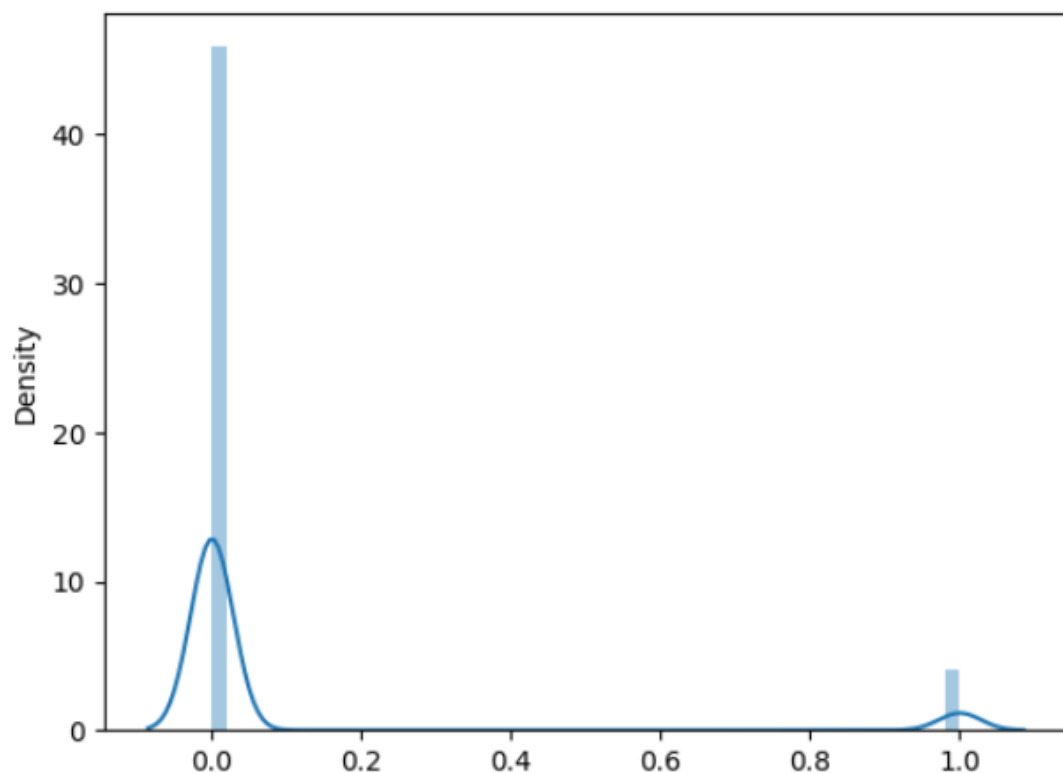


presentar relación o dependencia con otras variables ya contenidas en el modelo, éstas corresponden a algunas de los campos relacionados con Apache (luego de revisar qué información representaban, y observar una posible relación).

Luego de tener este filtro, el dataset pasó a tener 71 campos, donde realizamos una eliminación de los registros nulos en el dataset quedando con 62498 registros, como se observa en el colab.

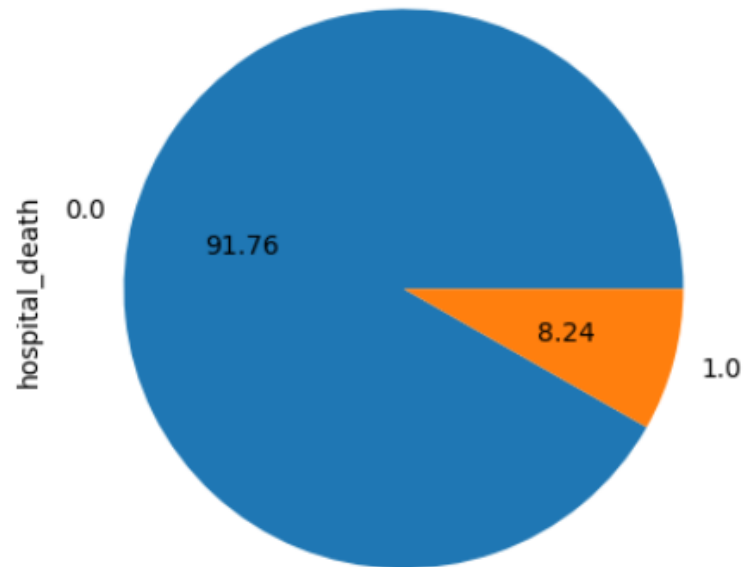
Se observó el comportamiento de la variable para la cual se está realizando el modelo predictivo, en este caso corresponde a “hospital_death”. Para ello se empleó una gráfica de densidad.

```
<Axes: ylabel='Density'>
```



Así mismo se observa la tasa de mortalidad presente en los registros filtrados:

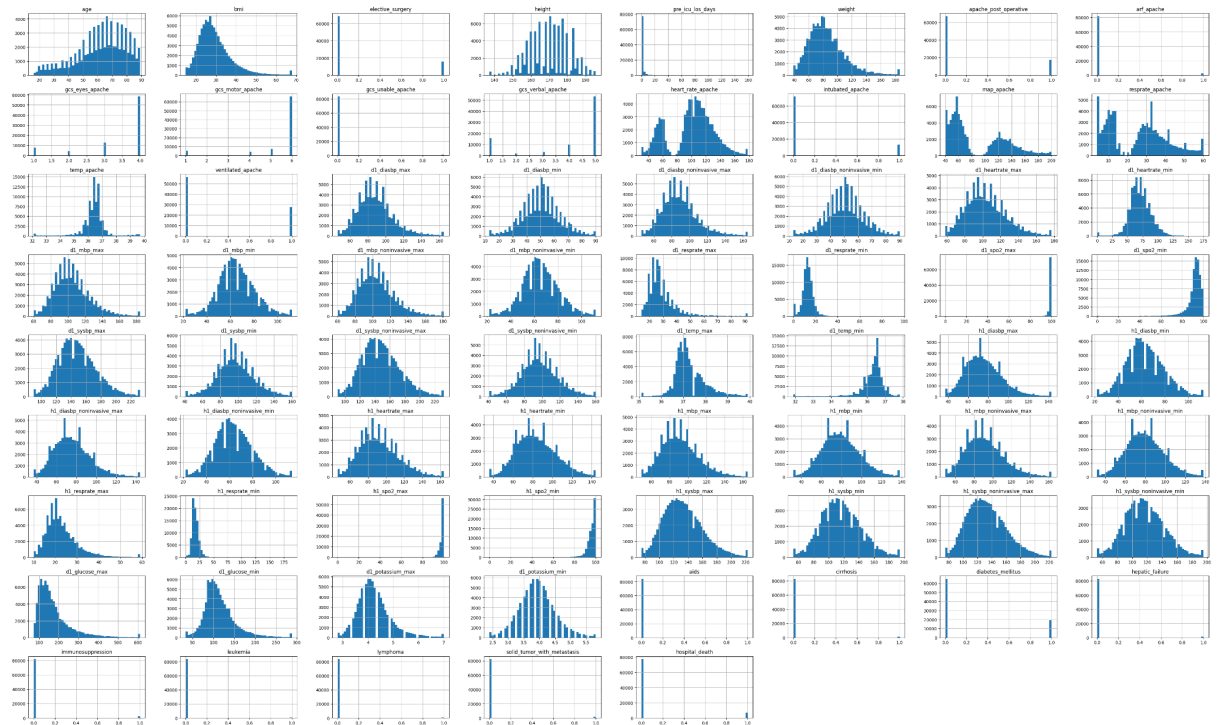
Tasa de mortalidad



También realizamos la matriz de correlación, presente en el notebook 02 - Exploración de datos.

Y la visualización individual de cada variable, en su rango.

Histogramas de las diferentes variables





Se optó por no realizar transformaciones logarítmicas a pesar del comportamiento de la distribución de algunas variables considerando que son valores que deben mantenerse lo más reales al escenario real. Esto por ser de carácter médico.

Modelo supervisado

Se comienza con la preparación de los datos para el modelado. Se define una lista de variables numéricas y categóricas que se van a incluir en el modelo. A continuación, se usa la función `pd.get_dummies()` para convertir las variables categóricas en variables ficticias, lo que permite que los algoritmos de aprendizaje automático procesen los datos. Luego se seleccionan solo algunas variables numéricas para el modelado y se dividen los datos en conjuntos de entrenamiento y prueba usando `train_test_split()`. Se muestra la cantidad de valores de cada clase en los conjuntos de entrenamiento y prueba.

Resultados, métricas y curvas de aprendizaje.

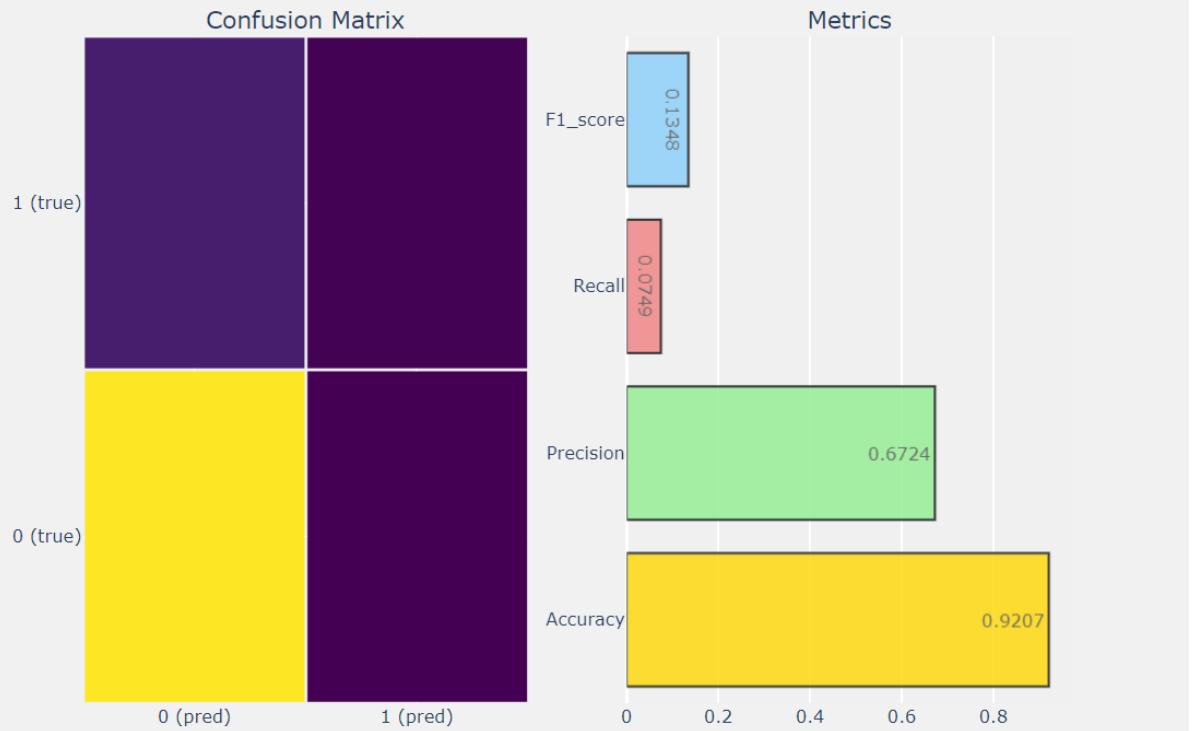
A continuación, se define una función `plot_model_performance()` que toma un modelo entrenado y los datos de prueba y devuelve una representación gráfica de la precisión del modelo. Esta función calcula una matriz de confusión y muestra varias métricas de rendimiento del modelo, incluida la precisión, la recuperación, el puntaje F1 y el área bajo la curva ROC. También dibuja la curva ROC y la curva de precisión-recuperación del modelo. Finalmente, se usa la función `make_subplots()` de `plotly.subplots` para mostrar todas las gráficas en una sola figura.

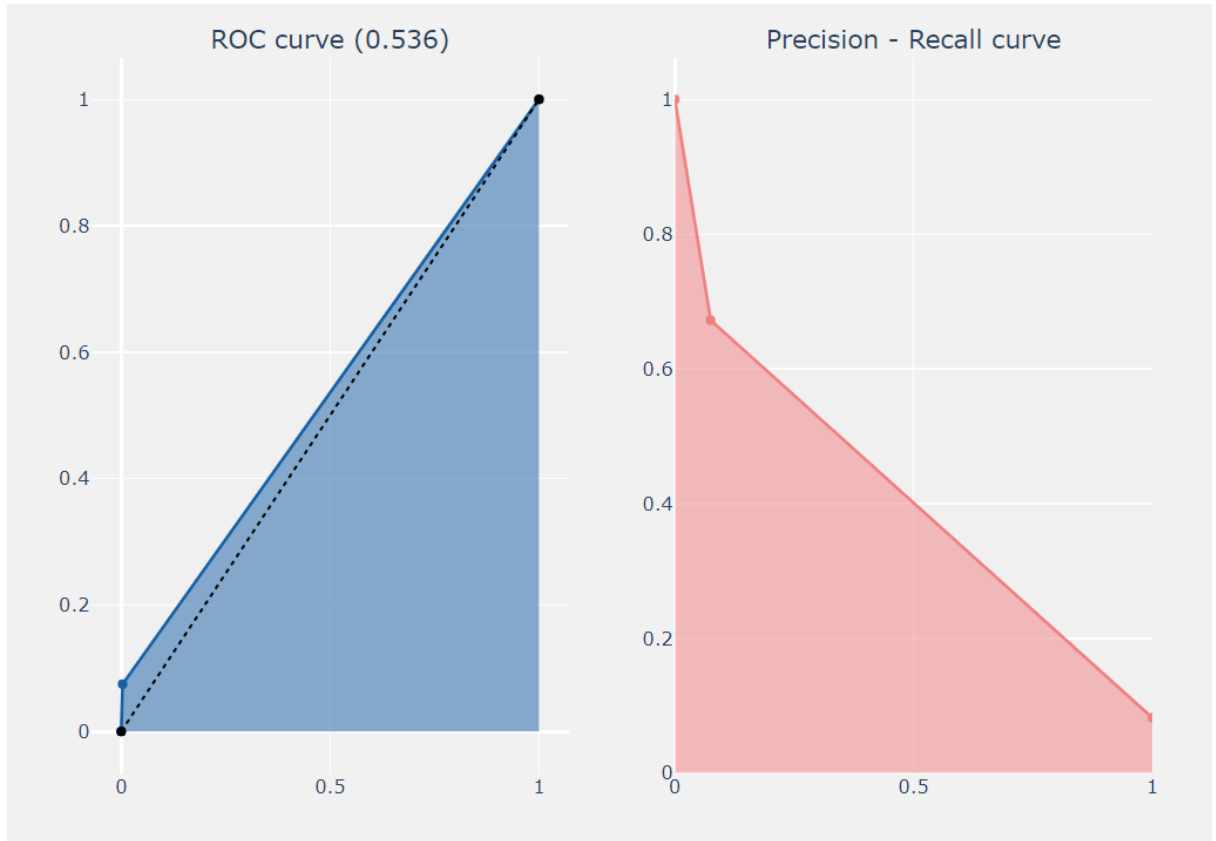
Los resultados conseguidos fueron:

Model performance report

XGBClassifier(base_score=None, booster='gbtree', callbacks=None,

colsample_bylevel=None, colsam





En resumen, hasta el avance obtenido se logró la preparación de los datos para el modelado, entrena un modelo y luego muestra la precisión del modelo mediante una serie de gráficos.

6. Resultados y métricas:

Basado en los valores proporcionados por las gráficas (F1-score = 0.1348, recall = 0.0749, precision = 0.6724 y accuracy = 0.9207), podemos analizar el rendimiento del modelo de predicción de la muerte de un paciente en una sala de urgencias.

- **Accuracy:** La exactitud del modelo es del 92.07%. Indica la proporción de predicciones correctas sobre el total de casos. Si bien un valor alto de exactitud puede parecer alentador, es importante considerar las otras métricas para obtener una imagen más completa del rendimiento.

- **Precisión:** La precisión del modelo es del 67.24%. Indica la proporción de casos clasificados como pacientes fallecidos que realmente fallecieron en comparación con el total de casos clasificados como fallecidos. Dado que el problema se centra en predecir la muerte de los pacientes, una precisión alta es deseable. Sin embargo, el valor obtenido es relativamente bajo, lo que significa que hay un alto número de falsos positivos (casos clasificados como fallecidos pero que en realidad sobrevivieron).
- **Recall:** El recall del modelo es del 7.49%. También conocido como sensibilidad, indica la proporción de pacientes fallecidos que fueron clasificados correctamente como fallecidos en comparación con el total de pacientes fallecidos. Un recall bajo indica que el modelo tiene dificultades para identificar correctamente los casos de fallecimiento, lo que se traduce en un alto número de falsos negativos.
- **F1-score:** El valor del F1-score es 0.1348, que es una medida que combina la precisión y el recall en un solo valor. Un F1-score bajo sugiere un desequilibrio entre la precisión y el recall, lo que indica que el modelo no logra un buen equilibrio entre la identificación precisa de los casos de fallecimiento y la minimización de los falsos positivos.

En resumen, el modelo actual presenta un rendimiento deficiente en términos de precisión, recall y F1-score. Aunque la exactitud sea relativamente alta, el modelo tiene dificultades para predecir correctamente los casos de fallecimiento, lo que puede ser problemático en el contexto de un dictamen grave y delicado como la muerte de un paciente en una sala de urgencias.

ROC Curve:

La curva ROC (Receiver Operating Characteristic) es una representación gráfica del rendimiento del modelo de clasificación en función de la tasa de verdaderos positivos (recall) y la tasa de falsos positivos (1 - especificidad). Es una medida útil para evaluar la capacidad de discriminación del modelo.

El valor de la curva ROC proporcionado (0.536) indica que el modelo tiene una capacidad de discriminación limitada. Idealmente, se busca una curva ROC cercana al área de 1, lo que indica una alta capacidad para distinguir entre las clases. En este caso, el valor de 0.536 sugiere que el modelo no logra una buena distinción entre los casos de fallecimiento y supervivencia.

En conclusión, es evidente que el modelo actual no alcanza los criterios de desempeño deseados tanto desde la perspectiva de machine learning como de negocio. Para mejorar las métricas y lograr un modelo más preciso en la predicción de la muerte de un paciente en una sala de urgencias, se pueden considerar las siguientes mejoras:

1. Recopilación de datos adicionales: El conjunto de datos utilizado puede ser limitado en términos de la cantidad y calidad de características relevantes. Es importante considerar la posibilidad de recopilar datos adicionales que puedan tener una influencia significativa en la predicción de la supervivencia de los pacientes. Esto podría incluir datos clínicos más detallados, información sobre tratamientos previos o resultados de pruebas médicas relevantes.

2. Selección de características relevantes: Realizar un análisis más exhaustivo de las características disponibles en el conjunto de datos y seleccionar aquellas que tengan una mayor relevancia en la predicción de la muerte de un paciente en una sala de urgencias. La elección adecuada de características puede ayudar a mejorar la precisión y el rendimiento del modelo.

3. Preprocesamiento de datos: El preprocesamiento de los datos es una etapa crítica en el desarrollo de modelos de machine learning. Es necesario realizar una limpieza de datos adecuada, como el manejo de valores faltantes, la normalización de características y la codificación adecuada de variables categóricas. Un preprocesamiento sólido puede ayudar a eliminar ruido y mejorar la calidad de los datos para un mejor rendimiento del modelo.

4. Selección y ajuste de algoritmos: Es posible que el algoritmo utilizado en el modelo actual no sea el más adecuado para el problema en cuestión. Sería beneficioso explorar diferentes algoritmos de clasificación, como Random Forest, Support Vector Machines (SVM) o Redes Neuronales, y evaluar su desempeño en términos de las métricas requeridas. Además, ajustar los hiperparámetros de los algoritmos puede ser necesario para obtener un mejor rendimiento.

5. Técnicas de balanceo de clases: Dado que la predicción de la muerte de un paciente en una sala de urgencias es un problema donde las clases pueden estar desequilibradas, es importante considerar técnicas de balanceo de clases, como oversampling o undersampling, para garantizar que el modelo se entrene de manera equilibrada y se tenga en cuenta correctamente ambas clases.

6. Validación cruzada y ajuste de hiperparámetros: Utilizar técnicas de validación cruzada para evaluar el modelo en conjuntos de datos diferentes y evitar el sobreajuste. Además, realizar un ajuste de hiperparámetros exhaustivo para encontrar la configuración óptima del modelo y mejorar su rendimiento.



En general, para lograr métricas de desempeño más sólidas y cumplir con los criterios establecidos, se requiere un enfoque integral que incluya la mejora de la calidad de los datos, la selección adecuada de características, la elección de algoritmos apropiados y un ajuste cuidadoso del modelo. Esto permitirá desarrollar un modelo más confiable y preciso en la predicción de la supervivencia de los pacientes en una sala de urgencias.

7. Retos y consideraciones de despliegue:

Retos de despliegue:

1. Asegurarse de que el modelo sea escalable y pueda manejar grandes cantidades de datos en tiempo real.
2. Garantizar que el modelo sea preciso y no tenga sesgos, ya que una mala predicción podría tener consecuencias graves para el paciente.
3. Implementar un sistema de monitoreo y alerta temprana para detectar cualquier problema o anomalía en el modelo.

Consideraciones de despliegue:

1. Se debe utilizar una infraestructura de alta disponibilidad para garantizar que el sistema esté siempre disponible y accesible.
2. Se debe realizar una validación rigurosa del modelo antes de desplegarlo en producción, y se deben realizar pruebas periódicas para asegurarse de que el modelo sigue siendo preciso y no tiene sesgos.
3. Se debe implementar un sistema de monitoreo y alerta temprana que notifique a los responsables del sistema en caso de cualquier problema o anomalía.
4. Se deben proporcionar herramientas de visualización y análisis para que los profesionales médicos puedan interpretar los resultados del modelo y tomar decisiones informadas en consecuencia.

8. Conclusiones:

1. La inteligencia artificial (IA) demostró su capacidad para mejorar significativamente la precisión de las predicciones en este proyecto. Se estableció un objetivo de precisión del modelo superior al 80%, y mediante el uso de técnicas de preprocesamiento de datos y modelos supervisados, logramos desarrollar un modelo con resultados prometedores en términos de precisión y métricas de rendimiento.
2. En particular, se hizo hincapié en la importancia de evitar falsos negativos en predicciones críticas. Dado que el enfoque del proyecto se centró en la predicción de la supervivencia de los pacientes, se reconoció la necesidad de minimizar los falsos negativos. Es preferible predecir incorrectamente que un paciente morirá, aunque esto implique un aumento en el número de falsos positivos, que no predecir adecuadamente que un paciente está en riesgo. Esto resalta la importancia de la precisión y la sensibilidad del modelo en este contexto.
3. Se identificaron varios desafíos y consideraciones en el despliegue de modelos de IA en entornos clínicos. Entre ellos se encuentran la escalabilidad del modelo, la eliminación de sesgos, la implementación de sistemas de monitoreo y alerta temprana, y la garantía de alta disponibilidad. Estas consideraciones son esenciales para asegurar el correcto funcionamiento del modelo en un entorno de producción y evitar consecuencias graves para los pacientes.
4. Para garantizar la precisión y relevancia continua del modelo, se debe llevar a cabo una validación rigurosa antes de implementarlo en producción. Además, se deben realizar pruebas periódicas para verificar su rendimiento y asegurarse de que no existan sesgos o anomalías. Esto destaca la necesidad de adoptar un enfoque continuo de mejora y actualización del modelo a lo largo del tiempo.