“Modelos predictivos de mortalidad: comparación de técnicas tradicionales y algoritmos de machine learning”

ACTIVIDAD FORMATIVA EQUIVALENTE A TESIS PARA OPTAR AL GRADO DE MAGÍSTER EN ANALÍTICA DE NEGOCIOS

Alumno: Christian Yañez Cornejo

Profesor Guía: David Diaz Solís

Santiago, mayo 2025

Tabla de contenido

Resumen ejecutivo 4
Introducción 7
Planteamiento y Formulación del Problema 10
Hipótesis 12
Objetivos 13
Metodología Propuesta 15
0. Flujograma del Proceso Metodológico 15
1. Recopilación y preprocesamiento de datos para algoritmos de aprendizaje automático: 16
2. Análisis descriptivo de los datos: 19
3. Preprocesamiento de los datos con pycaret 31
4. Construcción y preprocesamiento de la serie temporal de fallecimientos 36
5. Configuración de modelos y selección de hiperparámetros para algoritmos de aprendizaje automático 38
6. Interpretación de los modelo con SHAP values 41
7. Evaluación de las tablas de mortalidad vigentes: 43
8. Desarrollo y entrenamiento de algoritmos de aprendizaje automático y series temporales: 44
9. Comparación mensual entre algoritmos de aprendizaje automático, modelos de series temporales y las tablas de mortalidad vigentes: 44
10. Recomendaciones Basadas en los A/E: 45
Marco Teórico 46
1. Mortalidad y su importancia para el negocio de Rentas Vitalicias 46
2. Tablas de mortalidad: Conceptos y Metodologías de Construcción 46
2.1. Fuentes de Datos para Tablas de Mortalidad 47
2.2. Cálculo de las Tasas de Mortalidad 47
2.3. Métodos de Graduación y Ajuste de las Tasas 47
2.4. Mejoras en las Tablas de Mortalidad 48
3. Limitaciones de las Tablas de Mortalidad Convencionales 48
4. Algoritmos de aprendizaje automático en la predicción de mortalidad 49
4.1. Formulación matemática de modelos de machine learning 49
4.2. Modelos de Boosting para la predicción de mortalidad 49
4.3 Redes neuronales (NN) y su aplicación en la predicción de mortalidad 53
5.- Modelos de series temporales (SARIMA, GRU y LSTM) y su comparación con enfoques de aprendizaje automático 63
5.1 Diferencia al usar LSTM y GRU como algoritmos de aprendizaje automático frente a su uso en series temporales 64
6. Métricas de evaluación según el enfoque 65
6.1 Enfoque de clasificación individual (aprendizaje automático): 65
6.2 Enfoque de Series Temporales (SARIMA, GRU, LSTM como modeladores de secuencias): 66
6.3 Tablas de Mortalidad (Método Convencional): 67

7. Justificación del uso de algoritmos de aprendizaje automático y modelos de series temporales para la proyección de la mortalidad 68
Descripción del enfoque de solución 69
Paso 1: Recopilación y preprocesamiento de datos 69
Paso 2: Selección y entrenamiento de modelos 70
Paso 3: Evaluación y ajuste mensual de los modelos seleccionados 71
Paso 4: Comparación y selección del modelo final 72
Paso 5: Implementación y monitoreo continuo 73
Resultados: 74
1. Proceso de Selección y Evaluación de Modelos 74
2. Evaluación del Desempeño del Modelo 75
3. Evaluación de Modelos Supervisados Basados en Redes Neuronales 78
4. Comparación del rendimiento en entrenamiento y prueba 80
5. Introducción a la Interpretabilidad con SHAP Values 82
Shap Values para XGBooster 84
Shap values para RN MLP (Red Neuronal Densa) 86
Shap Values para la red neuronal GRU 88
Shap values para la red neuronal LTSM 89
6. Análisis de Modelos de Series Temporales 91
Resumen del modelo SARIMA y sus resultados 92
Resumen del modelo de series temporales GRU y sus resultados 96
Resumen del modelo LSTM y sus resultados 99
Resumen de Series Temporales y Hallazgos 101
Conclusiones del análisis comparativo entre modelos 102
Análisis con datos reales (enero - junio 2024) 103
Análisis Comparativo de los Modelos de Predicción de Mortalidad 107
Conclusiones 111
Bibliografía 114
Apéndice 115

Resumen ejecutivo

El presente estudio se centra en el análisis y predicción de la mortalidad dentro de la cartera de rentas vitalicias de una importante compañía multinacional de seguros que opera en Chile, utilizando tanto modelos basados en algoritmos supervisados como series temporales. Para ello, se realizó un análisis descriptivo detallado de la población asegurada, segmentándola por edad, sexo, tipo de asegurado, nivel de prima y distribución geográfica. Se identificó que la mayoría de los asegurados se concentran en edades entre los 60 y 75 años, con una mayor proporción de mujeres. Asimismo, se evidenció que los asegurados con primas bajas representan el segmento más amplio, y que la Región Metropolitana concentra la mayor cantidad de asegurados.

Un aspecto clave del análisis fue la evaluación de los modelos mediante la métrica **A/E (Actual vs Expected)**, que compara los fallecimientos observados (reales) vs los fallecimientos esperados (según los modelos basados en algoritmos de aprendizaje automático, series temporales y/o tablas de mortalidad). Esta métrica es fundamental en la industria del seguro, ya que permite medir la precisión de las proyecciones de mortalidad.

Se implementaron y evaluaron dos enfoques de modelamiento para la predicción de mortalidad: un modelo basado en algoritmos de aprendizaje supervisado **(XGBoost)** y un modelo de series temporales, la red **LSTM**. La selección de estos métodos respondió a su capacidad para capturar distintos aspectos de la mortalidad: mientras que **XGBoost** aprovecha información estructurada y diversas variables explicativas, **LSTM** se enfoca en la evolución temporal de la serie de fallecimientos. Ambos modelos fueron ajustados y validados utilizando datos históricos, mostrando niveles de precisión competitivos frente a las tablas de mortalidad tradicionales.

Al evaluar ambos modelos con datos reales de enero a junio de 2024, **XGBoost** presentó los resultados más cercanos a la métrica A/E esperada, con valores más estables en la mayoría de los segmentos, particularmente para los beneficiarios y causantes. En contraste, la red **LSTM** mostró una mayor capacidad de capturar tendencias temporales, pero con mayores desviaciones en algunos meses, especialmente en el segmento de inválidos.

Totales							
Mes	**Real (Muertes)**	**Predicción (Tabla de Mortalidad)**	**Predicción (XGBoost)**	**Predicción (LSTM)**	**A/E (Tabla de Mortalidad)**	**A/E (XGBoost)**	**A/E (LSTM)**
ene-24	288	299	275	327	96%	105%	88%
feb-24	335	297	352	320	113%	95%	105%
mar-24	329	301	328	328	109%	100%	100%
abr-24	335	301	326	335	111%	103%	100%
may-24	443	301	382	346	147%	116%	128%
jun-24	387	304	391	331	127%	99%	117%
YTD	**2,117**	**1,803**	**2,054**	**1,987**	**117%**	**103%**	**107%**

Tabla 1: Comparación de A/E por tipo de modelo – Resultados Totales

Causantes							
Mes	**Real (Muertes)**	**Predicción (Tabla de Mortalidad)**	**Predicción (XGBoost)**	**Predicción (LSTM)**	**A/E (Tabla de Mortalidad)**	**A/E (XGBoost)**	**A/E (LSTM)**
ene-24	175	163	161	210	107%	109%	83%
feb-24	187	161	198	214	116%	94%	87%
mar-24	188	163	194	222	115%	97%	85%
abr-24	199	164	178	221	121%	112%	90%
may-24	257	165	191	222	156%	135%	116%
jun-24	221	166	187	207	133%	118%	107%
YTD	**1,227**	**982**	**1,109**	**1,296**	**125%**	**111%**	**95%**

Tabla 2: Comparación de A/E por tipo de modelo – Causantes

Beneficiarios							
Mes	**Real (Muertes)**	**Predicción (Tabla de Mortalidad)**	**Predicción (XGBoost)**	**Predicción (LSTM)**	**A/E (Tabla de Mortalidad)**	**A/E (XGBoost)**	**A/E (LSTM)**
ene-24	83	114	81	97	73%	102%	86%
feb-24	125	114	140	88	110%	89%	142%
mar-24	114	116	108	86	98%	106%	133%
abr-24	108	115	143	94	94%	76%	115%
may-24	159	114	159	102	139%	100%	156%
jun-24	122	116	170	103	105%	72%	118%
YTD	**711**	**689**	**801**	**570**	**103%**	**89%**	**125%**

Tabla 3: Comparación de A/E por tipo de modelo – Beneficiarios

Además, para evaluar el desempeño de los modelos basados en algoritmos de aprendizaje automático (modelos binarios), se reportó el área bajo la curva **(AUC)** de la curva ROC. De los modelos basados en algoritmos supervisados, el **XGBoost**, logro un mejor desempeño en la métrica **AUC**, con un valor de 0.92, lo que indica una capacidad predictiva robusta en comparación con los otros modelos binarios evaluados. Estos valores refuerzan la utilidad del modelo XGBoost en la estimación de mortalidad.

Model	Dataset	Accuracy	AUC ROC	Re-call (Clase Muerto)	Precisión(Clase Muerto)	F1-Score (Clase Muerto)
XGBoost	Test	0.8300	0.9200	0.7820	0.5224	0.6264
DNN	Test	0.8100	0.8800	0.7591	0.4983	0.6032
LSTM	Test	0.8000	0.8900	0.7752	0.4756	0.5887
GRU	Test	0.8100	0.9000	0.7683	0.5127	0.6154

Tabla 4: Comparativo de métricas entre modelos basados en algoritmos supervisados

Por otro lado, un punto clave del análisis fue la comparación entre los modelos predictivos y las tablas de mortalidad. Si bien los modelos de machine learning ofrecen mejoras en la estimación de la mortalidad individual, las tablas de mortalidad continúan siendo una referencia fundamental en la industria del seguro. Se concluye que, bajo ningún escenario, los modelos basados en datos históricos reemplazarán las tablas tradicionales, sino que servirán como herramientas complementarias para mejorar la gestión del riesgo y la toma de decisiones en la tarificación y reservas actuariales.

Además, la correlación entre variables mostró que factores como la edad y la condición de invalidez tienen una relación significativa con la mortalidad, lo que respalda su inclusión en futuros modelos. La tasa de mortalidad (qx) sigue una tendencia creciente con la edad, lo que valida la aplicación de enfoques no lineales en la predicción. Este hallazgo es crucial para mejorar la precisión de los modelos y ajustar de manera más eficiente las políticas de gestión del portafolio.

En conclusión, este estudio/tesis demuestra el valor de la inteligencia artificial y los modelos estadísticos avanzados en la predicción de mortalidad. No obstante, para garantizar robustez en la toma de decisiones, se recomienda seguir utilizando las tablas de mortalidad como base, complementándolas con técnicas de machine learning para obtener estimaciones más precisas y dinámicas. A futuro, se sugiere explorar modelos híbridos que combinen ambos enfoques, así como la incorporación de variables macroeconómicas y biométricas que puedan mejorar la capacidad predictiva.

Introducción

La predicción de la mortalidad es un componente esencial en la gestión de riesgos para compañías de seguros de vida, especialmente en el negocio de rentas vitalicias, donde los compromisos financieros son de largo plazo. En este tipo de productos, el asegurador asume la responsabilidad de proveer un flujo de ingresos estable para los asegurados durante el resto de sus vidas. Por lo tanto, contar con estimaciones precisas sobre la longevidad de los asegurados permite a las aseguradoras garantizar la suficiencia de sus reservas técnicas, mantener la estabilidad financiera y optimizar la oferta de pensiones, lo cual es clave en un mercado altamente competitivo. Las rentas vitalicias son percibidas como un "**commodity**", y la diferenciación entre compañías se basa, en gran medida, en los precios ofertados. Por ello, disponer de una proyección de mortalidad precisa permite establecer precios justos y sostenibles que aseguren tanto la rentabilidad de la compañía como la satisfacción de los clientes.

La metodología convencional para predecir la mortalidad se basa en el uso de tablas de mortalidad estáticas, que reflejan la experiencia histórica y demográfica de poblaciones amplias. Sin embargo, este enfoque tiene limitaciones importantes, especialmente en contextos de cambios demográficos abruptos y eventos inesperados como la pandemia de COVID-19, los cuales alteraron significativamente las tasas de mortalidad en distintos segmentos etarios. En Chile, por ejemplo, las tasas de mortalidad en adultos mayores han mostrado una disminución sostenida en los últimos años, aunque se ha observado un leve repunte en 2024 debido a la estacionalidad de los meses de invierno. Estos cambios demográficos resaltan la necesidad de contar con modelos más dinámicos que puedan adaptarse a condiciones de salud fluctuantes y específicas de la población.

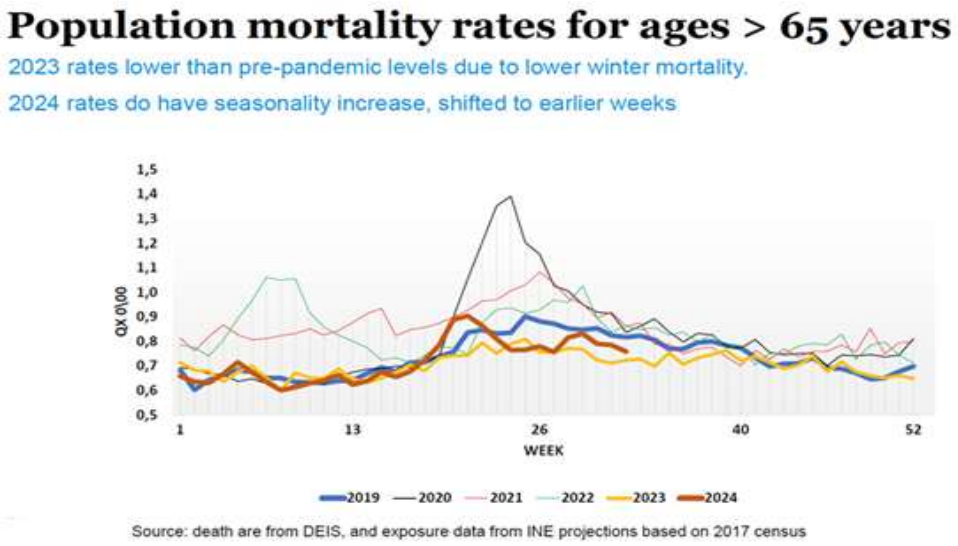


Figura 1: mortalidad poblacional

La presente tesis se enfoca en la cartera de rentas vitalicias de una importante compañía multinacional de seguros que opera en Chile, que abarca a más de 270,000 asegurados, distribuidos en diversos segmentos según nivel de prima, tipo de pensión y condición de salud. Esta cartera incluye grupos variados, donde los causantes sanos masculinos y los beneficiarios sanos femeninos representan una proporción importante, mientras que los inválidos parciales y totales constituyen un porcentaje menor. La heterogeneidad de estos segmentos añade complejidad al proceso de predicción de mortalidad, ya que cada grupo presenta características demográficas y de salud que influyen de manera diferente en la longevidad. Esto hace necesario ajustar los modelos para captar las particularidades de cada subgrupo y ofrecer proyecciones de mortalidad más exactas y personalizadas.

HOMBRES	TOTAL	MUJERES	TOTAL	TOTAL	TOTAL
CAUSANTES SANOS	74,348	CAUSANTES SANOS	33,992	CAUSANTES SANOS	108,340
BENEFICIARIOS SANOS	42,478	BENEFICIARIOS SANOS	108,204	BENEFICIARIOS SANOS	150,682
INVÁLIDOS TOTALES	7,612	INVÁLIDOS TOTALES	2,950	INVÁLIDOS TOTALES	10,562
INVÁLIDOS PARCIALES	467	INVÁLIDOS PARCIALES	387	INVÁLIDOS PARCIALES	854
TOTAL	124,905	TOTAL	145,533	TOTAL	270,438

Tabla 5: Elaboración Propia

Con el objetivo de mejorar la precisión de las predicciones de mortalidad, esta tesis explora el uso de algoritmos de aprendizaje supervisado y modelos de series temporales, dos enfoques que han demostrado ser herramientas robustas para identificar patrones complejos y adaptarse a nuevas realidades. A diferencia de las tablas de mortalidad tradicionales, los algoritmos de aprendizaje supervisado pueden incorporar un mayor número de variables y ajustarse en tiempo real, lo que les permite capturar tendencias y particularidades específicas de cada subgrupo de asegurados.

En esta tesis, se dividió el trabajo en dos grandes líneas metodológicas:

1. **Algoritmos de aprendizaje supervisado**, entre los que se incluyeron XGBoost, Redes Neuronales Densas, GRU y LSTM.
2. **Modelos de series temporales**, específicamente SARIMA, Redes GRU y LSTM aplicadas como modelos secuenciales.

Los modelos se entrenaron con datos históricos disponibles hasta diciembre de 2023 y posteriormente se validaron utilizando información real de enero a junio de 2024. Esta estrategia de evaluación fuera de la muestra permitió analizar la capacidad de generalización de cada enfoque y su desempeño ante escenarios futuros.

Para medir la efectividad de las predicciones, se emplearon métricas como precisión, recall, exactitud y MCC (Matthews Correlation Coefficient). Esto permitió identificar las metodologías con mayor potencial para anticipar la mortalidad en la cartera de rentas vitalicias.

En esta tesis se propuso comparar la eficacia de las tablas de mortalidad vigentes con modelos predictivos basados en algoritmos de aprendizaje automático y series temporales, aplicados a la cartera de rentas vitalicias. Si bien los resultados muestran similitudes con las estimaciones obtenidas mediante las tablas de mortalidad tradicionales, la adopción de estas metodologías avanzadas aporta la capacidad de capturar patrones de estacionalidad y mejorar la precisión de las proyecciones financieras. De este modo, se logra un entendimiento más detallado de la evolución de la mortalidad a lo largo del tiempo, lo cual permite a la compañía optimizar sus escenarios financieros y tomar decisiones con mayor respaldo analítico, reforzando así su competitividad y la satisfacción de sus clientes.

Planteamiento y Formulación del Problema

La estimación precisa de la mortalidad es crítica para la sostenibilidad financiera de las compañías de seguros de vida que operan en el segmento de rentas vitalicias, donde los compromisos asumidos son a largo plazo y las reservas deben ser suficientes para cubrir las pensiones de por vida de los asegurados.

En la compañía multinacional de seguros que nos proporcionó los datos del estudio, las proyecciones de mortalidad se basan en tablas propias, desarrolladas por la propia compañía y utilizando la metodología estándar para su construcción (***ver anexo***). Si bien estas tablas de mortalidad son estáticas en su estructura básica, funcionan en conjunto con factores de mejoramiento, los cuales les aportan el grado de dinamismo a las mismas y permiten que las tasas de mortalidad se ajusten en función a los avances en tecnológicos, médicos y de políticas públicas. Es importante señalar que la evidencia empírica muestra que la mortalidad en las poblaciones humanas tiende a decrecer a través del tiempo debido a los factores anteriormente mencionados.

El mejoramiento plantea que las tasas futuras de mortalidad son un factor de una tasa base. El modelo es descrito por la ecuación:

$$q_{x,t} = q_{x,t_0}\,(1 - AA_x)^{t-t_0}$$

Donde t_0 se llama año base, $q_{x,t}$ es la tasa de mortalidad, q_{x,t_0} es la tasa de mortalidad en el año base (dato conocido) y AA_x se llama Factor de Mejoramiento, el cual, aislando AA_x desde la ecuación anterior, es calculado como:

$$AA_x = 1 - \left(\frac{q_{x,t_p}}{q_{x,t_0}}\right)^{\frac{1}{t_p - t_0}}$$

Donde $t_p > t_0$ es un año en el futuro (habitualmente 60 años al futuro) y q_{x,t_p} es una tasa de mortalidad proyectada para el año t_p. Es importante notar que se calcula un factor distinto para cada edad o tramo etario x.

Sin embargo, estas tablas, aún con los factores de mejoramiento, presentan limitaciones significativas, especialmente frente a cambios abruptos en las condiciones de salud poblacional, como se evidenció durante la pandemia de COVID-19. A medida que las tasas de mortalidad en Chile retornan a niveles previos a la pandemia, se observan aumentos estacionales en invierno que revelan la inadecuada sensibilidad de las tablas

convencionales a variaciones rápidas, tanto estacionales como anuales, y a eventos pandémicos.

Tal como se visualizó en la ***figura 1***, durante el 2023 y 2024, la mortalidad ha vuelto a niveles prepandemia, pero también ha experimentado aumentos puntuales debido a la estacionalidad de invierno. Esto sugiere, tal como se mencionó anteriormente, que los métodos tradicionales no son suficientemente sensibles a las dinámicas estacionales o a las variaciones anuales derivadas de eventos pandémicos. Como resultado, existe una creciente necesidad de explorar nuevas metodologías que permitan ajustarse a cambios más frecuentes y precisos en las condiciones de mortalidad.

Este contexto plantea la necesidad de explorar enfoques alternativos que se adapten de forma más precisa y dinámica a las fluctuaciones en las tasas de mortalidad. El problema central de esta tesis es determinar qué método ofrece mejores proyecciones de mortalidad futura para la cartera de rentas vitalicias: las tablas de mortalidad ajustadas tradicionalmente, los modelos predictivos basados en algoritmos de aprendizaje automático o las series temporales. Las tablas de mortalidad convencionales basadas en datos históricos proporcionan proyecciones a largo plazo, pero carecen de la capacidad para adaptarse de inmediato a cambios demográficos o socioeconómicos significativos. En contraste, los algoritmos de aprendizaje automático permiten incorporar un conjunto amplio de variables, ajustarse a las variaciones de los datos en tiempo real y reflejar nuevas tendencias en longevidad de manera ágil. Adicionalmente, los modelos de series temporales contribuyen a captar la estacionalidad y los patrones cíclicos de la mortalidad, brindando una visión más detallada de la evolución a corto y mediano plazo que complementa la perspectiva tradicional y estática de las tablas de mortalidad.

Por lo tanto, el objetivo de esta tesis es desarrollar una metodología que permita a nuestra compañía multinacional de seguros obtener proyecciones de mortalidad más precisas para su cartera de rentas vitalicias, lo que no solo contribuirá a la exactitud de las proyecciones financieras de la compañía, sino que también facilitará una adaptación más ágil ante cambios demográficos y económicos. En este contexto, se evalúan dos grandes enfoques: las tablas de mortalidad ajustadas y los modelos basados en algoritmos de aprendizaje automático junto con series temporales. El desafío consiste en determinar cuál de estos métodos ofrece las predicciones más precisas y confiables para la toma de decisiones financieras a largo plazo, atendiendo tanto a su desempeño con datos reales como a la simplicidad de su implementación en la práctica.

Hipótesis

Hipótesis principal:

Los modelos predictivos basados en algoritmos de aprendizaje automático y/o los modelos de series temporales proporcionan predicciones de mortalidad más precisas que las tablas de mortalidad convencionales en la cartera de rentas vitalicias, debido a su capacidad para capturar patrones complejos, incluidas estacionalidades y tendencias, y adaptarse con mayor rapidez a los cambios demográficos de la población.

Hipótesis secundarias:

- ✓ Los modelos de *aprendizaje automático* y/o las aproximaciones basadas en *series temporales* se proyectan como más efectivos en grupos específicos, particularmente entre causantes y beneficiarios, que constituyen la mayor parte de la cartera de rentas vitalicias de nuestra compañía multinacional de seguros.

- ✓ Las tablas de mortalidad convencionales, al ser basadas en datos históricos, tenderán a sobreestimar o subestimar la mortalidad en períodos de alta variabilidad, como lo observado durante y después de la pandemia de COVID-19.

Estas hipótesis guiarán el análisis comparativo de las metodologías, con el objetivo de validar si los algoritmos de aprendizaje automático y/o las series temporales pueden ofrecer una ventaja significativa en la gestión de riesgos y proyecciones financieras para la compañía.

Objetivos

Objetivo General: Comparar la efectividad de las tablas de mortalidad convencionales (vigentes y desarrolladas por la compañía) con los algoritmos de aprendizaje automático y enfoques de *series temporales* para predecir la mortalidad en la cartera de rentas vitalicias, con el fin de mejorar la precisión de las proyecciones y, por ende, obtener proyecciones financieras más exactas.

Objetivos Específicos:

- ✓ Evaluar periódicamente el desempeño de las tablas de mortalidad mediante el cálculo de la métrica A/E (Actual vs. Expected), con el propósito de determinar su precisión y confiabilidad en la predicción de la mortalidad.

Contexto A/E (Actual vs. Expected):

a) El A/E compara los fallecimientos observados (Actuals) con los fallecimientos esperados (Expected) según las tablas de mortalidad.
b) Suele utilizarse un promedio móvil (rolling) de 12 meses para suavizar fluctuaciones debidas a factores estacionales o eventos puntuales.
c) En esta tesis, **se emplearán estimaciones mensuales puntuales** de la mortalidad, ya que el o los modelos seleccionados una vez que se encuentren en producción serán reentrenados cada mes con datos actualizados, permitiendo capturar cambios inmediatos en las tendencias de mortalidad y facilitando la adaptación continua a nuevas realidades demográficas.

- ✓ **Desarrollar y entrenar algoritmos de aprendizaje automático, así como enfoques de *series temporales* que incorporen variables adicionales,** por ejemplo, de índole socioeconómica para la predicción de la mortalidad de los asegurados, y comparar sus resultados con los A/E (Actual vs. Expected) calculados de manera periódica a partir de las tablas de mortalidad vigentes utilizadas por la compañía.

- ✓ **Comparar los resultados de ambos enfoques** en términos de precisión, sensibilidad y capacidad de adaptación a cambios demográficos y eventos imprevistos, utilizando los A/E como referencia.

- ✓ **Analizar la aplicabilidad de los algoritmos de aprendizaje automático y enfoques de *series temporales*** en diferentes segmentos de la cartera (p. ej., causantes, beneficiarios e inválidos) para identificar patrones más detallados de mortalidad y comparar su comportamiento de manera específica.

- ✓ **Proponer recomendaciones basadas en el análisis de A/E y los modelos predictivos**, sugiriendo el enfoque que permita mejorar la predicción de mortalidad y la gestión de riesgo de la compañía.

Metodología Propuesta

Para el desarrollo de este estudio, se ha seguido una metodología estructurada que abarca desde la recopilación de datos hasta la evaluación del desempeño de los modelos.

0. Flujograma del Proceso Metodológico

La siguiente figura ilustra la secuencia de pasos seguida en la investigación, proporcionando una visión clara de la estructura del análisis:

A continuación, se detallan cada uno de los pasos mencionados en el diagrama.

1. Recopilación y preprocesamiento de datos para algoritmos de aprendizaje automático:

Para este estudio, se utilizará la base de datos completa de la cartera de rentas vitalicias de nuestra compañía multinacional de seguros a diciembre 2023, que incluye información detallada sobre los asegurados, como edad, género, nivel de prima, tipo de pensión y condición de salud, entre otros factores relevantes. Además, se incorporarán los resultados de los cálculos de A/E (Actual vs Expected), los cuales proporcionan una comparación entre las tasas de mortalidad observadas y las esperadas según las tablas de mortalidad actuales. Esta métrica es crucial, ya que permitirá evaluar la precisión y confiabilidad de las tablas vigentes y servirá como una referencia comparativa para los modelos predictivos basados en algoritmos de aprendizaje automático y series temporales que se desarrollarán en este trabajo.

Como parte del preprocesamiento de datos, se utilizó la técnica de codificación *one-hot* para transformar las variables categóricas. Este proceso convierte cada categoría de una variable en una columna binaria independiente que toma el valor de 1 cuando se cumple la condición y 0 en caso contrario. La codificación one-hot es especialmente útil en el contexto de algoritmos, ya que permite que los modelos interpreten correctamente las relaciones entre categorías sin imponer un orden implícito entre ellas. Esto es fundamental en este caso, ya que variables como género, región, tipo de prima, y estado de salud no tienen un orden secuencial, pero sí contienen información importante para la predicción de mortalidad.

Por ejemplo:

- El sexo se codificó en variables binarias BEN_SEXO_F y BEN_SEXO_M, para indicar el género femenino o masculino respectivamente.
- El tipo de asegurado se descompuso en columnas binarias como causante, beneficiario e invalido, según el tipo de estado o rol que el individuo tiene dentro del seguro de renta vitalicia.
- La región de residencia se codificó en columnas separadas, como Region_1, Region_2, etc., que reflejan la pertenencia del asegurado a una región específica del país.

La implementación de la codificación one-hot asegura que los algoritmos de aprendizaje automático reciban la información en un formato que optimiza la precisión predictiva, al permitirles aprender patrones únicos asociados a cada categoría. Este proceso, junto con los procesos de limpieza, escalamiento, balanceo y validación cruzada de datos, constituye una parte fundamental del preprocesamiento de los datos, maximizando la utilidad de cada variable en el modelo sin introducir sesgos o interpretaciones incorrectas.

1.1 Análisis de las variables

En esta sección, se presenta una descripción detallada de las variables seleccionadas para los modelos. La cartera de datos contiene tanto variables numéricas como categóricas, cada una con su respectivo tratamiento y relevancia en la predicción de la mortalidad:

Variables numéricas: Incluyen la edad en años y el número de hijos:

a) **Edad_Inicio**: Representa la edad en años en el momento de inicio del estudio y en el caso que el individuo este muerto, representa la edad en años al momento de la muerte. Es una variable continua que ayuda a capturar la influencia de la edad en la mortalidad.
b) **N_Hijos**: Indica el número de hijos que tiene el asegurado. Esta variable es continua y puede influir en la mortalidad según patrones de carga familiar o soporte social.

Estas variables se procesan mediante normalización o estandarización para mejorar la consistencia en el modelado y asegurar que no dominen el proceso de aprendizaje por su magnitud.

Variables categóricas: Las siguientes variables categóricas se codificaron mediante técnicas de one-hot encoding, permitiendo que los algoritmos de aprendizaje automático las utilice como variables indicadoras:

a) **BEN_SEXO_F** y **BEN_SEXO_M**: Variables binarias que indican el sexo del asegurado. BEN_SEXO_F es igual a 1 si es mujer y 0 en caso contrario; BEN_SEXO_M es igual a 1 si es hombre.
b) **Prima_alta, Prima_media, Prima_baja**: Variables categóricas que representan el nivel socioeconómico del asegurado según el nivel de prima pagado. Ayudan a capturar diferencias en mortalidad asociadas al poder adquisitivo.
c) **Conyuge_no** y **Conyuge_si**: Variables binarias que indican el estado conyugal del asegurado. Conyuge_si es igual a 1 si tiene cónyuge y 0 en caso contrario.
d) **Causante** y **Beneficiario**: Identifican el rol del asegurado en la póliza. Causante es 1 si es el rentista, mientras que beneficiario indica si es el beneficiario de una renta vitalicia.
e) **Invalidez**: Esta variable indica si el asegurado presenta alguna condición de invalidez. Invalidez es igual a 1 si está inválido y 0 en caso contrario.
f) **Region_1 a Region_15**: Conjunto de variables binarias que indican la región de residencia del asegurado en Chile, desde la Región 1 hasta la Región 15. Permiten capturar variaciones geográficas en las tasas de mortalidad.

La variable dependiente (Y) es mortalidad, que se codifica como binaria (1 si la persona fallece, 0 si permanece viva). Se verifico que ninguna variable predicadora (X) esté vinculada directamente con la variable objetivo, evitando así sesgos o redundancias que puedan comprometer la robustez de los modelos.

2. Análisis descriptivo de los datos:

El análisis descriptivo de los datos es un paso fundamental para comprender la distribución y características de la población asegurada antes de aplicar modelos predictivos. En esta sección, se exploran las variables clave utilizadas en el estudio, identificando patrones, tendencias y posibles anomalías que puedan influir en los resultados.

Distribución de la edad de Inicio de los asegurados

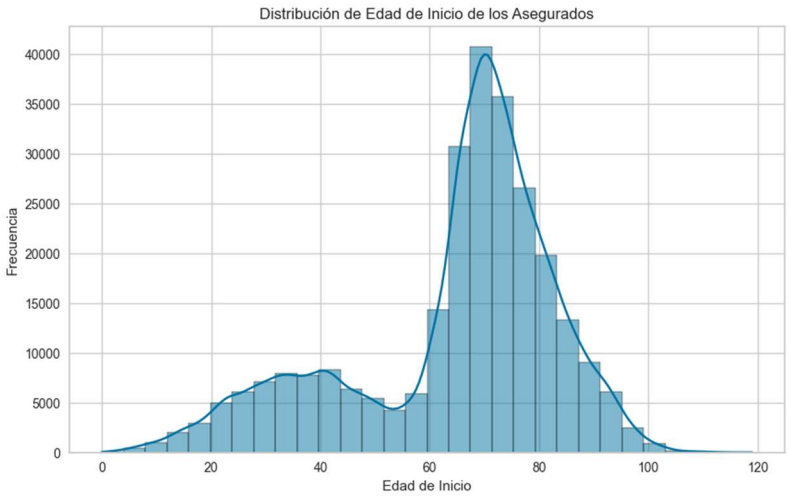


Figura 2: Distribución de edades de los asegurados

El siguiente histograma muestra la distribución de la edad de los asegurados en la base de datos analizada. La variable Edad de Inicio representa la edad al momento del inicio del estudio o la edad de muerte del individuo, lo que es un factor determinante en la evaluación de la mortalidad y la estimación de rentas vitalicias.

Como se observa en la gráfica, la distribución de edades presenta un comportamiento bimodal. Se pueden identificar dos grupos principales de asegurados:

- Un primer grupo con edades menores a 50 años, que están asociados a pensiones de invalidez y beneficiarios femeninos.
- Un segundo grupo, con un claro pico entre los 60 y 70 años que están asociados a los causantes tanto masculinos como femeninos.

Adicionalmente, la cola derecha de la distribución indica la presencia de asegurados con edades avanzadas al momento del inicio del estudio, aunque en menor frecuencia. Este comportamiento es relevante, ya que la edad de inicio tiene una relación directa con la longevidad esperada y, por lo tanto, con la proyección de mortalidad.

Distribución de Asegurados por Sexo

El siguiente gráfico de barras presenta la distribución de asegurados según su sexo, diferenciando entre **Masculino** y **Femenino**. Se observa que la población asegurada se encuentra relativamente equilibrada, aunque con una ligera predominancia de mujeres sobre los hombres.

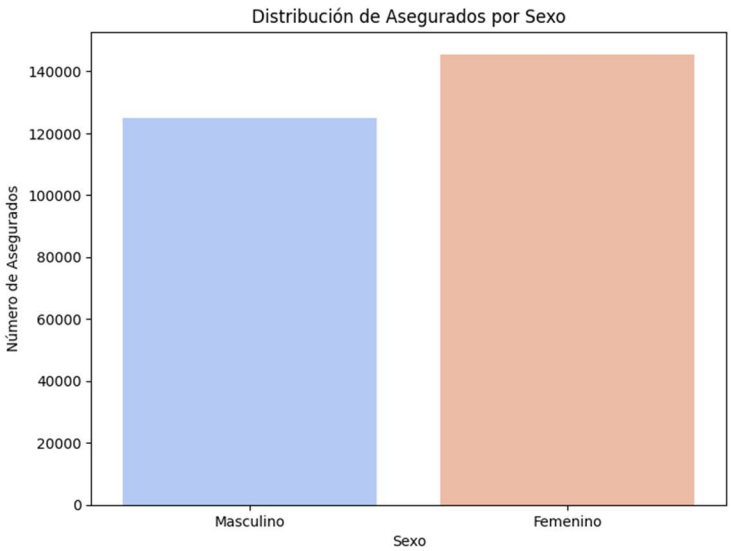


Figura 3: Distribución de asegurados por sexo

Esta diferencia es consistente con las tendencias observadas en los sistemas de pensiones y seguros, donde las mujeres suelen representar una mayor proporción de los beneficiarios debido a su mayor esperanza de vida en comparación con los hombres. Este aspecto es relevante en el análisis de mortalidad, ya que las mujeres tienden a recibir pagos de pensión por períodos más prolongados, lo que puede influir en la gestión de reservas y en la calibración de modelos predictivos.

Además, la distribución por sexo impacta en las tasas de mortalidad y en la precisión de los modelos de predicción, por lo que es un factor clave a considerar en el análisis de riesgos y en la tarificación de rentas vitalicias.

Proporción de Asegurados Vivos y Fallecidos

El gráfico de torta presentado muestra la distribución de los asegurados según su estado de vida, diferenciando entre **Vivos** y **Fallecidos**. Se observa que el **82.2%** de los asegurados aún se encuentra con vida, mientras que el **17.8%** ha fallecido.

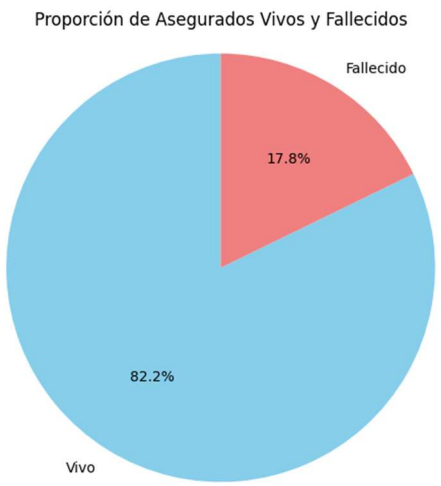


Figura 4: Proporción de asegurados vivos y muertos dentro de la base de datos

Esta distribución es clave para el análisis de mortalidad, ya que la proporción de asegurados fallecidos nos permite evaluar la estabilidad del portafolio y la validez de las predicciones generadas por los modelos utilizados. Una proporción relativamente baja de fallecidos es esperada en una cartera de seguros de largo plazo, pero su evolución en el tiempo es un indicador importante del comportamiento de la mortalidad y de la suficiencia de las reservas técnicas.

Además, esta información es fundamental para calibrar correctamente los modelos predictivos, ya que un desbalance en las clases puede generar sesgos en los resultados. En particular, es crucial para modelos de aprendizaje supervisado como **XGBoost**, que requiere estrategias adecuadas de balanceo para evitar predicciones sesgadas hacia la clase mayoritaria.

Segmentación de Asegurados por Nivel de Prima

El gráfico muestra la distribución de los asegurados según su segmento de prima, dividiéndolos en Alta, Media y Baja. Se observa que la mayor proporción de asegurados pertenece al segmento de prima baja, seguido del segmento de prima media, mientras que la menor cantidad de asegurados se encuentra en el segmento de prima alta.

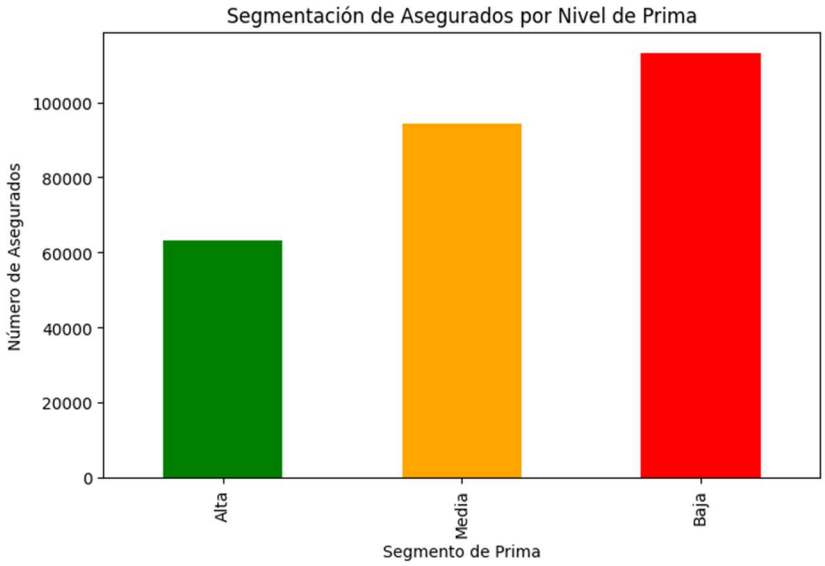


Figura 5: Segmentación de asegurados por nivel de prima

Esta segmentación es relevante en el análisis de mortalidad, ya que los asegurados con primas más bajas pueden estar asociados a perfiles de mayor riesgo o menor esperanza de vida, mientras que aquellos con primas más altas podrían tener mejores condiciones socioeconómicas y acceso a mejores servicios de salud, lo que influye en la longevidad esperada.

Distribución de Asegurados por Región

El gráfico presenta la distribución de los asegurados según su región de residencia, evidenciando una concentración significativa en Región_13, que representa la mayor parte de la población asegurada. Otras regiones con una cantidad considerable de asegurados incluyen Región_5 y Región_8, mientras que el resto de las regiones presentan niveles más homogéneos, pero considerablemente menores en comparación con la principal.

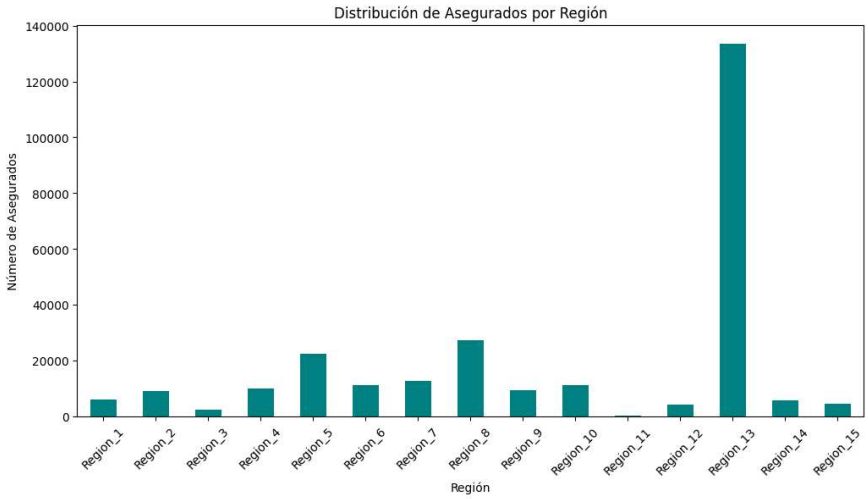


Figura 6: Distribución de asegurados por región

Este hallazgo es relevante para el análisis de mortalidad, ya que las diferencias geográficas pueden influir en la esperanza de vida debido a factores como acceso a servicios de salud, calidad de vida y características socioeconómicas. En este sentido, es importante evaluar si la mortalidad observada en **Región_13** sigue patrones similares a otras regiones o si se requiere un ajuste específico en los modelos predictivos.

Distribución de asegurados por tipo de rol

El gráfico muestra la distribución de asegurados según su rol en la póliza, diferenciando entre causantes y beneficiarios. Se observa que el número de beneficiarios supera al de causantes, lo que es esperable dado que en muchas pólizas de rentas vitalicias un solo causante puede tener múltiples beneficiarios asociados.

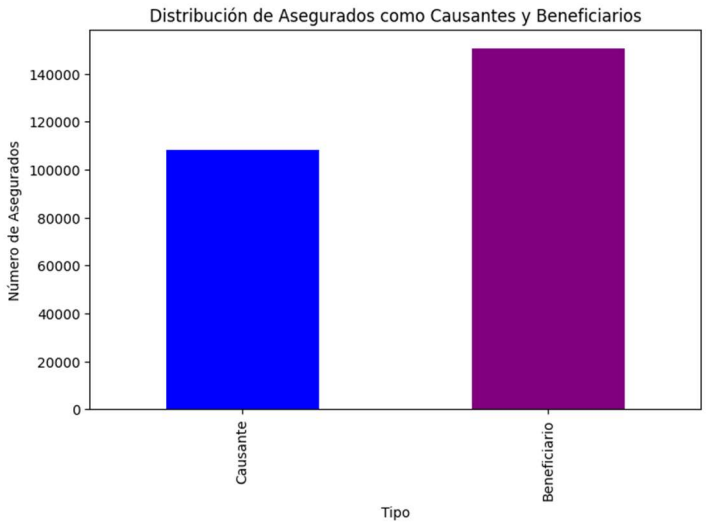


Figura 7: Distribución de asegurados entre causantes y beneficiarios

Proporción de Asegurados Inválidos

El gráfico de torta muestra la distribución de asegurados según su condición de invalidez. Se observa que solo 4.2% de los asegurados tienen la condición de inválidos, mientras que la gran mayoría (95.8%) no presenta esta condición.

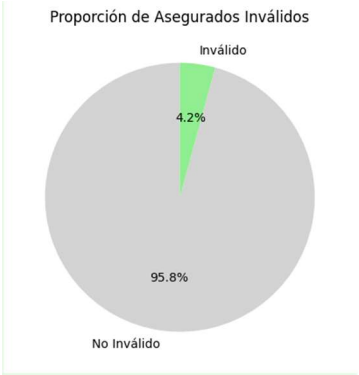


Figura 8: Proporción de asegurados inválidos dentro de la población total

Matriz de Correlación de Variables Clave

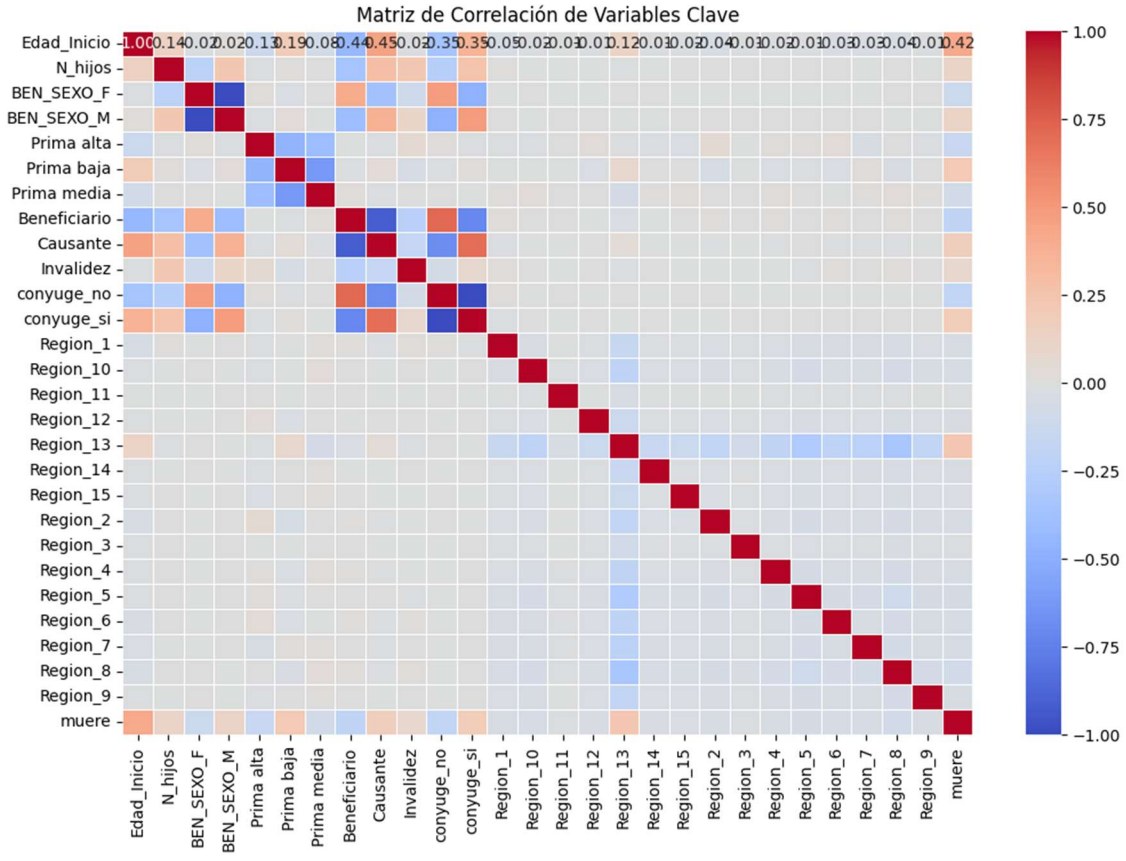


Figura 9: Matriz de correlación de variables (Pearson)

El análisis de correlación revela los siguientes hallazgos importantes:

- ✓ **La edad de inicio tiene una correlación positiva con la mortalidad (+0.42)**, lo que indica que los asegurados de mayor edad tienen mayor probabilidad de fallecer dentro del período analizado.
- ✓ **El nivel de prima presenta correlaciones moderadas con la mortalidad**, con la prima baja mostrando una correlación más positiva con la mortalidad, lo que podría indicar que asegurados con menores ingresos tienen menor esperanza de vida.
- ✓ **El número de hijos muestra una correlación negativa con la mortalidad**, lo que sugiere que las personas con más hijos podrían tener una mejor planificación financiera o recibir apoyo familiar que influya en su longevidad.
- ✓ **Las variables de género presentan correlaciones esperadas**, con **BEN_SEXO_F correlacionado negativamente con la mortalidad**, lo que reafirma la tendencia de que las mujeres tienen una mayor esperanza de vida.

- ✓ **Las variables regionales no muestran una correlación fuerte con la mortalidad**, salvo la Región 13, que presenta una leve correlación positiva, lo que podría estar relacionado con el acceso a sistemas de salud o patrones socioeconómicos de la zona.

Distribución de Asegurados por Nivel de Prima

Tabla de Asegurados por tipo y prima con totales

	Prima alta	Prima media	Prima baja	Total
CAUSANTES	24,305	36,675	47,360	108,340
BENEFICIARIOS	34,723	53,630	62,329	150,682
INVÁLIDOS	4,011	4,028	3,377	11,416
Total	63,039	94,333	113,066	270,438

Proporciones de asegurados

	Prima alta	Prima media	Prima baja	Total
CAUSANTES	8.99%	13.56%	17.51%	40.06%
BENEFICIARIOS	12.84%	19.83%	23.05%	55.72%
INVÁLIDOS	1.48%	1.49%	1.25%	4.22%
Total	23.31%	34.88%	41.81%	100.00%

Tabla 6: Tabla de asegurados por tipo y prima

Las tablas muestran la segmentación de asegurados según el tipo de asegurado (Causante, Beneficiario, Inválido) y su nivel de prima (Alta, Media y Baja). Se observa lo siguiente:

- **Más del 55% de los asegurados son beneficiarios**, mientras que el 40% son causantes y solo un 4.2% corresponden a inválidos.
- **La mayoría de los asegurados se concentran en primas medias y bajas**. Específicamente, el 34.88% de los asegurados pertenecen a la prima media y el 41.81% a la prima baja, dejando solo un 23.31% en la prima alta.
- **Los beneficiarios tienen mayor presencia en los segmentos de prima baja y media**, mientras que los causantes se distribuyen de manera más equitativa en los tres segmentos.

Esta segmentación es clave para evaluar diferencias en la mortalidad y determinar si las predicciones de los modelos varían según el poder adquisitivo de los asegurados.

Distribución de Asegurados por Tipo y Sexo

La siguiente tabla desglosa la cantidad de asegurados según su tipo y género

Tabla de Asegurados por tipo y sexo con totales

	Femenino	Masculino	Total
CAUSANTES	33,992	74,348	108,340
BENEFICIARIOS	108,204	42,478	150,682
INVÁLIDOS	3,337	8,079	11,416
Total	145,533	124,905	270,438

Proporciones de asegurados por sexo

	Prima alta	Prima media	Prima baja
CAUSANTES	12.57%	27.49%	40.06%
BENEFICIARIOS	40.01%	15.71%	55.72%
INVÁLIDOS	1.23%	2.99%	4.22%
Total	53.81%	46.19%	100.00%

Tabla 7: Tabla de asegurados por tipo y sexo

- Las mujeres representan el 53.81% del total de asegurados, mientras que los hombres el 46.19%.
- Entre los beneficiarios, hay una mayor proporción de mujeres (40.01%) en comparación con los causantes (12.57%).
- En el caso de los causantes, el 27.49% son hombres, lo que indica que hay una mayor presencia de hombres como causantes en comparación con beneficiarios.

Este resultado es consistente con la mayor esperanza de vida de las mujeres, lo que explica su predominancia en la categoría de beneficiarios.

Distribución de Asegurados Fallecidos por Sexo y Tipo

La siguiente tabla que muestra la mortalidad observada desglosada por tipo de asegurado y género evidencia los siguientes patrones:

Tabla de asegurados fallecidos por tipo y sexo

	Femenino	Masculino	Total
CAUSANTES	3,958	23,937	27,895
BENEFICIARIOS	14,939	1,447	16,386
INVÁLIDOS	753	3,053	3,806
Total	19,650	28,437	48,087

Proporciones de asegurados fallecidos por sexo

	Prima alta	Prima media	Prima baja
CAUSANTES	8.23%	49.78%	58.01%
BENEFICIARIOS	31.07%	3.01%	34.08%
INVÁLIDOS	1.57%	6.35%	7.91%
Total	40.86%	59.14%	100.00%

Tabla 8: Tabla de asegurados fallecidos por tipo y sexo

- El 58.01% de los fallecidos son causantes, mientras que el 34.08% son beneficiarios y el 7.91% inválidos.
- La mortalidad de los causantes es mayor en hombres (49.78%) que en mujeres (8.23%). Esto refuerza la diferencia en esperanza de vida entre géneros.
- En el caso de beneficiarios, el 31.07% de los fallecidos son mujeres, mientras que solo el 3.01% son hombres. Esto sugiere que, si bien hay más beneficiarios femeninos en la cartera, su tasa de mortalidad es menor en comparación con la de los causantes.
- Los inválidos tienen una menor representación en la población general, pero su proporción de fallecimiento es significativa dentro de su grupo. Este segmento requiere una evaluación diferenciada en la modelización de la mortalidad.

Tasa de Mortalidad por Grupo de Edad

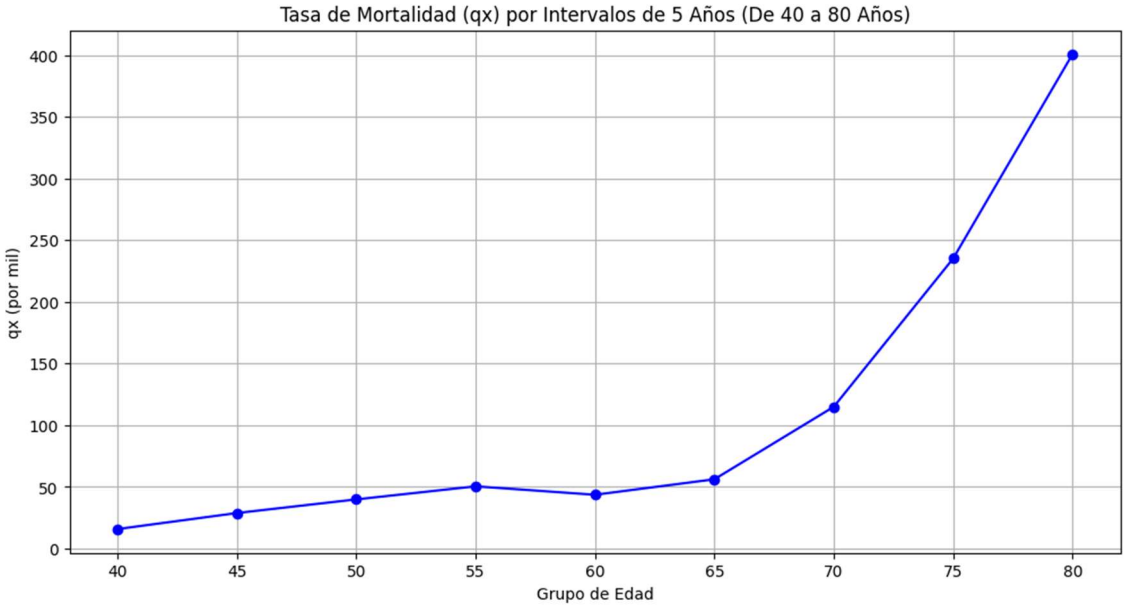


Figura 10: Evolución del la mortalidad dentro de la cartera de rentas vitalicias

El gráfico muestra la evolución de la tasa de mortalidad qx por cada 1,000 asegurados en intervalos de 5 años, abarcando edades entre los 40 y 80 años. Se observan los siguientes patrones clave:

- **Crecimiento Exponencial:** La tasa de mortalidad aumenta gradualmente hasta los 60 años, momento en el cual se estabiliza temporalmente antes de presentar un crecimiento más pronunciado a partir de los 65 años. Este patrón es esperado debido al incremento de riesgos de salud asociados con el envejecimiento.
- **Punto de Inflexión:** A los 70 años, la tasa de mortalidad comienza a aumentar a un ritmo acelerado, alcanzando su valor máximo en el grupo de 80 años, donde supera los 400 fallecimientos por cada 1,000 asegurados.
- **Tendencia Esperada:** La curva sigue la tendencia esperada según modelos actuariales de mortalidad, lo que valida la consistencia de los datos en relación con tablas de mortalidad tradicionales.

Este análisis realizado con la cartera de clientes de rentas vitalicias de nuestra compañía multinacional de seguros refuerza la importancia de segmentar correctamente los asegurados según su edad para evaluar la precisión de los modelos predictivos, ya que la mortalidad no sigue un comportamiento lineal, sino que aumenta de manera exponencial en edades avanzadas.

Resumen del análisis descriptivo

El análisis descriptivo de los datos revela patrones claves en la cartera de clientes de rentas vitalicias de nuestra compañía multinacional de seguros. En términos de distribución etaria, la mayoría de los asegurados se concentran entre los 60 y 75 años, lo que es coherente con la naturaleza de los contratos de rentas vitalicias. En cuanto a género, se observa una ligera preponderancia de aseguradas mujeres sobre los hombres.

El análisis de estado de los asegurados muestra que aproximadamente un 17.8% de la población ha fallecido, lo que sugiere la necesidad de modelos robustos para predecir la mortalidad. Respecto a segmentación por prima, los asegurados con primas bajas representan el mayor porcentaje, lo que indica una mayor presencia de clientes con menores ingresos.

Desde la perspectiva geográfica, la mayoría de los asegurados se concentran en la Región Metropolitana, lo que sugiere que la localización es un factor clave a considerar en el análisis de mortalidad. Además, la segmentación por tipo de asegurado muestra que hay más beneficiarios que causantes, reflejando la estructura típica de este tipo de productos.

Finalmente, la matriz de correlación sugiere que la edad de inicio y la condición de invalidez tienen las correlaciones más significativas con la mortalidad, lo que respalda su inclusión en los modelos predictivos. Además, la tasa de mortalidad (qx) confirma un patrón creciente con la edad, destacando la importancia de ajustar los modelos en función de este comportamiento exponencial.

3. Preprocesamiento de los datos con pycaret

Para esta tesis se empleó la biblioteca pycaret para realizar el preprocesamiento de datos, aprovechando su capacidad para automatizar y simplificar tareas de limpieza y transformación de datos. A continuación, se describen los pasos específicos de preprocesamiento configurados en pycaret.

3.1 Identificación de variables numéricas y categóricas

Pycaret identificó dos variables numéricas y veinticinco variables categóricas en el conjunto de datos. Esta distinción es fundamental para aplicar técnicas de imputación y transformación adecuadas a cada tipo de variable.

- **Numeric features (2)**: Variables numéricas a las que se les aplicará imputación de valores faltantes usando la media.
- **Categorical features (25)**: Variables categóricas que serán transformadas mediante técnicas de codificación y a las que se les aplicará imputación usando la moda.

```
# Importar la librería correcta
from pycaret.classification import *


# Definir las variables numéricas y categóricas a partir del dataframe
numerical_variables = ['Edad_Inicio', 'N_hijos']
categorical_variables = ['BEN_SEXO_F', 'BEN_SEXO_M', 'Prima_alta', 'Prima_media', 'Prima_baja','conyuge_no','conyuge_si',
                         'Causante', 'Beneficiario', 'Invalidez','Region_1','Region_2',
                         'Region_3','Region_4','Region_5','Region_6','Region_7','Region_8','Region_9','Region_10',
                         'Region_11','Region_12','Region_13','Region_14','Region_15']

# Verificar el número total de características disponibles
total_features = len(numerical_variables) + len(categorical_variables)
print(f"Total de características disponibles: {total_features}")

# Ajustar el valor de n_features_to_select si es mayor que el total de características
n_features_to_select = min(50, total_features)
```

Figura 11: Definición de variables dentro del Jupyter notebook

3.2 Activación de preprocesamiento automático

La configuración de **preprocess igual a true** indica que pycaret aplica automáticamente todos los pasos de preprocesamiento necesarios. Esto incluye imputación de valores faltantes, escalado de variables numéricas, codificación de variables categóricas y otros pasos esenciales para preparar los datos para el entrenamiento de modelos. Este parámetro es clave, ya que permite automatizar procesos que de otra manera requerirían implementarse manualmente.

7	Numeric features	2
8	Categorical features	25
9	Preprocess	True
10	Imputation type	simple
11	Numeric imputation	mean
12	Categorical imputation	mode
13	Maximum one-hot encoding	25

Figura 12: Configuración del pycaret dentro del jupyter notebook

3.3. Imputación de valores faltantes

La configuración de **imputation type igual a simple** establece que pycaret empleará una imputación básica para valores faltantes:

- **Numeric imputation = mean**: Los valores faltantes en las variables numéricas son reemplazados por la media de cada variable. Esta técnica ayuda a mitigar el impacto de los valores nulos sin distorsionar excesivamente la distribución de los datos.
- **Categorical imputation = mode**: En el caso de las variables categóricas, los valores faltantes se imputan utilizando la moda, es decir, el valor más frecuente en cada variable. Esto asegura que las categorías dominantes se mantengan y evita la introducción de categorías artificiales.

3.4 Codificación One-Hot de variables categóricas

Para que los modelos de machine learning puedan interpretar variables categóricas, pycaret aplica una codificación one-hot. Esto significa que cada categoría en una variable se convierte en una columna binaria (0 o 1), lo cual evita la introducción de un orden o jerarquía artificial en los datos. La configuración **Maximum one-hot encoding= 25** permite que hasta 25 categorías en las variables categóricas sean codificadas de esta forma. Este límite asegura que no se generen demasiadas columnas adicionales, lo cual podría incrementar la complejidad del modelo y el tiempo de procesamiento.

17 Normalize	True
18 Normalize method	minmax

Figura 13: Configuración del pycaret para el preproceso de escalamiento

3.5 Normalización de datos

En el proceso de preprocesamiento de datos, se ha configurado la normalización automática en pycaret, como se indica en la figura 13. La normalización es fundamental para garantizar que todas las variables numéricas estén en una escala comparable, especialmente cuando se utilizan algoritmos de aprendizaje automático que son sensibles a las magnitudes de las variables.

1. Activación de la Normalización

El parámetro **Normalize = True** permite que pycaret aplique automáticamente la normalización a todas las variables numéricas del conjunto de datos. La normalización es crucial en modelos que se basan en distancias o en gradientes, ya que asegura que cada variable contribuya de manera equitativa en el proceso de entrenamiento del modelo.

2. Método de Normalización: Min-Max

La configuración de **Normalize method = minmax** indica que se está utilizando el método de normalización Min-Max. Este método transforma cada valor en una variable para que esté en un rango de 0 a 1, aplicando la siguiente fórmula:

$$X_{normalizado} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

donde X es el valor original de la variable, X_{min} es el valor mínimo de la variable y X_{max} es el valor máximo de la variable. Este método es especialmente útil cuando se desea preservar las relaciones originales entre los valores, ya que todos los datos se ajustan proporcionalmente en el nuevo rango.

3.6 Desbalanceo de clases

Para abordar el problema de desbalanceo en las clases del conjunto de datos, se utilizó la funcionalidad de balanceo automático de pycaret, configurada para corregir el desbalance mediante el método SMOTE (Synthetic Minority Over-sampling Technique). Esta técnica de sobre-muestreo genera nuevas instancias de la clase minoritaria (en este caso la clase muere) al sintetizar muestras basadas en la combinación de observaciones cercanas en el espacio de características.

15 Fix imbalance	True
16 Fix imbalance method	SMOTE

Figura 14: Configuración del pycaret para el preproceso de balanceo de muestra

El proceso de SMOTE ayuda a mejorar el desempeño de los modelos de clasificación en problemas de clases desbalanceadas al reducir la probabilidad de que los algoritmos de machine learning ignoren o subestimen la clase minoritaria. Esto es especialmente relevante en un modelo de predicción de mortalidad, donde el desbalance podría provocar una menor precisión en la identificación de casos de mortalidad.

3.7 Validación cruzada

Para la validación cruzada del modelo, se empleó un esquema de Stratified K-Fold Cross-Validation con 10 particiones (folds), utilizando la funcionalidad de pycaret. Este método de validación es especialmente útil en problemas de clasificación binaria como la predicción de mortalidad, ya que garantiza que cada partición mantiene la misma proporción de clases (en este caso, vivos y muertos) que el conjunto de datos original.

El uso de Stratified K-Fold asegura que el modelo sea evaluado de manera consistente y robusta en cada fold, proporcionando una mejor estimación de su rendimiento general y reduciendo el riesgo de sobreajuste. Esto es fundamental para validar la capacidad predictiva de los modelos en datos no observados y garantiza una selección de modelo más precisa y confiable.

19	Fold Generator	StratifiedKFold
20	Fold Number	10

Figura 15: Configuración del pycaret para la validación cruzada.

3.8 Preprocesamiento para las Redes Neuronales

Por otro lado, para el entrenamiento de las redes neuronales en esta tesis, se optó por reutilizar las variables preprocesadas generadas a través de pycaret. Esto permitió aprovechar el preprocesamiento ya realizado, asegurando una consistencia en el tratamiento de los datos entre los modelos tradicionales de machine learning y las redes neuronales, y evitando redundancias en el pipeline de preprocesamiento.

En lugar de repetir el preprocesamiento, las variables ya transformadas y ajustadas por pycaret se extrajeron directamente. PyCaret había realizado un preprocesamiento exhaustivo que incluyó la imputación de valores faltantes, la codificación de variables categóricas mediante *one-hot encoding*, la normalización de los datos y el balanceo de clases. Esta extracción aseguró que los datos estuvieran completamente preparados para ser utilizados en las redes neuronales sin necesidad de realizar pasos adicionales de preprocesamiento.

Al utilizar las variables preprocesadas por pycaret, se garantiza que las redes neuronales y los modelos tradicionales (algoritmos de boosting) partan de la misma base de datos. Esto es crucial para una comparación justa de los resultados, ya que ambos enfoques están trabajando sobre datos idénticamente preprocesados. Además, la reutilización del preprocesamiento de pycaret reduce el riesgo de introducir errores adicionales que podrían surgir al duplicar los pasos de transformación.

4. Construcción y preprocesamiento de la serie temporal de fallecimientos

A continuación, se presenta el punto correspondiente al tratamiento de los datos para conformar las series temporales de fallecimientos en la cartera de rentas vitalicias. El objetivo fue transformar la base de datos cruda en una estructura mensual, donde cada fila representara el número de muertes ocurridas en un mes específico desde 1990 hasta 2023. El proceso incluyó:

1. Lectura de la base de datos cruda:

Se utilizó la librería *pandas* para leer el archivo que contenía la información de fallecimientos (columna BEN_FEC_MUE_BEN) y otras variables relevantes.

2. Conversión y limpieza de fechas:

Se transformó la columna de fechas al formato *datetime* (%d/%m/%y) para un correcto manejo temporal. A partir de dicha fecha, se obtuvieron el año y el mes de cada registro para facilitar la agrupación posterior.

3. Creación de la estructura temporal completa:

Se definió un rango de fechas mensuales desde *enero de 1990* hasta *diciembre de 2023* (con pd.date_range), garantizando así que cada mes dentro de ese intervalo figurara en la serie, incluso si no se registraron fallecimientos (en cuyo caso la cuenta sería cero).

Se creó un nuevo *DataFrame* con este rango de fechas, facilitando la posterior fusión de información.

4. Agregación y combinación de datos:

Se realizó una agrupación (*groupby*) por año y mes para contar el número total de fallecimientos en cada periodo.

Se fusionó este resultado con el *DataFrame* de fechas completas, de modo que se pudieran asignar valores de 0 a los meses en los que no hubo muertes o no existía información en la fuente original. En esta etapa aseguró la coherencia temporal y evitó la aparición de lagunas o saltos de datos.

5. Renombrado y exportación de resultados:

Se estandarizaron las columnas (por ejemplo, renombrando a “muertes” la columna final) y se guardó el conjunto resultante en un nuevo archivo CSV, denominado *muerte_series_temporales.csv*, para su uso posterior en el modelado.

6. Visualización preliminar:

Se incluyó un breve análisis gráfico con *matplotlib* para comparar la evolución mensual de fallecimientos en distintos años (2019, 2020, 2021, 2022 y 2023).

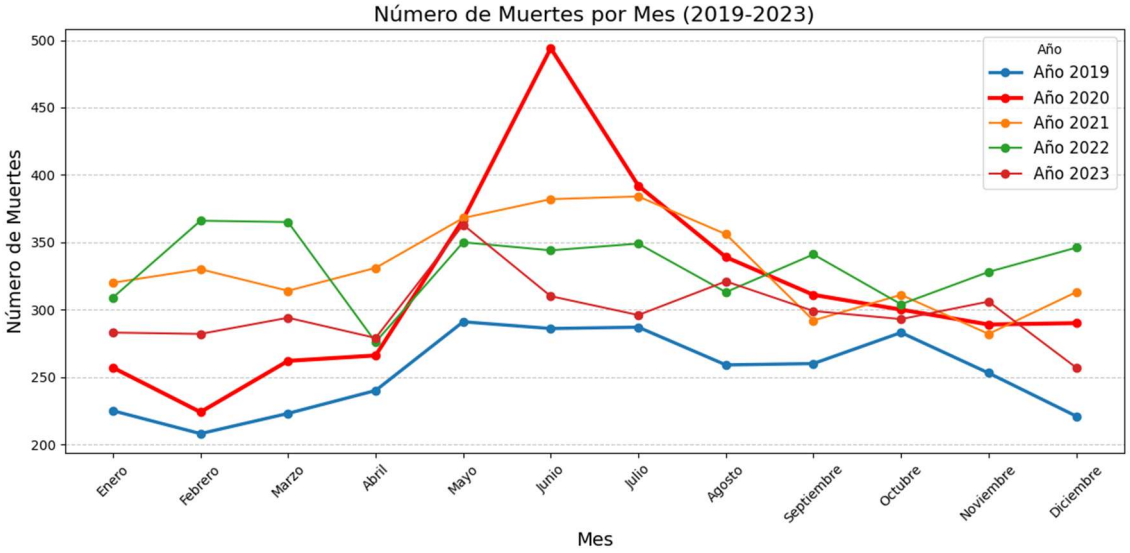


Figura 16: Numero de muertes por meses para los años seleccionados

Esto permitió identificar posibles tendencias estacionales o picos anómalos en la mortalidad que podrán ser capturados por los futuros modelos de series temporales.

Con estos pasos, se logró estandarizar la información mensual de mortalidad y contar con un dataset íntegro y limpio para su posterior análisis. Este preprocesamiento es esencial para la implementación de modelos de series temporales, los cuales requieren datos estructurados en intervalos regulares y coherentes en el tiempo.

5. Configuración de modelos y selección de hiperparámetros para algoritmos de aprendizaje automático

5.1. Modelos de Boosting (LGBMClassifier, XGBClassifier, Gradient Boosting Classifier)

Para los modelos de boosting, como LGBMClassifier, XGBClassifier y Gradient Boosting Classifier, se utilizarán inicialmente las configuraciones predeterminadas de PyCaret en la fase exploratoria del análisis. Esto se debe a que PyCaret está optimizado para seleccionar hiperparámetros estándar que suelen ofrecer un buen rendimiento en una amplia variedad de problemas de clasificación.

Sin embargo, una vez identificado el modelo con mejor desempeño, este será sometido a un ajuste adicional de hiperparámetros mediante validación cruzada y optimización de métricas clave, como la precisión y el recall. Este proceso permitirá refinar el modelo seleccionado para maximizar su capacidad predictiva sin incurrir en sobreajuste.

Los principales hiperparámetros que serán evaluados incluyen:

- **Learning rate**: Define la tasa de aprendizaje en cada iteración de ajuste de los árboles. Se probarán valores en un rango de 0.01 a 0.3, buscando un balance entre convergencia estable y reducción del sobreajuste.
- **Número de árboles (n_estimators)**: Se evaluará un rango de **100 a 500 árboles**, con el objetivo de identificar un valor óptimo que maximice el rendimiento en validación sin incrementar innecesariamente el tiempo de entrenamiento.
- **Max_depth y num_leaves (LGBM)**: Se explorarán profundidades de árbol entre **3 y 10** y un número de hojas entre **20 y 50**. Se espera que profundidades moderadas (entre **5 y 7**) permitan captar patrones relevantes sin sobreajustar el modelo.
- **Subsample y colsample_bytree**: Ambos parámetros ayudarán a **reducir el sobreajuste**, permitiendo la introducción de regularización mediante muestreo de datos y características. Se evaluarán valores de subsample entre 0.5 y 1.0 y colsample_bytree entre 0.6 y 1.0, con el fin de identificar el equilibrio óptimo.
- **Gamma (XGB) y min_samples_split, min_samples_leaf (Gradient Boosting):** Se analizará la mejor configuración para evitar divisiones

innecesarias en nodos con poca ganancia. En XGBoost, Gamma se evaluará en un rango de 0.1 a 0.5, mientras que en Gradient Boosting se explorarán valores de min_samples_split y min_samples_leaf con restricciones mínimas que eviten sobreajustes excesivos.

En la siguiente fase del análisis, estos hiperparámetros serán ajustados utilizando técnicas de búsqueda en malla (Grid Search) y optimización bayesiana, con el objetivo de seleccionar el modelo más robusto y eficiente para su implementación en la predicción de mortalidad.

5.2. Redes Neuronales Profundas (LSTM y GRU)

Para las redes neuronales recurrentes, como LSTM y GRU, la selección de hiperparámetros se realizará mediante pruebas iterativas y observación de métricas de desempeño en el conjunto de validación. Inicialmente, se utilizarán configuraciones estándar, pero posteriormente se explorarán diferentes valores con el objetivo de maximizar la capacidad de modelado y minimizar el sobreajuste.

Los principales hiperparámetros que serán evaluados incluyen:

- **Número de capas y neuronas por capa**: Se analizarán configuraciones entre una y tres capas y se probará con un número de neuronas entre 50 y 200 por capa. Se espera que configuraciones moderadas (entre 1 y 2 capas con 50 a 100 neuronas) permitan capturar relaciones sin inducir sobreajuste.
- **Función de activación**: Se estudiará el impacto de distintas funciones de activación, aunque tanh es un candidato fuerte debido a su capacidad para manejar relaciones no lineales y normalizar salidas entre -1 y 1, lo que podría mejorar la estabilidad del modelo en series temporales.
- **Dropout y recurrent_dropout**: Se probarán valores de dropout entre 0.2 y 0.5, con el objetivo de encontrar un equilibrio entre evitar el sobreajuste y conservar la capacidad de aprendizaje del modelo. Para recurrent_dropout, se evaluará un rango entre 0.1 y 0.2, con el fin de evitar una pérdida excesiva de información en las conexiones recurrentes.
- **Tasa de aprendizaje**: La optimización se realizará utilizando el algoritmo Adam, con pruebas en un rango de 0.001 a 0.01. Se espera que valores cercanos a 0.001 permitan una convergencia estable en GRU, mientras que en LSTM se

considerará un rango entre 0.003 y 0.005 para mejorar la estabilidad del entrenamiento.

- **Unidades ocultas (GRU)**: Dado que las GRU tienen una estructura más simplificada que las LSTM, se probarán entre 50 y 150 unidades ocultas, con un rango óptimo estimado entre 75 y 100, buscando equilibrar el rendimiento y la eficiencia computacional.

En la fase siguiente del análisis, se llevará a cabo una optimización sistemática de estos hiperparámetros mediante validación cruzada y ajuste iterativo, con el objetivo de identificar la configuración que maximice el rendimiento predictivo y minimice el sobreajuste. Esta selección final se basará en métricas de desempeño obtenidas en el conjunto de validación y su capacidad para generalizar correctamente en nuevos datos.

6. Interpretación de los modelo con SHAP values

SHAP (SHapley Additive exPlanations) es un método de interpretación de modelos que asigna a cada característica un valor de importancia en la predicción basada en la teoría de juegos de Shapley. En términos formales, el valor SHAP de una característica se calcula considerando todas las posibles combinaciones de características en el modelo y evaluando cómo cambia la predicción cuando se incluye dicha característica.

El análisis de SHAP permite:

- Identificar la importancia de las variables en el modelo, revelando cuáles son las más influyentes.
- Comprender el efecto (positivo o negativo) de cada variable en la predicción. Esto es crucial para evaluar cómo las características individuales contribuyen a la probabilidad de que un evento ocurra, en este caso, la mortalidad.
- Mejorar la transparencia del modelo en entornos regulatorios, donde la interpretación de decisiones es clave para la confianza y el cumplimiento normativo.

Formalización Matemática

Para un modelo de predicción $f(x)$ que estima una salida de un conjunto de características $x = \{x_1, x_2, x_3, \ldots\ldots, x_n\}$, el valor SHARP de una característica x_i se calcula de la siguiente manera:

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!\,(|N| - |S| - 1)!}{|N|!} (f(S \cup \{i\}) - f(S))$$

Donde:

N es el conjunto de todas las características

$S \subseteq N \setminus \{i\}$ representa un subconjunto de características que no incluye x_i

$(f(S \cup \{i\}) - f(S))$ es el cambio en la predicción cuando se incluye x_i en el conjunto de características S.

En esta ecuación:

ϕ_i es el valor SHAP de la característica x_i , que representa su contribución promedio a la predicción considerando todas las posibles combinaciones con otras características.

El peso $\frac{|S|!(|N|-|S|-1)!}{|N|!}$ asegura que el valor de cada característica se calcule de manera justa y consistente.

Interpretación de los Valores SHAP

El valor SHAP puede interpretarse en dos niveles:

- Globalmente: evaluando la importancia promedio de cada característica en el conjunto completo de datos.
- Localmente: interpretando cómo una característica específica afecta la predicción de un individuo particular.

Un valor SHAP positivo indica que la característica incrementa la predicción del modelo, mientras que un valor SHAP negativo sugiere una disminución en la predicción.

Ejemplo de Interpretación

Supongamos que estamos interpretando una predicción de mortalidad para un individuo específico con los siguientes valores SHAP:

- Edad: $\phi_{edad} = +0.05$
- Nivel de primas: $\phi_{Primas} = -0.03$

En este caso:

- La Edad está aumentando la probabilidad de predicción de mortalidad en 0.05, lo que podría interpretarse como un incremento en la probabilidad de mortalidad.
- El Nivel de primas está reduciendo la probabilidad en 0.03, lo que sugiere que un mayor nivel de primas está asociado con una menor probabilidad de mortalidad.

Aplicación en Resultados: El análisis de SHAP se aplicará a los modelos seleccionados para:

1. Evaluar la importancia global de las variables: obteniendo un ranking de las características que más influyen en la predicción de mortalidad, tales como Edad, Primas y otras variables.

2. Interpretar el signo de las relaciones: analizando si características como la Edad o el Nivel de Primas tienen un efecto positivo o negativo en la probabilidad de mortalidad.

Este enfoque proporcionara una comprensión detallada de cómo cada variable impacta las decisiones del modelo, ayudando a interpretar los resultados y, en última instancia, ofrecer recomendaciones informadas.

El análisis con SHAP asegura una interpretación justa y coherente, proporcionando transparencia en un entorno regulatorio y apoyando la toma de decisiones basadas en los factores más influyentes.

7. Evaluación de las tablas de mortalidad vigentes:

Mensualmente se calcularán los ratios A/E (Actual vs Expected) para comparar los valores de mortalidad reales observados con los valores esperados según las tablas de mortalidad vigentes de la compañía. Esta evaluación se realizará por segmento de cartera (causantes, beneficiarios e inválidos) con el fin de identificar los grupos que presentan mayores desviaciones entre los valores observados y esperados. En lugar de utilizar un promedio móvil de 12 o 24 meses, se trabajará con los resultados puntuales de cada mes, lo que permitirá una detección más inmediata de variaciones en la mortalidad y facilitará el reentrenamiento mensual de los algoritmos de aprendizaje automático y series temporales con datos actualizados. Esta metodología contribuirá a mejorar la precisión en las predicciones de mortalidad.

8. Desarrollo y entrenamiento de algoritmos de aprendizaje automático y series temporales:

En esta etapa, se desarrollarán y entrenarán tanto algoritmos de aprendizaje automático como modelos de series temporales, ajustándolos de forma dinámica a partir de los resultados mensuales de los ratios A/E (Actual vs Expected). Esto permitirá no solo afinar la capacidad predictiva y reflejar tendencias recientes en la mortalidad, sino también capturar estacionalidades y variaciones puntuales. El objetivo principal es que estos métodos superen las proyecciones de las tablas de mortalidad actuales, usando los A/E como métrica clave de desempeño. Con este enfoque iterativo, los algoritmos no solo aprovecharán los patrones históricos, sino que también incorporarán de forma continua cualquier cambio demográfico o socioeconómico reciente, mejorando así la precisión de la mortalidad en la cartera de rentas vitalicias de nuestra compañía multinacional de seguros.

9. Comparación mensual entre algoritmos de aprendizaje automático, modelos de series temporales y las tablas de mortalidad vigentes:

Tras el entrenamiento y selección de los algoritmos de aprendizaje automático y los modelos de series temporales más adecuados, se llevará a cabo una comparación mensual detallada de las predicciones generadas por estos enfoques, los valores esperados según las tablas de mortalidad vigentes y los valores reales observados. La métrica clave para esta comparación será el ratio A/E (Actual vs. Expected), analizado en cada segmento de la cartera (causantes, beneficiarios e inválidos). Este análisis mensual permitirá evaluar de forma más precisa y actualizada la mortalidad, detectando con rapidez cualquier cambio en los patrones observados y asegurando una respuesta ágil en la gestión de reservas, así como en el ajuste de estrategias para el negocio de rentas vitalicias de nuestra compañía multinacional de seguros.

10. Recomendaciones Basadas en los A/E:

Las recomendaciones finales se fundamentarán en el análisis de los resultados obtenidos a partir de los cálculos de A/E (Actual vs. Expected). Si los algoritmos de aprendizaje automático y los modelos de series temporales demuestran una capacidad predictiva consistentemente superior a la de las tablas de mortalidad vigentes en los distintos segmentos de la cartera, se propondrá su implementación como método complementario de proyección de mortalidad para nuestra compañía multinacional de seguros. Asimismo, se analizará la factibilidad de un enfoque híbrido que combine las tablas de mortalidad tradicionales con estas metodologías avanzadas, particularmente en aquellos segmentos donde uno u otro enfoque muestre un mejor rendimiento relativo según los resultados de A/E.

Este análisis permitirá identificar si ciertos segmentos de la cartera (por ejemplo, causantes o beneficiarios) pueden beneficiarse de modelos ajustados dinámicamente mediante aprendizaje automático y series temporales, mientras que otros segmentos podrían continuar usando tablas de mortalidad mejoradas. De esta forma, se optimizará el uso de ambos enfoques en función de sus fortalezas, garantizando así una estrategia de gestión de mortalidad que sea precisa, adaptable y alineada con la evolución de la población asegurada.

Marco Teórico

1. Mortalidad y su importancia para el negocio de Rentas Vitalicias

La mortalidad se refiere a la frecuencia de muertes en una población específica y se ha estudiado extensamente en la demografía y la estadística actuarial debido a su impacto en las proyecciones de longevidad y en la solvencia financiera de las compañías de seguros. En el negocio de rentas vitalicias, la precisión en las estimaciones de mortalidad es crítica, ya que estas proyecciones permiten establecer las reservas necesarias para respaldar los pagos a los asegurados durante sus vidas.

Las rentas vitalicias son contratos de largo plazo en los que una compañía de seguros garantiza pagos regulares a una persona hasta su fallecimiento. Dado que la duración de estos pagos está directamente ligada a la longevidad del asegurado, la capacidad para predecir la mortalidad de forma precisa influye en la estabilidad financiera de la compañía y en la competitividad de sus productos en el mercado. Una subestimación de la mortalidad podría llevar a una sobreestimación en las reservas, mientras que una sobreestimación de la mortalidad podría resultar en insuficiencia de reservas, afectando tanto a la compañía como a los asegurados.

2. Tablas de mortalidad: Conceptos y Metodologías de Construcción

Las tablas de mortalidad son herramientas actuariales diseñadas para representar la probabilidad de muerte a diferentes edades dentro de una población. Son esenciales en la valuación de contratos de seguro y rentas vitalicias, ya que proporcionan una base estadística para calcular el riesgo de longevidad y determinar las reservas requeridas. Las tablas de mortalidad pueden construirse para diferentes poblaciones y segmentos, tomando en cuenta variables como edad, sexo y estado de salud.

2.1. Fuentes de Datos para Tablas de Mortalidad

Las tablas de mortalidad se construyen a partir de datos de mortalidad histórica que incluyen observaciones sobre el número de muertes y la exposición al riesgo para cada grupo etario en un periodo específico. Los datos utilizados generalmente provienen de estadísticas nacionales de mortalidad, censos y bases de datos actuariales internas, especialmente cuando se requieren tablas personalizadas para poblaciones específicas, como la cartera de asegurados de una compañía de seguros.

2.2. Cálculo de las Tasas de Mortalidad

La tasa de mortalidad q_x para una edad específica x se calcula dividiendo el número de muertes observadas en esa edad por el número de individuos expuestos al riesgo. Este cálculo estadístico implica suposiciones sobre la independencia de los eventos de muerte y requiere un ajuste para evitar distorsiones en los datos, especialmente en poblaciones pequeñas o con características atípicas.

2.3. Métodos de Graduación y Ajuste de las Tasas

Para representar de manera precisa los patrones de mortalidad, las tasas brutas deben ser ajustadas mediante técnicas de graduación. Este proceso permite suavizar las irregularidades y garantiza que las tasas de mortalidad reflejen de forma consistente las tendencias observadas en la población. Entre los métodos de graduación se incluyen:

- **Graduación no paramétrica**, como el suavizado de Whittaker y las series polinómicas, que permiten ajustar las tasas sin suposiciones explícitas sobre la forma de la curva.
- **Graduación paramétrica**, utilizando modelos como Gompertz o Makeham, que asumen una relación exponencial en el incremento de la mortalidad con la edad. Estos modelos son especialmente útiles en el análisis actuarial para representar el envejecimiento.

2.4. Mejoras en las Tablas de Mortalidad

Para ajustar las tablas de mortalidad a los cambios en la longevidad, se incorporan factores de mejoramiento que reflejan avances médicos, tecnológicos y de condiciones de vida. Los factores de mejoramiento permiten ajustar las tasas base proyectando la mortalidad futura como una función de la tasa observada en un año base. Un factor de mejoramiento común se expresa de la siguiente forma:

$$q_{(x,t)} = q_{(x,t_0)} * (1 - AA_x)^{(t-t_0)}$$

donde $q_{(x,t)}$ es la tasa de mortalidad en un año futuro t, y AA_x es el factor de mejoramiento calculado para una edad x. Aunque estos factores brindan cierta flexibilidad, su aplicación sigue siendo limitada en contextos de variación rápida o eventos inesperados, como pandemias.

3. Limitaciones de las Tablas de Mortalidad Convencionales

Las tablas de mortalidad tradicionales, si bien son una herramienta confiable para la proyección de riesgos en horizontes de largo plazo, enfrentan limitaciones cuando los patrones de mortalidad cambian drásticamente o cuando hay factores estacionales que generan fluctuaciones temporales en la mortalidad. La pandemia de COVID-19 es un ejemplo de cómo eventos imprevistos pueden alterar significativamente las tasas de mortalidad, haciendo que los modelos basados exclusivamente en datos históricos no respondan adecuadamente a estas variaciones.

Dado que las tablas convencionales utilizan promedios y factores de mejoramiento estáticos, su capacidad de adaptación a cambios abruptos es limitada. Esto plantea la necesidad de explorar métodos complementarios, como los algoritmos de aprendizaje automático y series temporales, que puedan adaptarse de manera dinámica y continua a nuevos patrones de mortalidad observados.

4. Algoritmos de aprendizaje automático en la predicción de mortalidad

Los algoritmos de aprendizaje automático ofrecen un enfoque innovador para la predicción de mortalidad, especialmente en contextos donde las tendencias de mortalidad pueden variar significativamente en el tiempo. A diferencia de las tablas de mortalidad tradicionales, que son estáticas, los algoritmos de aprendizaje automático permiten un ajuste dinámico en función de los datos disponibles, lo cual los hace particularmente útiles para las compañías de seguro y AFPs en donde la que la precisión en la estimación de mortalidad tiene un impacto financiero directo.

4.1. Formulación matemática de modelos de machine learning

Los modelos de machine learning, especialmente los algoritmos supervisados, se basan en la relación entre una variable objetivo y (en este caso, la mortalidad) y un conjunto de variables predictoras $X = (x_1, x_2, x_3, x_4, \dots, x_n)$ que pueden incluir edad, sexo, estado de salud, entre otros factores demográficos y socioeconómicos.

4.2. Modelos de Boosting para la predicción de mortalidad

Los modelos de boosting son una clase de algoritmos de machine learning que construyen modelos de manera secuencial, donde cada modelo intenta corregir los errores del modelo anterior. Este proceso resulta en un modelo final que es una combinación de varios modelos “débiles” (árboles de decisión en este caso), que en conjunto conforman un modelo “fuerte” con alto poder predictivo.

4.2.1 LGBM Classifier (Light Gradient Boosting Machine)

LightGBM es una implementación optimizada del algoritmo de boosting que utiliza una técnica llamada “Leaf-wise growth” en lugar de “Level-wise growth” tradicional. En cada iteración, LightGBM selecciona el nodo de la hoja que maximiza la ganancia de información, lo que reduce significativamente el tiempo de entrenamiento en comparación con el boosting tradicional.

1. **Función de Pérdida**: LightGBM minimiza una función de pérdida L definida como:

$$L = \sum_{i=1}^{N} l(y_i, \widehat{y_i}) + \sum j = 1^k \, \Omega(f_j)$$

donde:

- $l(y_i, \widehat{y_i})$ es el error entre la predicción $\widehat{y_i}$ y el valor real y_i
- $\Omega(f_j)$ es un término de regularización para evitar el sobreajuste, definido como:

$\Omega(f_j)$ = $\gamma T + \frac{1}{2}\lambda \sum_{k=1}^{T} w_k^2$

Donde T es el número de hojas en el árbol y w_k es el peso de cada hoja.

2. **Optimización**: LightGBM utiliza el gradiente descendente para minimizar la función de pérdida, calculando el gradiente g_i y el Hessiano h_i (segunda derivada) para ajustar el modelo:

$$g_i = \frac{dl(y_i, \widehat{y_i})}{d\widehat{y_i}} \quad , \qquad h_i = \frac{d^2 l(y_i, \widehat{y_i})}{d\widehat{y_i}^2}$$

3. **Actualización de Predicciones**: Para cada nodo en el árbol, el modelo ajusta los valores de los nodos usando los gradientes g_i y Hessians h_i :

$$w_k = -\frac{\sum_{i \epsilon leaf_k} g_i}{\sum i \epsilon leaf_k h_i + \lambda}$$

Este ajuste en las hojas del árbol permite a LightGBM mejorar las predicciones iterativamente.

4.2.2 XGBClassifier (Extreme Gradient Boosting)

XGBoost es otra implementación de gradient boosting optimizada que se enfoca en la precisión y la eficiencia computacional. Utiliza la segunda derivada de la función de pérdida (Hessiano) para ajustar el modelo, similar a LightGBM, pero con algunas diferencias clave en su método de regularización y optimización.

1. **Función de Pérdida**: Al igual que en LightGBM, XGBoost minimiza una función de pérdida:

$$L = \sum_{i=1}^{N} l(y_i, \widehat{y_i}) + \sum j = 1^k \, \Omega(f_j)$$

En XGBoost, la regularización de cada árbol se define de manera similar:

$$\Omega(f_j) = \gamma T + \frac{1}{2}\lambda \sum_{k=1}^{T} w_k^2$$

2. **Expansión de Taylor de Segundo Orden**: XGBoost optimiza la función de pérdida mediante una expansión de Taylor de segundo orden, aproximando la función de pérdida usando el gradiente g_i y el Hessian h_i:

$$L^{(t)} \approx \sum_{i=1}^{N} \left[g_i f(x_i) + \frac{1}{2} h_i f(x_i)^2 \right] + \Omega(f^{(t)})$$

Esta aproximación permite que XGBoost realice ajustes más precisos en cada iteración.

3. **Criterio de División (Split Criterion)**: XGBoost calcula la ganancia de información para cada posible división de nodo, seleccionando la que maximiza la reducción en la pérdida:

$$Gain = \frac{1}{2}\left[\frac{\left(\sum_{i\epsilon left} g_i\right)^2}{\sum_{i\epsilon left} h_i + \lambda} + \frac{\left(\sum_{i\epsilon right} g_i\right)^2}{\sum_{i\epsilon right} h_i + \lambda} - \frac{\left(\sum_{i\epsilon all} g_i\right)^2}{\sum_{i\epsilon all} h_i + \lambda} \right] - \gamma$$

donde el término γ controla el número de divisiones, lo que evita divisiones innecesarias que puedan causar sobreajuste.

4.2.3 Gradient Boosting Classifier

El **Gradient Boosting** clásico se centra en minimizar iterativamente una función de pérdida mediante la adición secuencial de modelos. A diferencia de LightGBM y XGBoost, este modelo no necesariamente utiliza una expansión de Taylor de segundo orden, lo que lo hace menos complejo matemáticamente, pero aún eficaz en muchos casos.

1. **Función de Pérdida y Gradiente**: En cada iteración t, el modelo agrega un nuevo árbol $f(x)_t$ que minimiza la función de pérdida. La predicción en cada paso se define como:

$$\hat{y}^{(t)} = \hat{y}^{(t-1)} + \eta\, f(x)_t$$

Donde η es el factor de aprendizaje y controla la contribución de cada árbol al modelo final.

2. **Cálculo de Residuos**: En cada iteración, se calculan los residuos r_i^t que representan el error entre la predicción actual y el valor real:

$$r_i^t = y_i - \hat{y}^{(t-1)}$$

Estos residuos se utilizan para ajustar el siguiente árbol, que se entrena para predecir los errores restantes en lugar del valor objetivo directamente.

3. **Actualización del Modelo**: El nuevo modelo ajustado a los residuos se agrega al modelo acumulativo para corregir los errores. De esta forma, Gradient Boosting Classifier minimiza los errores progresivamente, aproximando cada vez más la función de predicción a los valores reales.

$$\hat{y}^{(t)} = \hat{y}_i^{(t-1)} + \eta f_t(x_i)$$

4. **Criterio de División y Selección de Características**: Gradient Boosting utiliza el criterio de reducción de pérdida en cada nodo del árbol para decidir la mejor división. No obstante, no optimiza mediante una expansión de segundo orden como XGBoost o LightGBM, por lo cual suele ser un poco más lento y menos eficiente en datasets grandes.

Comparación matemática entre los modelos de boosting

Cada uno de estos modelos sigue la misma idea general de gradient boosting, pero con variaciones en la regularización y en la forma de optimización:

- **LightGBM** y **XGBoost** utilizan una expansión de Taylor de segundo orden para optimizar la función de pérdida, lo que permite divisiones y ajustes de predicción más precisos.
- **Gradient Boosting Classifier** es un modelo de boosting más clásico que depende solo de la primera derivada, y, por lo tanto, es más simple y directo en su implementación, aunque menos preciso en comparación.

Estos modelos representan un avance significativo respecto a los modelos de predicción tradicionales en el ámbito actuarial, ya que pueden adaptarse a datos complejos y permitir ajustes en tiempo real, lo cual es crucial en la predicción de mortalidad para rentas vitalicias.

4.3 Redes neuronales (NN) y su aplicación en la predicción de mortalidad

Las redes neuronales son un subconjunto del aprendizaje automático y están en el corazón de los algoritmos de aprendizaje profundo. Están compuestas por capas de nodos, que contienen una capa de entrada, una o más capas ocultas y una capa de salida. Cada nodo se conecta a otro y tiene un peso y un umbral asociados.

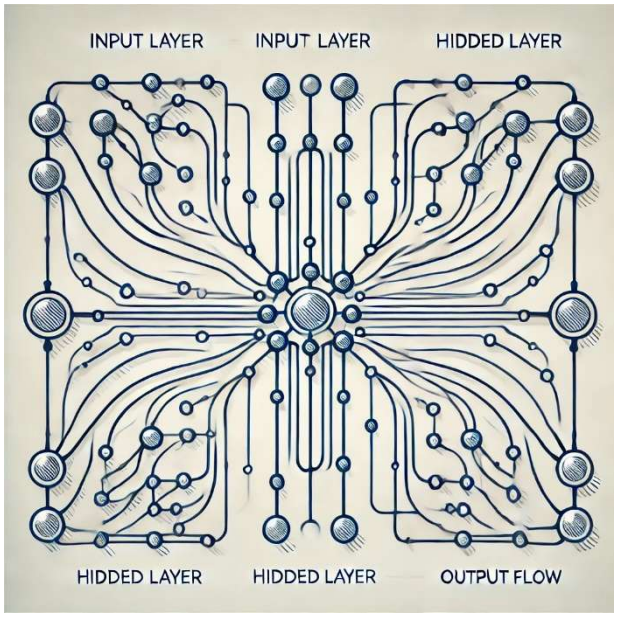


Figura 17: OpenAI. (2024). Neural network illustration.

Redes Neuronales Recurrentes (RNN)

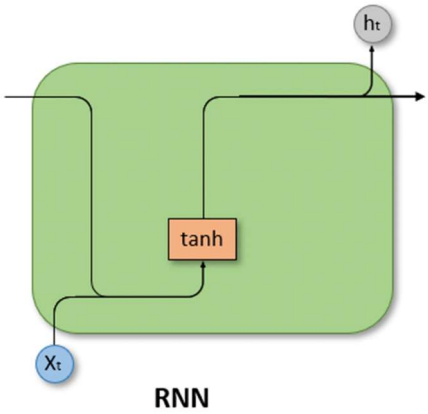


Fig 18: RNN architecture

Las redes neuronales recurrentes (RNN) son un tipo de modelo de aprendizaje profundo diseñado para procesar secuencias de datos. A diferencia de las redes neuronales tradicionales, las RNN pueden manejar entradas secuenciales, como series temporales o texto, y generar una salida específica basada en la secuencia completa.

Lo que diferencia a las RNN es la incorporación de conexiones recurrentes entre los nodos, lo que les permite mantener una memoria interna. Esto significa que la información de los pasos previos se retroalimenta en la red y afecta el procesamiento de los siguientes pasos. De esta manera, las RNN pueden aprender dependencias temporales y manejar secuencias de longitud variable, capturando patrones contextuales de los datos anteriores para predecir resultados futuros.

Limitaciones de las RNN

1. Desvanecimiento del Gradiente:

Las RNN presentan dificultades para aprender relaciones de largo plazo en secuencias extensas debido al problema del desvanecimiento del gradiente. Este fenómeno ocurre cuando, durante el entrenamiento, los gradientes que indican la magnitud y dirección de los ajustes de los pesos se vuelven cada vez más pequeños a medida que retroceden por las capas de la red. Como resultado, la red tiene dificultades para actualizar de manera efectiva sus pesos, lo que impacta su capacidad para capturar patrones a largo plazo. Esto puede dar lugar a un entrenamiento lento y, en algunos casos, a la incapacidad de la red para aprender dependencias importantes.

Existen varias técnicas para mitigar este problema, como la inicialización adecuada de los pesos, la normalización por lotes, el recorte de gradientes y las conexiones de salto. Estas estrategias ayudan a estabilizar el flujo de gradientes y permiten un aprendizaje más eficiente de las relaciones a largo plazo en las secuencias.

2. Limitaciones de memoria:

Las RNN también enfrentan restricciones de memoria cuando se enfrentan a secuencias muy largas. Debido a la naturaleza secuencial de su arquitectura, las RNN deben almacenar y procesar información de manera progresiva, lo que aumenta su carga computacional y puede afectar su rendimiento en secuencias extensas.

LSTM y GRU

Para superar estas limitaciones, se han desarrollado variantes de las RNN, como las LSTM (Long Short-Term Memory) y las GRU (Gated Recurrent Unit). Ambas arquitecturas están diseñadas para mantener información a largo plazo y mitigar el problema del desvanecimiento del gradiente. Estas estructuras incluyen mecanismos de puertas que controlan el flujo de información a través de la red, permitiendo a la memoria interna actualizarse de manera eficiente y preservar la información relevante durante períodos más largos.

Gated Recurrent Units (GRU)

Arquitectura GRU

Las GRU combinan los conceptos de estado oculto y celda en un solo vector, lo que simplifica el flujo de información. La fórmula para actualizar el estado oculto en una GRU es más sencilla, lo que facilita su implementación y uso.

Las GRU utilizan puertas que controlan el flujo de información a través de la red, pero su estructura es más simple. La principal diferencia con las LSTM es que las GRU combinan algunas de las funciones de las puertas, lo que resulta en un menor número de parámetros y un tiempo de entrenamiento más rápido. En lugar de las tres puertas presentes en las LSTM, las GRU tienen solo dos:

- Puerta de reinicio (Reset Gate): Controla cuánta información previa se debe olvidar en el estado actual. Permite que la red decida cuándo reiniciar la memoria para capturar nuevas características de la secuencia.
- Puerta de actualización (Update Gate): Determina cuánto de la información anterior se debe llevar al siguiente estado, controlando la cantidad de memoria que se retiene de la información previa y cuánto de la nueva información se introduce.

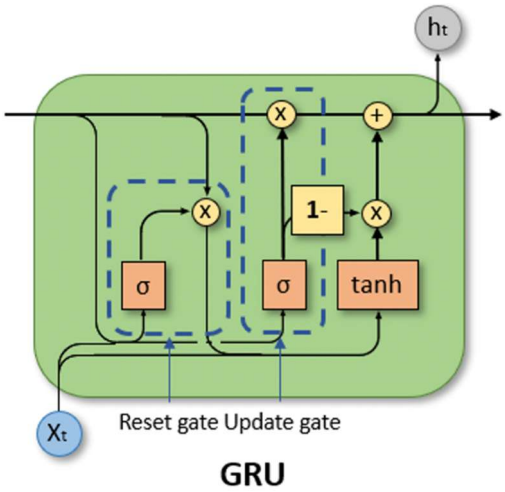


Figura 19. GRU architecture.

Fórmulas matemáticas de las GRU

Como se mencionó anteriormente la arquitectura de una celda GRU consta de dos puertas principales:

1. Puerta de actualización (z_t)
2. Puerta de reinicio (r_t)

Estas puertas gestionan el flujo de información, determinando qué información debe ser retenida y qué debe ser olvidada para influir en las predicciones futuras.

Puerta de Actualización (z_t):

La puerta de actualización controla cuánto del estado oculto anterior (h_{t-1}) se retiene y cuánto del nuevo estado candidato $\widetilde{h_t}$ se incorpora. Esta puerta permite equilibrar la preservación de la información pasada con la incorporación de nueva información.

La fórmula para la puerta de actualización es:

$$z_t = \sigma(W_z[h_{t-1}, x_t] + b_z)$$

donde:

- z_t: Puerta de actualización en el paso de tiempo t.
- σ: Función de activación sigmoide, que mapea el valor entre 0 y 1.

- W_z: Matriz de pesos asociada con la puerta de actualización.
- h_{t-1}: Estado oculto previo.
- x_t: Entrada actual.
- b_z: Término de sesgo para la puerta de actualización.

La puerta de actualización z_t decide en qué medida el estado oculto anterior h_{t-1} contribuye al estado oculto actual h_t.

Puerta de Reinicio (r_t):

La puerta de reinicio controla cuánto del estado oculto anterior debe "olvidarse". Cuando la puerta de reinicio se acerca a cero, la red "reinicia" su memoria, ignorando el estado oculto anterior. Esto permite a la GRU olvidar información irrelevante del pasado.

La fórmula para la puerta de reinicio es:

$$r_t = \sigma(W_r[h_{t-1}, x_t] + b_r)$$

donde:

- r_t: Puerta de reinicio en el paso de tiempo t.
- W_r: Matriz de pesos asociada con la puerta de reinicio.
- b_r: Término de sesgo para la puerta de reinicio.

Estado Candidato ($\widetilde{h_t}$)

El estado candidato, $\widetilde{h_t}$, es un estado temporal que combina la entrada actual y el efecto de la puerta de reinicio en el estado oculto anterior. Este estado candidato se combina después con el estado oculto anterior, controlado por la puerta de actualización, para producir el estado oculto final h_t.

La fórmula para el estado candidato es:

$$\widetilde{h_t} = tanh(W[r_t * h_{t-1}, x_t] + b)$$

donde:

- $\widetilde{h_t}$: Estado candidato en el paso de tiempo t.
- tanh: Función de activación tangente hiperbólica, que mapea valores entre -1 y 1.

- W: Matriz de pesos asociada con el estado candidato.
- b: Término de sesgo para el estado candidato.
- $r_t * h_{t-1}$: Multiplicación elemento a elemento entre la puerta de reinicio y el estado oculto anterior, permitiendo que la red decida cuánto del estado oculto previo influirá en el estado candidato.

Estado Oculto Final (h_t)

El estado oculto final h_t en el paso de tiempo t se calcula combinando el estado oculto anterior h_{t-1} y el estado candidato $\widetilde{h_t}$, utilizando la puerta de actualización z_t.

La fórmula para el estado oculto final es:

$$h_t = (1 - z_t) * h_{t-1} + z_t * \widetilde{h_t}$$

donde:

- h_t: Estado oculto final en el paso de tiempo t.
- $(1 - z_t) * h_{t-1}$: Contribución del estado oculto anterior, ponderada por $1 - z_t$, indicando la cantidad de información pasada que se retiene.
- $z_t * \widetilde{h_t}$: Contribución del estado candidato, ponderada por z_t, indicando la cantidad de nueva información que se incorpora.

Long Short-Term Memory (LSTM)

Las LSTM abordan el problema del desvanecimiento de gradientes al incorporar celdas de memoria especializadas y mecanismos de compuerta que preservan y controlan el flujo de gradientes en secuencias extendidas. Esto permite que la red capture dependencias a largo plazo de manera más efectiva, mejorando significativamente su capacidad para aprender de datos secuenciales. Las LSTM cuentan con tres compuertas (entrada, olvido y salida) y se destacan en la captura de dependencias a largo plazo.

Arquitectura LSTM

Las LSTM están formadas por celdas o celdas de memoria que tienen la capacidad de mantener información durante períodos prolongados. Cada celda de memoria tiene tres puertas principales:

- Puerta de entrada (Input Gate): Determina qué nueva información entra en la celda de memoria y cómo se actualiza.
- Puerta de olvido (Forget Gate): Decide qué información de la celda se debe olvidar y qué información se debe conservar.
- Puerta de salida (Output Gate): Controla la salida de la celda y determina qué partes de la memoria interna se utilizarán para la siguiente salida.

Estas puertas se controlan mediante funciones sigmoides y tangentes hiperbólicas, que permiten regular el flujo de información a través de la red. Esto posibilita que las LSTM recuerden o ignoren ciertos datos a medida que procesan la secuencia.

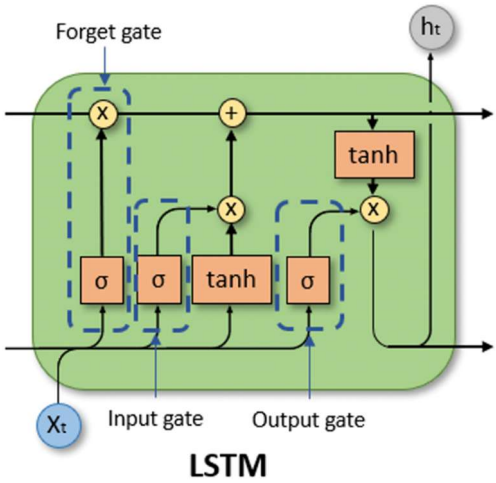


Figura 20. LSTM architecture.

Componentes Principales y Fórmulas Matemáticas de las LSTM

Cada celda LSTM tiene tres "puertas" principales:

1. Puerta de olvido (f_t)
2. Puerta de entrada (i_t)
3. Puerta de salida (o_t)

Además, las LSTM cuentan con un estado de celda (C_t), que actúa como un "canal" de memoria a largo plazo, y un estado oculto (h_t), que se actualiza en cada paso de tiempo. A continuación, se describen las operaciones matemáticas de cada componente.

1. Puerta de Olvido (f_t)

La puerta de olvido controla cuánta información del estado de celda anterior (C_{t-1}) debe ser retenida o descartada. Esta puerta permite a la LSTM "olvidar" cierta información que ya no es relevante.

La fórmula para la puerta de olvido es:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)$$

donde:

- f_t: Valor de la puerta de olvido en el paso de tiempo t.
- σ: Función de activación sigmoide, que mapea el valor entre 0 y 1.
- W_f: Matriz de pesos de la puerta de olvido.
- h_{t-1}: Estado oculto en el paso de tiempo anterior.
- x_t: Entrada en el paso de tiempo actual.
- b_f: Término de sesgo para la puerta de olvido.

Cuando f_t se aproxima a 1, se retiene más información del estado de celda anterior; cuando se acerca a 0, se olvida más.

2. Puerta de Entrada (i_t) y Estado de Candidatura de Celda ($\widetilde{C_t}$)

La puerta de entrada decide cuánta de la nueva información se almacenará en el estado de celda actual. Junto con la puerta de entrada, se calcula el estado de candidatura de celda ($\tilde{C}_t$), que representa la nueva información potencial que podría agregarse al estado de celda.

Las fórmulas para la puerta de entrada y el estado de candidatura de celda son:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh\,(W_c[h_{t-1}, x_t] + b_c)$$

donde:

- i_t: Valor de la puerta de entrada en el paso de tiempo t.
- $\tilde{C}_t$: Estado de candidatura de celda en el paso de tiempo t.

- tanh: Función de activación tangente hiperbólica, que mapea el valor entre -1 y 1.
- W_i: Matriz de pesos de la puerta de entrada.
- W_i: Matriz de pesos del estado de candidatura de celda.
- b_i y b_c: Términos de sesgo correspondientes.

La puerta de entrada i_t controla cuánto de $\tilde{C}_t$ debe agregarse al estado de celda.

3. Actualización del Estado de Celda (C_t)

El estado de celda (C_t) actúa como una "memoria de largo plazo" en la LSTM. En cada paso de tiempo, C_t se actualiza combinando el estado de celda anterior, ajustado por la puerta de olvido, y la nueva información del estado de candidatura, controlada por la puerta de entrada.

La fórmula para actualizar el estado de celda es:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

donde:

- C_t: Estado de celda actualizado en el paso de tiempo t.
- $f_t * C_{t-1}$: Parte del estado de celda anterior que se retiene.
- $i_t * \tilde{C}_t$: Nueva información agregada al estado de celda.

Esta combinación permite que la LSTM conserve información relevante durante largos periodos y actualice dinámicamente su memoria cuando es necesario.

4. Puerta de Salida (o_t) y Estado Oculto (h_t)

La puerta de salida controla cuánto del estado de celda actualizado se utilizará para el estado oculto (h_t), que es la "salida" de la celda LSTM en el paso de tiempo t. La puerta de salida permite que la LSTM filtre información y determine cuál debe ser la salida en función del estado de celda actual.

Las fórmulas para la puerta de salida y el estado oculto son:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o$$

$$h_t = o_t \tanh\,(C_t)$$

donde:

- o_t: Valor de la puerta de salida en el paso de tiempo t.
- h_t: Estado oculto (salida) de la LSTM en el paso de tiempo t.
- $\tanh(C_t)$: Estado de celda actualizado pasado por la función tangente hiperbólica para mantener los valores entre -1 y 1.
- W_o: Matriz de pesos de la puerta de salida.
- b_o: Término de sesgo para la puerta de salida.

La puerta de salida o_t modula el estado de celda C_t para producir el estado oculto h_t, que es la salida final en cada paso de tiempo.

Importancia de las LSTM y GRU en el Aprendizaje de Dependencias Temporales para Mortalidad

Gracias a sus arquitecturas basadas en puertas, tanto las LSTM como las GRU pueden aprender y recordar patrones en secuencias de datos a largo plazo, siendo ideales para modelar la mortalidad, que puede estar influenciada por factores temporales complejos. A diferencia de las RNN tradicionales, las LSTM y GRU abordan el problema del desvanecimiento del gradiente, permitiendo actualizaciones de pesos eficientes y capturando relaciones complejas entre periodos temporales, lo que es fundamental en la predicción de eventos de vida o muerte.

Esta capacidad de manejar la información a largo plazo hace que las LSTM y GRU se adapten bien a las fluctuaciones y patrones en datos históricos de mortalidad. Por lo tanto, ambos modelos resultan en herramientas potentes para construir predictores de mortalidad confiables, que pueden adaptarse al cambio de patrones de longevidad, estacionalidad y eventos atípicos en la población.

Estas arquitecturas no solo ofrecen precisión en la predicción, sino que permiten una comprensión más detallada de los factores temporales que influyen en la mortalidad, proporcionando una ventaja significativa en la gestión de riesgos en la industria de seguros y en la planificación actuarial.

5.- Modelos de series temporales (SARIMA, GRU y LSTM) y su comparación con enfoques de aprendizaje automático

Los modelos de series temporales se emplean cuando el principal objetivo es pronosticar cómo evolucionará una variable a lo largo del tiempo, considerando explícitamente la dependencia temporal entre observaciones consecutivas. En el ámbito de la mortalidad, esto implica modelar directamente el número de fallecimientos en cada periodo (por ejemplo, mensual o anual) y predecir su valor futuro con base en el patrón histórico. A continuación, se describen los principales enfoques empleados en esta tesis:

1. **SARIMA (Seasonal ARIMA)**

- ✓ **ARIMA** (Autoregressive Integrated Moving Average) es un modelo estadístico clásico que combina componentes autorregresivas (AR), de media móvil (MA) y de integración (I) para ajustar la tendencia y la autocorrelación en los datos.
- ✓ **SARIMA** extiende ARIMA para manejar **estacionalidad**, lo cual resulta relevante cuando la mortalidad presenta fluctuaciones periódicas, por ejemplo, en épocas del año con más incidencia de enfermedades estacionales.

2. **GRU (Gated Recurrent Unit) y LSTM (Long Short-Term Memory) como modelos de series temporales**

- ✓ Tanto GRU como LSTM pertenecen a la familia de **Redes Neuronales Recurrentes (RNN)**, diseñadas para procesar secuencias de datos y capturar relaciones temporales a largo plazo.

Cuando se aplican como modelos de series temporales, se construye una única secuencia que representa la evolución de la mortalidad (por ejemplo, el conteo de fallecidos mensual). El entrenamiento se realiza para pronosticar los valores futuros de la serie partiendo de la evolución histórica.

A diferencia de los métodos estadísticos como SARIMA, las RNN pueden capturar patrones no lineales y complejos, así como reaccionar mejor a cambios abruptos en la tendencia.

5.1 Diferencia al usar LSTM y GRU como algoritmos de aprendizaje automático frente a su uso en series temporales

Algoritmos de aprendizaje automático (enfoque supervisado a nivel individual): En este caso, la red GRU o LSTM se configura como un clasificador o regresor sobre registros individuales. Por ejemplo, cada asegurado constituye una instancia con múltiples características (edad, sexo, estado de salud, variables socioeconómicas, etc.), y el algoritmo aprende a predecir la probabilidad de fallecimiento en un lapso específico. El enfoque está orientado a una predicción por persona, generalmente en un periodo fijo (ej. un mes o un año), y no se modela directamente el conteo total de muertes en el tiempo.

Modelos de series temporales (enfoque secuencial agregado):
Aquí, GRU o LSTM se alimentan con la serie histórica de mortalidad agregada (por ejemplo, número total de fallecimientos al mes en la cartera) y el entrenamiento se orienta a pronosticar el valor de esa serie en los próximos periodos. En lugar de tratar con múltiples individuos, se trabaja con una sola secuencia (la serie temporal), potenciando la capacidad de la RNN para memorizar patrones y estacionalidades a lo largo de todo el historial.

En síntesis, la elección de utilizar GRU o LSTM como un método de aprendizaje automático (enfoque supervisado tradicional) o como un modelo de series temporales depende del tipo de pregunta que se busca responder y de cómo se estructuran los datos. Cuando el objetivo es clasificar o estimar la probabilidad de fallecimiento de un individuo según ciertas características, lo más habitual es un enfoque supervisado a nivel de persona. Sin embargo, si se desea proyectar la evolución de la mortalidad agregada en el tiempo (por ejemplo, la mortalidad mensual de una cartera), se recurre a la formulación secuencial propia de un modelo de series temporales.

SARIMA, GRU y LSTM como modelos de proyección temporal constituyen alternativas robustas para capturar la dinámica de la mortalidad, mientras que sus contrapartes en aprendizaje automático (con la misma arquitectura de red neuronal, pero distinta configuración de entrada-salida) permiten identificar patrones a nivel individual. Ambas aproximaciones pueden aportar información valiosa en el contexto actuarial, ya sea para pronósticos globales de mortalidad o para clasificaciones más detalladas por asegurado.

6. Métricas de evaluación según el enfoque

La métrica A/E (Actual vs Expected) es fundamental para evaluar el desempeño tanto de las tablas de mortalidad convencionales como de los algoritmos de aprendizaje automático y series temporales. Esta métrica compara el número de muertes observadas con el número de muertes predichas. Matemáticamente, el ratio A/E para un grupo poblacional específico se define como:

$$A/_E = \frac{\sum_{i=1}^{N} y_i}{\sum_{i=1}^{N} \hat{y}_i}$$

Donde y_i es el valor real (1 si el individuo fallece, 0 si sobrevive) e $\hat{y}_i$ es la predicción del modelo para el individuo i. Valores cercanos a 1 indican una predicción precisa, mientras que valores significativamente mayores o menores a 1 pueden señalar sobreestimación o subestimación de mortalidad.

Para comparar de forma sólida el desempeño de los algoritmos de aprendizaje automático, los modelos de series temporales y las tablas de mortalidad vigentes, es necesario utilizar métricas adecuadas al tipo de salida de cada enfoque:

6.1 Enfoque de clasificación individual (aprendizaje automático):

- ✓ **Precisión (Accuracy):** Proporción de predicciones correctas sobre el total de casos evaluados.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

1. Precisión y Recall: Para cada clase, se calcula la precisión y el recall, proporcionando una medida de cuán eficaz es el modelo para predecir correctamente la mortalidad (clase positiva) y la supervivencia (clase negativa).

$$Precision_1 = \frac{TP}{TP + FP}$$

$$Recall_1 = \frac{TP}{TP + FN}$$

Estas métricas son útiles para entender la efectividad del modelo al predecir correctamente la mortalidad (clase positiva) y la supervivencia (clase negativa).

2. Matriz de Confusión: Una herramienta visual que permite entender el rendimiento del modelo al mostrar los verdaderos positivos (TP), verdaderos negativos (TN), falsos positivos (FP) y falsos negativos (FN).

3. Coeficiente de Matthews (MCC): Métrica que combina la precisión y el recall y es especialmente útil para evaluar modelos con clases desbalanceadas, como puede ser el caso de la predicción de mortalidad en rentas vitalicias.

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)}(TP + FN)(TN + FP)(TN + FN)}$$

Este conjunto de métricas se aplica cuando el modelo predice la probabilidad de fallecimiento para cada individuo y se decide si es un fallecimiento (1) o no (0) en un periodo determinado.

6.2 Enfoque de Series Temporales (SARIMA, GRU, LSTM como modeladores de secuencias):

Los modelos de series temporales pronostican valores agregados (por ejemplo, fallecimientos mensuales), por lo que no generan una salida binaria (fallece o no fallece un individuo), sino un valor numérico continuo en cada periodo futuro.

En este caso, se utilizan métricas como:

- ✓ **RMSE (Root Mean Squared Error):**

Mide el error promedio al cuadrado entre los valores pronosticados y los reales.

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(\widehat{y_t} - y_t)^2}$$

- ✓ **MAE (Mean Absolute Error):**

Promedio de los valores absolutos de los errores.

$$MAE = \frac{1}{n}\sum_{t=1}^{n}|\widehat{y_t} - y_t|$$

- ✓ **MAPE (Mean Absolute Percentage Error):**

Indica el error porcentual promedio respecto a los valores reales.

$$MAPE = \frac{100\%}{n}\sum_{t=1}^{n}\left|\frac{\hat{y}_t - y_t}{y_t}\right|$$

- ✓ No se emplea la matriz de confusión aquí, ya que no hay clases positivas o negativas, sino pronósticos de valores continuos en el tiempo.

6.3 Tablas de Mortalidad (Método Convencional):

Las tablas de mortalidad son, en esencia, estimaciones deterministas de la probabilidad de fallecimiento para cada edad y sexo, basadas en datos históricos.

La métrica más relevante para evaluar su desempeño frente a los datos reales es el ratio A/E (Actual vs. Expected), que compara el número de fallecimientos observados con el número esperado según la tabla:

$$\frac{A}{E} = \frac{Fallecidos\ observados(reales)}{Fallecidos\ Esperados}$$

No se utiliza Accuracy, Precision, Recall ni matriz de confusión, debido a que el resultado de una tabla de mortalidad no es una clasificación binaria por individuo, sino una probabilidad histórica o tasa para un grupo de asegurados.

En conclusión, cada enfoque (clasificación individual, series temporales y tablas de mortalidad) demanda métricas de evaluación distintas, alineadas con el tipo de salida (binaria vs. continua) y con la naturaleza de la modelación (probabilidades por individuo vs. proyecciones agregadas). Este sistema de métricas permite comparar en términos de exactitud, robustez y pertinencia la eficacia de los distintos métodos para la predicción de mortalidad en la cartera de rentas vitalicias.

7. Justificación del uso de algoritmos de aprendizaje automático y modelos de series temporales para la proyección de la mortalidad

La inclusión de algoritmos de aprendizaje automático y modelos de series temporales en la predicción de mortalidad responde a la necesidad de adaptarse a variaciones rápidas y complejas en las tasas de mortalidad. A diferencia de las tablas de mortalidad convencionales, que se construyen con datos históricos y permanecen relativamente estáticas en el tiempo, estos enfoques pueden reentrenarse periódicamente con nueva información, lo que les permite capturar patrones recientes, detectar tendencias emergentes y ajustarse a cambios demográficos y en la salud pública.

Los modelos de series temporales, como SARIMA, GRU y LSTM, aportan una ventaja adicional al identificar estacionalidades y fluctuaciones cíclicas, lo que contribuye a mejorar la precisión de las proyecciones financieras y permite prever con mayor exactitud la evolución de la mortalidad a lo largo del tiempo. Por su parte, los algoritmos de aprendizaje automático pueden procesar un amplio conjunto de variables individuales, proporcionando una estimación más detallada de la mortalidad en distintos segmentos de la cartera.

En conclusión, la implementación de estos enfoques en el ámbito actuarial representa un avance significativo hacia una predicción más precisa y dinámica de la mortalidad en rentas vitalicias. Al incorporar técnicas modernas de modelado, nuestra compañía de segiros podrá fortalecer su capacidad de análisis y planificación, mejorando la gestión del riesgo en un entorno caracterizado por el cambio demográfico y la aparición de eventos imprevistos.

Descripción del enfoque de solución

El enfoque de solución propuesto consta de los siguientes pasos, diseñados para optimizar la precisión y adaptabilidad en la predicción de mortalidad en la cartera de rentas vitalicias de nuestra compañía de seguros. Para ello, se combinan algoritmos de aprendizaje automático, modelos de series temporales y las tablas de mortalidad vigentes, con el fin de evaluar cuál de estos enfoques proporciona las proyecciones más confiables y representativas.

Paso 1: Recopilación y preprocesamiento de datos

La base de datos de la cartera de rentas vitalicias proporciona información clave, que incluye variables demográficas, de salud y comportamiento histórico de mortalidad. Estas variables se combinan con los valores de A/E (Actual vs Expected) obtenidos a partir de las tablas de mortalidad vigentes, proporcionando una referencia histórica para evaluar la efectividad de los modelos.

El preprocesamiento incluye:

- Limpieza de datos: Imputación de valores faltantes y eliminación de valores atípicos que puedan sesgar el modelo.
- Transformación de variables: Normalización y escalado de variables continuas, codificación de variables categóricas, y balanceo de las clases para evitar un sesgo en la predicción de mortalidad.

Estructuración de datos para modelos de series temporales:

- **SARIMA:** Se genera una serie temporal mensual del número de fallecimientos, asegurando la coherencia temporal de los datos.
- **GRU y LSTM como modelos de series temporales:** Se estructuran los datos en secuencias temporales, permitiendo a las redes neuronales captar dependencias temporales en los patrones de mortalidad.
- **GRU y LSTM como modelos de aprendizaje automático:** Se preparan los datos en formato tabular con características extraídas de las observaciones individuales.

Paso 2: Selección y entrenamiento de modelos

El enfoque de solución integra dos grandes categorías de modelos:

2.1 Algoritmos de Aprendizaje Automático

- **Modelos de Boosting (LGBMClassifier, XGBClassifier, Gradient Boosting Classifier):**

  a) Se basan en la construcción de múltiples árboles de decisión que se entrenan secuencialmente, mejorando los errores de los árboles previos.
  b) Son efectivos en la clasificación binaria de mortalidad (vive/muere) y permiten procesar múltiples variables explicativas.
  c) Su desempeño se mide mediante Accuracy, Precision, Recall y MCC.

- **Redes Neuronales Recurrentes (GRU y LSTM) como algoritmos de aprendizaje automático:**

  a) Se entrenan a nivel individual con variables demográficas y socioeconómicas, buscando estimar la probabilidad de fallecimiento en un periodo determinado.
  b) Su estructura secuencial permite capturar relaciones **no lineales** entre múltiples factores de mortalidad.

2.2 Modelos de Series Temporales

- **SARIMA (Seasonal ARIMA):**

  a) Se entrena utilizando series temporales de mortalidad mensual y captura patrones estacionales y tendencias a largo plazo.
  b) Su evaluación se basa en **RMSE, MAE y MAPE**, midiendo la precisión en la proyección de fallecimientos futuros.

- **Redes Neuronales Recurrentes (GRU y LSTM) como modelos de series temporales:**

  a) Se entrenan con la serie mensual de mortalidad y buscan predecir la cantidad de fallecimientos en los próximos meses en función del comportamiento histórico.
  b) Son más flexibles que SARIMA al capturar tendencias no lineales y patrones más complejos en la mortalidad.

Paso 3: Evaluación y ajuste mensual de los modelos seleccionados

Para asegurar que los modelos reflejen correctamente la evolución de la mortalidad, se realiza una evaluación mensual basada en:

- **Ratio A/E (Actual vs Expected):** Compara los valores reales observados de mortalidad con los valores esperados según las tablas de mortalidad y las predicciones de los modelos.
- **Métricas de Evaluación:**
  a) Para los modelos de aprendizaje automático: Accuracy, Precision, Recall, MCC.
  b) Para los modelos de series temporales: RMSE, MAE y MAPE.
- **Ajuste de Hiperparámetros:**
  a) Los modelos de boosting y redes GRU/LSTM se reentrenan mensualmente con datos actualizados.
  b) Se optimizan parámetros como la tasa de aprendizaje, número de árboles en boosting o tamaño de las capas recurrentes en redes neuronales.
  c) En modelos de series temporales, se ajustan parámetros como el orden autorregresivo en SARIMA y la longitud de secuencias en GRU/LSTM.

Paso 4: Comparación y selección del modelo final

Cada mes, se comparan los resultados de los diferentes enfoques de predicción de mortalidad:

1. Tablas de mortalidad (modelo tradicional).
2. Algoritmos de aprendizaje automático (XGBoost, LGBM, GRU y LSTM).
3. Modelos de series temporales (SARIMA, GRU y LSTM).

Para determinar cuál de estos modelos es el más adecuado para nuestra compañía de seguros, se evaluará su desempeño utilizando datos reales de enero 2024 a junio 2024, es decir, información que los modelos no han visto durante su entrenamiento. Este proceso permitirá analizar la capacidad de generalización de cada modelo y su precisión en la predicción de la mortalidad en un contexto actualizado.

Las métricas utilizadas para la comparación serán:

- **Para los modelos de aprendizaje automático:**

a) Accuracy, Precision, Recall y MCC, evaluando su capacidad para clasificar correctamente los casos de fallecimiento.

- **Para los modelos de series temporales:**

b) RMSE, MAE y MAPE, midiendo el error en la proyección de fallecimientos mensuales.

- **Ratio A/E (Actual vs Expected):**

c) Se calculará para cada modelo, comparando los valores proyectados con los observados en la realidad, lo que permitirá validar su aplicabilidad actuarial.

Si los modelos avanzados presentan una precisión significativamente superior y una mejor capacidad para capturar tendencias y estacionalidades, se considerarán para su implementación.

Adicionalmente, se analizará la viabilidad de un enfoque híbrido, donde:

- Las tablas de mortalidad continúen como referencia base en segmentos con predicciones estables.
- Los modelos de aprendizaje automático se utilicen para estimar probabilidades individuales de mortalidad.
- Los modelos de series temporales ayuden a proyectar la estacionalidad y fluctuaciones en la mortalidad, mejorando la precisión de las proyecciones financieras.

Este análisis garantizará que la selección final del modelo se base en su desempeño en datos reales y en su capacidad de adaptación a cambios recientes en la mortalidad.

Paso 5: Implementación y monitoreo continuo

Una vez seleccionado el modelo final, se establece un sistema de monitoreo continuo para:

- Evaluar la precisión de las predicciones en cada ciclo mensual.
- Ajustar los modelos según los cambios en las tendencias de mortalidad.
- Validar la utilidad de los modelos en la planificación financiera de la compañía.

La implementación de este enfoque permitirá que nuestra compañía de seguros disponga de herramientas más robustas para la proyección de mortalidad, incorporando tendencias recientes y mejorando la estabilidad de las proyecciones financieras en el largo plazo.

Resultados:

En esta sección se presentan los resultados obtenidos tras la evaluación de los algoritmos de aprendizaje automático y las series temporales para la predicción de mortalidad en la cartera de rentas vitalicias de nuestra compañía de seguros. Se utilizó PyCaret para la selección y evaluación de modelos, facilitando la comparación y optimización de los mejores algoritmos en términos de desempeño predictivo. Además, se probaron modelos supervisados basados en redes neuronales recurrentes (Densa, LSTM y GRU) y modelos de series temporales (SARIMA, LSTM y GRU).

1. Proceso de Selección y Evaluación de Modelos

El proceso de selección y evaluación de modelos siguió una metodología estructurada en varias etapas:

1.1 Selección del Mejor Modelo

En primer lugar, se ejecutó la función compare_models() de **pycaret**, permitiendo evaluar distintos algoritmos y seleccionar el que ofreciera un mejor desempeño en términos de **Kappa**. A partir de este análisis, **XGBoost (Extreme Gradient Boosting)** fue identificado como el modelo más robusto para la tarea de predicción de mortalidad.

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
xgboost	Extreme Gradient Boosting	0.8368	0.9016	0.7754	0.5280	0.6282	0.5284	0.5444	4.0860
lightgbm	Light Gradient Boosting Machine	0.8350	0.9035	0.7861	0.5240	0.6288	0.5281	0.5461	3.6640
gbc	Gradient Boosting Classifier	0.8297	0.9032	0.7987	0.5136	0.6252	0.5217	0.5429	15.1420
knn	K Neighbors Classifier	0.8664	0.8404	0.5466	0.6475	0.5926	0.5134	0.5161	30.6700
rf	Random Forest Classifier	0.8321	0.8811	0.7484	0.5194	0.6132	0.5104	0.5243	10.8160
et	Extra Trees Classifier	0.8333	0.8698	0.7370	0.5221	0.6112	0.5090	0.5213	18.5280
dt	Decision Tree Classifier	0.8323	0.8616	0.7363	0.5201	0.6096	0.5068	0.5193	2.9660
ada	Ada Boost Classifier	0.8098	0.8953	0.8104	0.4794	0.6024	0.4881	0.5171	6.3820
svm	SVM - Linear Kernel	0.8090	0.8955	0.8103	0.4797	0.6020	0.4874	0.5168	2.8680
lr	Logistic Regression	0.8048	0.8966	0.8206	0.4719	0.5992	0.4823	0.5146	3.5940
ridge	Ridge Classifier	0.7718	0.8898	0.8445	0.4281	0.5682	0.4348	0.4812	2.6780
lda	Linear Discriminant Analysis	0.7717	0.8898	0.8445	0.4281	0.5682	0.4348	0.4812	3.2920
qda	Quadratic Discriminant Analysis	0.7223	0.7672	0.6641	0.3514	0.4567	0.2939	0.3237	3.2660
nb	Naive Bayes	0.6450	0.7908	0.8086	0.3094	0.4475	0.2562	0.3206	3.0880
dummy	Dummy Classifier	0.8222	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	2.4860

Figura 21: Salida de pycaret con los mejores modelos

1.2 Ajuste de Hiperparámetros

Para optimizar su rendimiento, se aplicó tune_model() con el objetivo de encontrar la mejor combinación de hiperparámetros. Sin embargo, el modelo original de **XGBoost** demostró un mejor desempeño en validación cruzada que su versión ajustada, por lo que se decidió mantenerlo sin modificaciones adicionales.

1.3 Validación del Modelo

Tras la selección del modelo óptimo, se utilizó la función pull() para extraer y analizar los resultados de validación cruzada. Esto permitió confirmar la estabilidad y confiabilidad del modelo seleccionado antes de proceder a su implementación.

2. Evaluación del Desempeño del Modelo

2.1 Evaluación Gráfica del Modelo

Para comprender mejor su desempeño, se generaron visualizaciones utilizando **plot_model()**, incluyendo:

Curva ROC-AUC

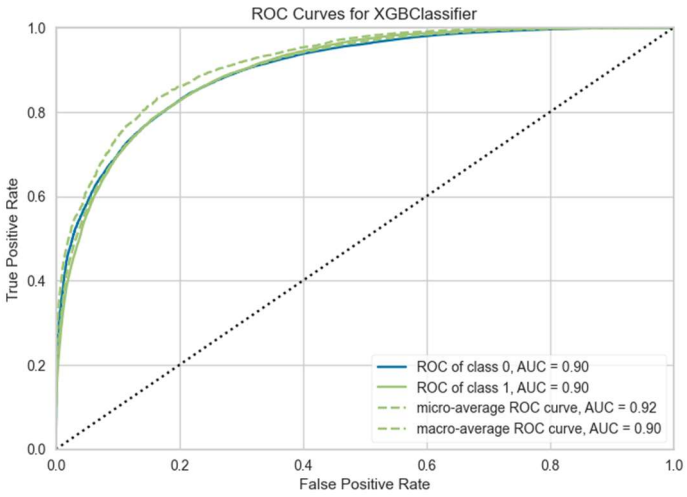


Figura 22: Curva AUC-ROC

La curva ROC muestra la tasa de verdaderos positivos frente a la tasa de falsos positivos. En este caso, el modelo XGBoost obtuvo un AUC de 0.92 en promedio, lo que indica una alta capacidad de discriminación entre clases (muerto y vivo). Esto confirma que el modelo es sólido en términos de clasificación binaria.

Curva de Precisión-Recall

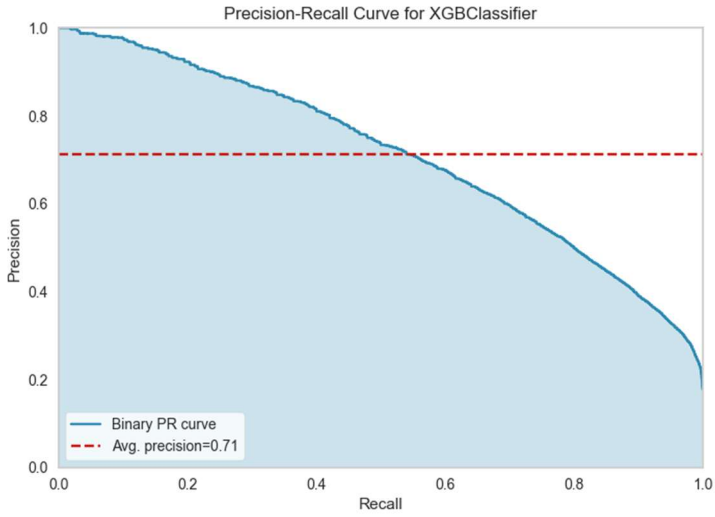


Figura 23: Curva precisión-recall

La curva de precisión-recall es crucial para evaluar el rendimiento del modelo en datasets desbalanceados. Se observa que el modelo mantiene una precisión aceptable en rangos de alto recall, lo que sugiere que es capaz de capturar correctamente la mayoría de los casos de mortalidad, aunque con algunas limitaciones en la precisión.

2.2 Comparación del Rendimiento en Entrenamiento y Prueba

Se realizó una comparación detallada entre los conjuntos de entrenamiento y prueba para evaluar el nivel de generalización del modelo:

Métrica	Training	Test	Diferencia
Precisión(Clase Muerto)	0.5280	0.5224	-0.0056
Re-call (Clase Muerto)	0.7754	0.782	0.0066
F1-Score (Clase Muerto)	0.6282	0.6264	-0.0018
Accuracy Global	0.83	0.83	0.0000
AUC ROC	0.92	0.92	0.0000

Tabla 9: Comparación de métricas para los conjuntos de entrenamiento y prueba

El modelo generaliza bien: No hay grandes diferencias entre training y test, lo que indica que no está sobreajustado.

El recall aumentó en test (0.78): Esto sugiere que el modelo es incluso más eficaz detectando fallecimientos en datos reales.

La precisión se mantuvo estable (~0.52): Aunque sigue habiendo falsos positivos, la tasa no se deteriora en test.

El AUC ROC de 0.92 en ambos datasets confirma que el modelo tiene una alta discriminación de clases.

2.3 Entrenamiento Final y Almacenamiento

Finalmente, se entrenó el modelo con el conjunto de datos completo usando **finalize_model()** y se guardó con **save_model()**, asegurando su disponibilidad para futuras implementaciones en nuestra compañía de seguros.

Conclusión

El modelo XGBoost ha mostrado un desempeño estable y generalizable entre el conjunto de entrenamiento y prueba, lo que indica que puede ser utilizado de manera confiable para predecir la mortalidad en la cartera de rentas vitalicias de nuestra compañía de seguros.

3. Evaluación de Modelos Supervisados Basados en Redes Neuronales

Los modelos de redes neuronales fueron entrenados y evaluados utilizando el mismo conjunto de datos que el modelo XGBoost, con el objetivo de comparar su desempeño en términos de predicción de mortalidad.

Arquitecturas probadas:

1. **Red Neuronal Densa (DNN)**

Para evaluar la capacidad predictiva de una red neuronal en la clasificación binaria de mortalidad, se implementó un modelo de red neuronal densa (MLP - Multi-Layer Perceptron) compuesto por tres capas**:** dos ocultas con 64 y 32 neuronas, respectivamente, utilizando la función de activación **ReLU**, y una capa de salida con activación sigmoide**.** La red fue entrenada utilizando la función de pérdida **binary_crossentropy**, optimizada con **Adam**, y empleando la métrica de precisión (accuracy)**.** Para garantizar una adecuada evaluación del rendimiento generalizado, se utilizó un esquema de validación con un conjunto de prueba independiente.

Finalmente, el modelo fue almacenado en formato **Keras** y **PyCaret** para futuras predicciones y análisis.

2. **Red LSTM (Long Short-Term Memory)**

Para abordar la tarea de clasificación binaria de mortalidad, se implementó una red neuronal LSTM (Long Short-Term Memory) debido a su capacidad para modelar secuencias temporales y capturar dependencias a largo plazo en los datos. La arquitectura del modelo se compone de una capa LSTM con 64 unidades, seguida de una capa Dropout con una tasa de 30% para mitigar el sobreajuste y una capa de salida con una activación sigmoide, adecuada para la clasificación binaria. El modelo fue optimizado mediante el algoritmo Adam con una tasa de aprendizaje de 0.001 y utilizando la función de pérdida binary_crossentropy.

Los datos fueron transformados para cumplir con los requerimientos de entrada de las redes LSTM, donde cada muestra de entrada se estructuró en tres dimensiones: (muestras, pasos de tiempo, características). Se aplicó una estrategia de validación

cruzada utilizando un conjunto de prueba separado para evaluar la capacidad de generalización del modelo.

El modelo fue entrenado por 50 épocas con un tamaño de batch de 64. Durante el entrenamiento, la precisión (accuracy) mostró una tendencia de mejora estable, alcanzando un valor final de **80%**, mientras que el área bajo la curva ROC (AUC) indicó un sólido desempeño en la diferenciación entre clases. La matriz de confusión evidenció una adecuada clasificación de la clase "Muerto" con **un recall de 77.52%**, lo que sugiere que el modelo logra identificar correctamente la mayoría de los casos positivos. No obstante, se observaron algunos falsos positivos, lo que indica que el modelo podría estar capturando ciertas señales erróneas en los datos.

En comparación con otras arquitecturas evaluadas, el modelo LSTM demostró ser una alternativa efectiva para la predicción de mortalidad, ya que logra captar la estructura temporal de los datos y mejorar la capacidad predictiva en escenarios donde la evolución en el tiempo es relevante

3. **Red GRU (Gated Recurrent Unit)**

Para abordar la predicción de mortalidad como un problema de clasificación binaria, se implementó un modelo de red neuronal recurrente basado en Gated Recurrent Units (GRU). Esta arquitectura se seleccionó debido a su capacidad para manejar secuencias temporales, capturando relaciones a largo plazo en los datos. El modelo construido cuenta con una capa GRU de 128 unidades, seguida de una capa de dropout con una tasa del 0.1 para evitar el sobreajuste, y una capa densa de salida con activación sigmoide para la clasificación binaria. Se empleó el optimizador Adam con una tasa de aprendizaje de 0.01 y la función de pérdida binary_crossentropy.

Los resultados obtenidos indican que el modelo alcanzó una precisión (accuracy) del 81.00% en el conjunto de prueba, lo que sugiere un buen desempeño en la clasificación de individuos vivos y fallecidos. La matriz de confusión muestra una adecuada capacidad discriminativa, con una precisión de 0.93 en la clase "Vivo" y un recall de 0.77 en la clase "Muerto". Esto indica que el modelo logra identificar correctamente una alta proporción de fallecimientos, aunque presenta cierta tendencia a clasificar erróneamente algunos casos de personas fallecidas como vivas (falsos negativos). Además, se realizó un análisis de sensibilidad ajustando los umbrales de decisión, observándose que un umbral de 0.5 ofrece un balance óptimo entre precisión y recall.

Finalmente, el modelo fue almacenado en formato HDF5 para facilitar su reutilización y comparación con otras arquitecturas.

4. Comparación del rendimiento en entrenamiento y prueba

Se realizó una comparación detallada entre los conjuntos de entrenamiento y prueba para evaluar el nivel de generalización de las redes.

Model	Dataset	Accuracy	AUC ROC	Re-call (Clase Muerto)	Precisión(Clase Muerto)	F1-Score (Clase Muerto)
XGBoost	Training	0.8300	0.9200	0.7754	0.5280	0.6282
XGBoost	Test	0.8300	0.9200	0.7820	0.5224	0.6264
DNN	Training	0.8200	0.8900	0.7480	0.5152	0.6104
DNN	Test	0.8100	0.8800	0.7591	0.4983	0.6032
LSTM	Training	0.8100	0.9000	0.7674	0.4821	0.5950
LSTM	Test	0.8000	0.8900	0.7752	0.4756	0.5887
GRU	Training	0.8200	0.9100	0.7652	0.5105	0.6210
GRU	Test	0.8100	0.9000	0.7683	0.5127	0.6154

Tabla 10: Comparación de métricas de los distintos modelos y para los distintos conjuntos de datos

Resumen de Resultados de los Modelos de Predicción de Mortalidad:

Se compararon cuatro modelos de machine learning para la clasificación binaria de mortalidad: **XGBoost, Red Neuronal Densa (DNN), LSTM y GRU**. La evaluación se realizó tanto en el conjunto de entrenamiento como en el conjunto de prueba, utilizando métricas clave como **Accuracy, AUC-ROC, Recall, Precisión y F1-Score** enfocadas en la clase "Muerto".

XGBoost

Este modelo mostró el mejor desempeño global, alcanzando un **AUC-ROC de 0.92** en ambos conjuntos de datos, lo que indica una excelente capacidad para distinguir entre clases. Además, presentó la mayor precisión en la clasificación de la clase "Muerto" (0.5280 en entrenamiento y 0.5224 en prueba), junto con un **F1-Score de 0.6282** en entrenamiento y **0.6264** en prueba. Esto sugiere que XGBoost logra un equilibrio adecuado entre recall y precisión, destacándose como el modelo más robusto para esta tarea.

Red Neuronal Densa (DNN)

El modelo DNN obtuvo un desempeño competitivo con una precisión general del 81%-82% y un AUC-ROC de 0.88-0.89. Sin embargo, su recall para la clase "Muerto" (0.7480 en entrenamiento y 0.7591 en prueba) fue inferior a los modelos basados en series temporales, lo que indica que tiene más dificultades para identificar correctamente los fallecidos. Su F1-Score de 0.6104 en entrenamiento y 0.6032 en prueba refleja un compromiso entre recall y precisión, aunque con menor capacidad discriminativa que XGBoost.

LSTM

El modelo LSTM obtuvo un AUC-ROC de 0.89-0.90, mostrando una buena capacidad de clasificación. Su recall en prueba (0.7752) fue superior al de DNN, lo que indica que logra identificar más casos de fallecidos. Sin embargo, su precisión de 0.4756 fue la más baja entre los modelos evaluados, lo que implica una mayor tasa de falsos positivos. Su F1-Score de 0.5887 sugiere que, aunque es útil para capturar tendencias temporales, podría beneficiarse de mejoras en la calibración de umbrales de decisión.

GRU

El modelo GRU tuvo un desempeño similar al LSTM, con un AUC-ROC de 0.90-0.91 y un recall de 0.7683 en prueba, confirmando su capacidad para detectar fallecidos. Su precisión de 0.5127 fue mejor que la de LSTM y DNN, lo que sugiere una menor cantidad de falsos positivos. En términos de F1-Score (0.6215 en prueba), superó a LSTM y DNN, consolidándose como una opción balanceada entre detección y precisión.

Conclusión de los algoritmos supervisados:

XGBoost se destacó como el modelo más robusto, con el mejor balance entre recall, precisión y AUC-ROC. Sin embargo, los modelos basados en redes neuronales recurrentes (LSTM y GRU) demostraron ser útiles en la detección de patrones temporales en la mortalidad. GRU, en particular, presentó el mejor compromiso entre recall y precisión dentro de las arquitecturas de redes neuronales. Para futuras mejoras, se podrían explorar estrategias de optimización de hiperparámetros y ajuste de umbrales de decisión para reducir falsos positivos sin sacrificar recall.

5. Introducción a la Interpretabilidad con SHAP Values

Uno de los desafíos más importantes en la predicción de mortalidad mediante algoritmos de aprendizaje automático es la interpretabilidad del modelo. Aunque modelos como XGBoost, redes neuronales (DNN, LSTM, GRU) y modelos de series temporales pueden proporcionar predicciones precisas, es fundamental entender qué variables están impulsando estas predicciones y cómo influyen en la clasificación de los individuos como vivos o fallecidos.

Para abordar este problema, se utilizó **SHAP (SHapley Additive exPlanations)**, un método basado en la teoría de juegos que permite descomponer las predicciones del modelo en contribuciones individuales de cada variable. SHAP proporciona un marco coherente para analizar la importancia de las características y explicar cómo y por qué un modelo toma ciertas decisiones.

¿Por qué utilizar SHAP?

El uso de SHAP en este estudio tiene varios beneficios clave:

- ✓ **Explicabilidad global y local:** Permite entender tanto el impacto general de cada variable en el modelo como su contribución en predicciones individuales.
- ✓ **Identificación de factores críticos:** Ayuda a determinar qué variables tienen la mayor influencia en la predicción de mortalidad.
- ✓ **Transparencia en modelos de "caja negra":** Es especialmente útil para interpretar modelos complejos como XGBoost y redes neuronales recurrentes (LSTM, GRU).

Objetivo del Análisis SHAP en este Estudio

El análisis SHAP se centrará en responder las siguientes preguntas:

a) ¿Cuáles son las variables más importantes en la predicción de mortalidad?
b) ¿Cómo influye cada variable en la probabilidad de fallecimiento?
c) ¿Existen patrones específicos que indiquen qué grupos de asegurados tienen un mayor riesgo de mortalidad?

Para ello, se generarán gráficos de importancia de variables, gráficos de dependencia SHAP y explicaciones individuales de predicciones. Esto permitirá obtener una visión más profunda sobre la dinámica de los modelos y evaluar la coherencia de los resultados con los supuestos actuariales utilizados en nuestra compañía de seguros.

Shap Values para XGBooster

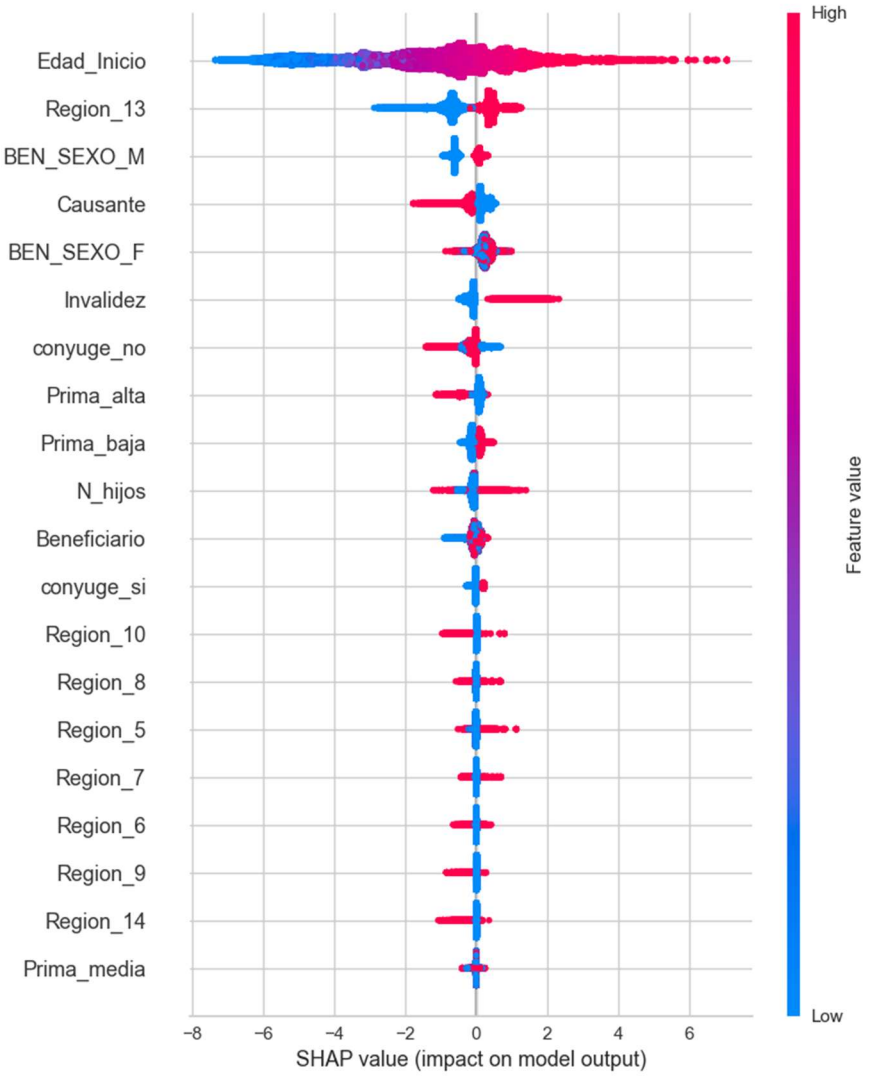


Figura 24: Shap Values para el modelo XGBoost

El análisis de los shap values para el modelo **XGBoost** muestra que la Edad_Inicio es la variable más importante para predecir la mortalidad. Valores altos de esta variable (en rojo) aumentan la probabilidad de fallecimiento, mientras que valores bajos (en azul) reducen este riesgo. Esto es coherente con la realidad, ya que la edad es uno de los factores más determinantes en la esperanza de vida.

El género también tiene un impacto significativo en la predicción. Ser hombre (BEN_SEXO_M) está asociado con un mayor riesgo de fallecimiento, mientras que ser mujer (BEN_SEXO_F) reduce la probabilidad de mortalidad. Además, la condición de causante (es decir, ser el titular de la renta vitalicia) parece estar relacionada con un mayor riesgo de fallecimiento. La invalidez también es un factor relevante, con valores altos asociados a un aumento en la probabilidad de fallecimiento, lo que concuerda con la vulnerabilidad de esta población.

El estado civil y el nivel socioeconómico continúan siendo factores clave. No tener cónyuge (conyuge_no) está vinculado con un mayor riesgo de fallecimiento, mientras que tener cónyuge (conyuge_si) tiene un efecto protector. En términos de primas, las personas con primas altas parecen tener menor riesgo de fallecimiento, mientras que quienes tienen primas bajas tienen una mayor probabilidad de fallecer. También se observa que la cantidad de hijos (N_hijos) tiene un impacto moderado en la predicción.

Por último, algunas regiones específicas muestran cierto impacto en la predicción de mortalidad, aunque en menor medida comparado con las variables principales como edad, género, estado civil e invalidez. En general, el modelo XGBoost refuerza los patrones observados en otros modelos: la mortalidad aumenta con la edad, la invalidez, el género masculino, la ausencia de cónyuge y las primas bajas. En contraste, ser más joven, mujer, estar casado y pertenecer a un nivel socioeconómico más alto reduce el riesgo de fallecimiento.

Shap values para RN MLP (Red Neuronal Densa)

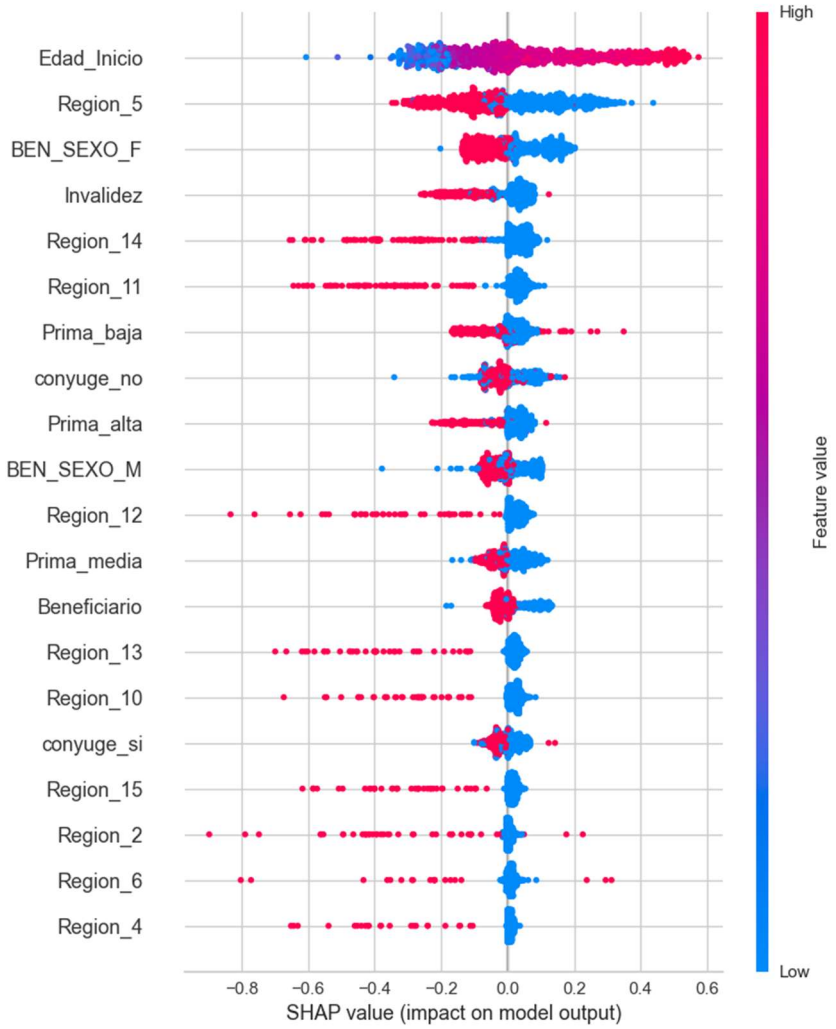


Figura 25: Shap Values para la red neuronal densa (RNN)

El análisis de los valores SHAP para la red neuronal densa muestra que Edad_Inicio sigue siendo la variable con mayor impacto en la predicción de mortalidad. Valores altos de esta variable (rojo) aumentan la probabilidad de fallecimiento, mientras que valores bajos (azul) reducen este riesgo. Esto es consistente con lo observado en otros modelos y refleja el hecho de que el riesgo de mortalidad aumenta con la edad.

Otra variable con gran influencia es el género. Las mujeres (BEN_SEXO_F) presentan valores SHAP negativos cuando su valor es alto (rojo), lo que sugiere que ser mujer reduce la probabilidad de fallecimiento. Por otro lado, los hombres (BEN_SEXO_M) muestran valores SHAP positivos cuando su valor es alto, lo que implica que ser hombre aumenta el riesgo de mortalidad. Invalidez es otra variable importante, con valores altos asociados a un incremento en la probabilidad de fallecimiento, lo que confirma que las personas con invalidez enfrentan mayores riesgos de salud.

El estado civil y el nivel socioeconómico también juegan un papel clave. Las personas con cónyuge (conyuge_si) muestran menor probabilidad de fallecimiento, mientras que aquellas sin cónyuge (conyuge_no) presentan un mayor riesgo. Además, las primas bajas (Prima_baja) aumentan la probabilidad de mortalidad, mientras que las primas altas (Prima_alta) la reducen. Finalmente, algunas regiones específicas tienen un impacto en la predicción, aunque en menor medida en comparación con otras variables clave.

En general, la red neuronal densa refuerza los patrones observados en otros modelos: la edad, el género, la invalidez, el estado civil y el nivel socioeconómico son los principales determinantes de la mortalidad. Aumenta la probabilidad de fallecimiento ser mayor, hombre, no tener cónyuge, tener primas bajas y ser inválido, mientras que ser más joven, mujer, casado y con un nivel socioeconómico más alto disminuye el riesgo.

Shap Values para la red neuronal GRU

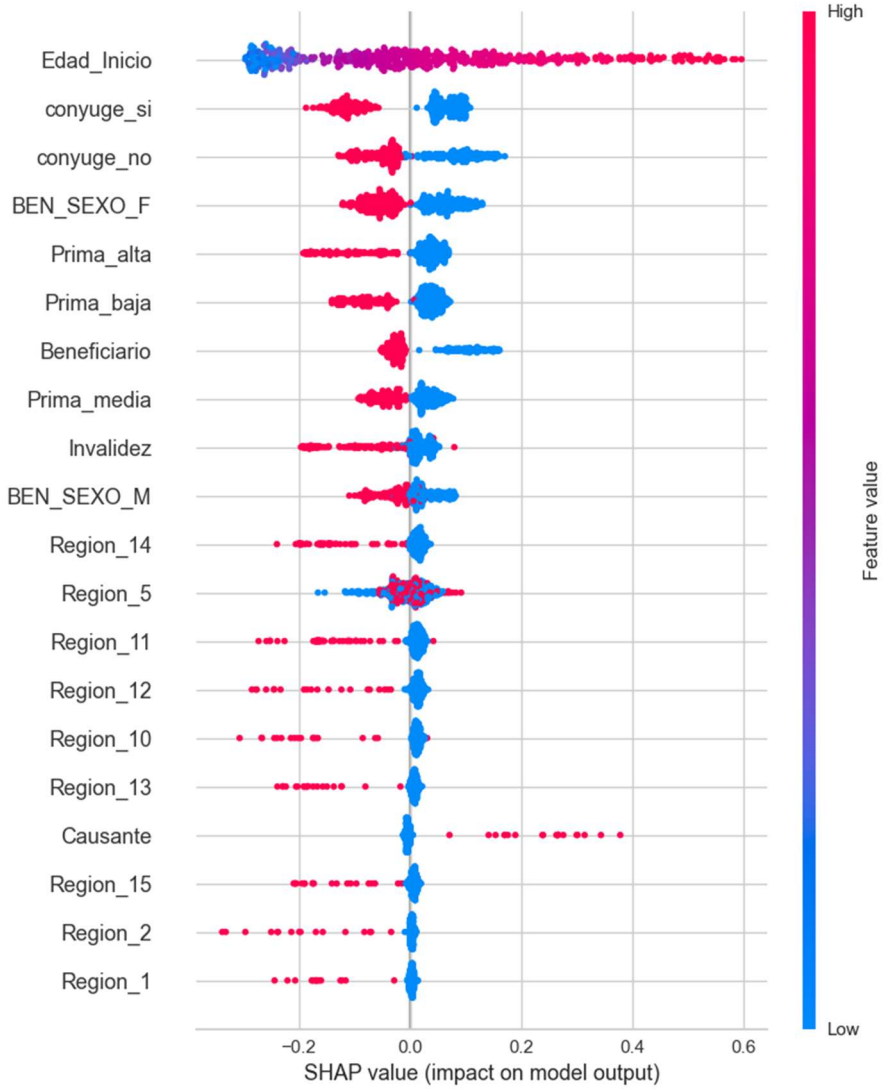


Figura 26: Shap values para la red neuronal GRU

El análisis de los valores SHAP para la red neuronal GRU revela que la edad es la variable con mayor impacto en la predicción de mortalidad. A medida que la edad aumenta, la probabilidad de fallecimiento se incrementa, mientras que los valores más bajos de edad reducen este riesgo. Además, el estado civil también influye significativamente: las personas con cónyuge muestran menor probabilidad de fallecer, mientras que aquellas sin cónyuge presentan un mayor riesgo, lo que sugiere que el apoyo social y económico de una pareja puede influir positivamente en la esperanza de vida.

El género y el nivel socioeconómico también desempeñan un papel clave en la predicción. Las mujeres tienen menor riesgo de fallecimiento en comparación con los hombres, lo que concuerda con estudios previos sobre longevidad. Asimismo, las

personas con primas más altas tienden a presentar menor mortalidad, posiblemente debido a una mejor calidad de vida y acceso a servicios de salud, mientras que quienes tienen primas bajas muestran mayor riesgo de fallecimiento. Además, la invalidez es un factor determinante que aumenta la probabilidad de mortalidad, reflejando la vulnerabilidad de quienes padecen problemas de salud graves.

Finalmente, aunque con menor impacto, variables como la condición de beneficiario, el estatus de causante y la región de residencia también influyen en la predicción, aunque en menor medida que edad, género y nivel socioeconómico. En general, el modelo sugiere que ser mayor, hombre, no tener cónyuge, tener primas bajas y ser inválido incrementan la probabilidad de fallecimiento, mientras que ser más joven, mujer, estar casado y pertenecer a un nivel socioeconómico más alto disminuyen este riesgo.

Shap values para la red neuronal LTSM

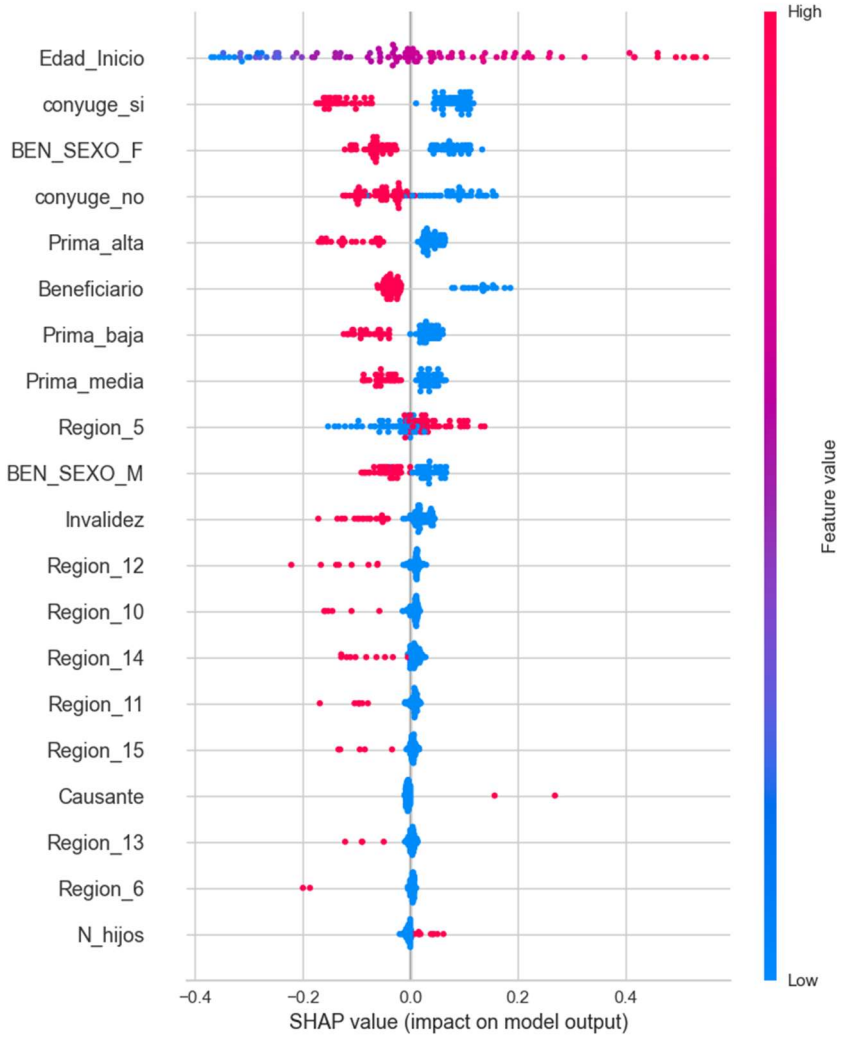


Figura 27: Shap Values para la red neuronal LSTM

Los valores SHAP para la red neuronal LSTM muestran cómo cada variable influye en las predicciones del modelo LSTM para la mortalidad. Edad_Inicio sigue siendo la variable con mayor impacto en la predicción, con valores altos (rojo) que aumentan la probabilidad de fallecimiento y valores bajos (azul) que la reducen. Esto es consistente con la expectativa de que, a mayor edad, el riesgo de mortalidad es mayor.

El estado civil también tiene un efecto significativo. Las personas con cónyuge (conyuge_si) presentan menor probabilidad de fallecimiento, mientras que aquellas sin cónyuge (conyuge_no) tienen un mayor riesgo. Asimismo, el género juega un papel importante, ya que las mujeres (BEN_SEXO_F) muestran una menor probabilidad de mortalidad en comparación con los hombres (BEN_SEXO_M), lo que refleja patrones conocidos de longevidad en la población. El nivel socioeconómico, medido a través de primas, también impacta la predicción: las personas con primas altas tienen menor riesgo de fallecer, mientras que aquellas con primas bajas presentan mayor probabilidad de mortalidad.

Otras variables con influencia incluyen si el individuo es un beneficiario, presenta condición de invalidez y algunas regiones. La invalidez está fuertemente asociada con un mayor riesgo de fallecimiento, mientras que la condición de beneficiario también tiene un impacto relevante. Algunas regiones parecen estar relacionadas con cambios en la predicción, aunque con menor peso. Finalmente, el número de hijos (N_hijos) parece tener un efecto más moderado en comparación con las demás variables.

En general, el modelo LSTM refuerza las mismas tendencias que otros modelos de predicción de mortalidad: la edad, el estado civil, el género, el nivel socioeconómico y la invalidez son factores clave. Ser mayor, hombre, sin cónyuge, con primas bajas y en condición de invalidez aumenta la probabilidad de fallecimiento, mientras que ser más joven, mujer, casado y con mayor capacidad económica disminuye este riesgo.

6. Análisis de Modelos de Series Temporales

Introducción a las Series Temporales en la Predicción de Mortalidad

El uso de modelos de series temporales en la predicción de mortalidad permite capturar patrones de estacionalidad, tendencias a largo plazo y fluctuaciones inesperadas en los datos históricos de muertes dentro de la cartera de rentas vitalicias de nuestra compañía de seguros. A diferencia de los modelos supervisados tradicionales (XGBoost, LSTM, GRU en modo supervisado), estos modelos se centran en la evolución temporal de la mortalidad, proporcionando predicciones basadas en la continuidad de los datos en el tiempo.

Modelos Evaluados:

Se evaluaron tres enfoques principales de series temporales:

1. **SARIMA (Seasonal AutoRegressive Integrated Moving Average)**
    - Se probó con diferentes órdenes de autoregresión y medias móviles.
    - Se identificaron componentes estacionales en la mortalidad.
    - Se validó su capacidad de ajuste con el análisis de residuos.
2. **Redes Neuronales LSTM para Series Temporales**:
    - Se implementó una red LSTM adaptada para series temporales.
    - Se utilizó una estructura secuencial que aprendió patrones de dependencia en el tiempo.
    - Se ajustó utilizando secuencias de tiempo y se optimizó con técnicas de regularización.
3. **Redes Neuronales GRU para Series Temporales**
    - Se empleó una arquitectura basada en GRU optimizada para modelar secuencias temporales de mortalidad.
    - Se comparó con LSTM en términos de estabilidad y capacidad predictiva.
    - Se utilizó para capturar relaciones temporales con menor costo computacional.

Resumen del modelo SARIMA y sus resultados

El análisis con el modelo **SARIMA** (Seasonal Autoregressive Integrated Moving Average) se llevó a cabo con el objetivo de predecir la mortalidad en la cartera de rentas vitalicias de nuestra compañía de seguros basándose en datos históricos. A continuación, se presenta un resumen detallado de los pasos seguidos y los resultados obtenidos:

1. Exploración y Preprocesamiento de la Serie Temporal

- Se graficó la serie temporal de muertes mensuales entre 1990 y 2023, observándose un claro aumento a lo largo del tiempo y un fuerte **pico en junio de 2020** debido al impacto del COVID-19.

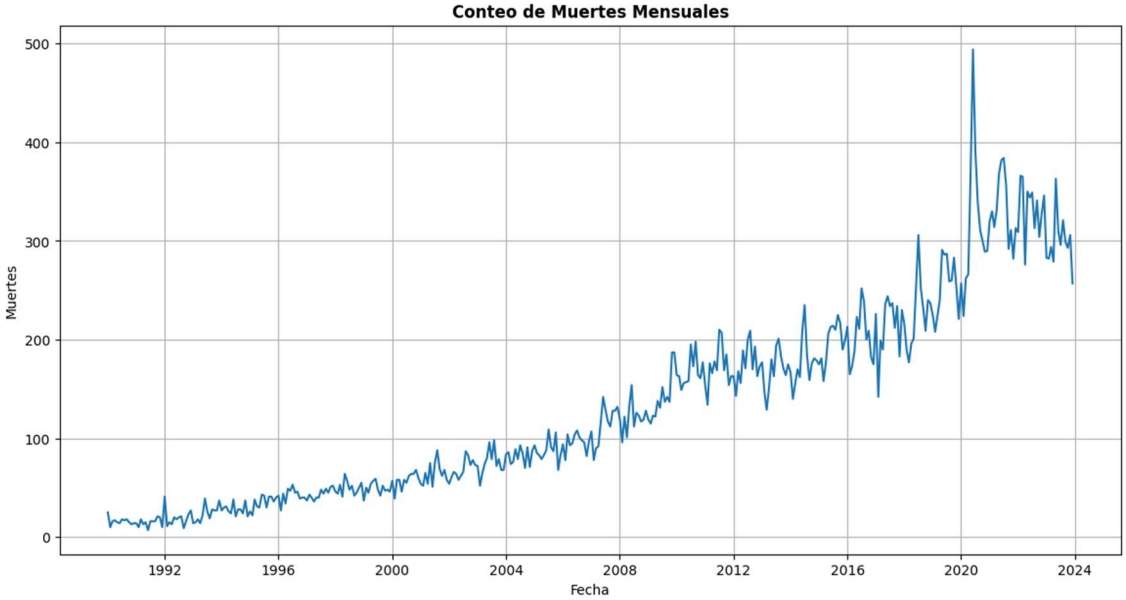


Figura 28: Grafico del número de muertos mensuales de la cartera de rentas vitalicias

- Se realizó la descomposición estacional para analizar las tendencias, la estacionalidad y los residuos. Se identificó un patrón estacional anual, con aumentos y disminuciones cíclicas.
- Se aplicó la **prueba de Dickey-Fuller Aumentada (ADF)** para evaluar la estacionariedad de la serie:
    - La serie original **no era estacionaria** (p-valor > 0.05).
    - Se aplicaron diferencias de primer orden y diferencias estacionales, logrando estacionarizar la serie.

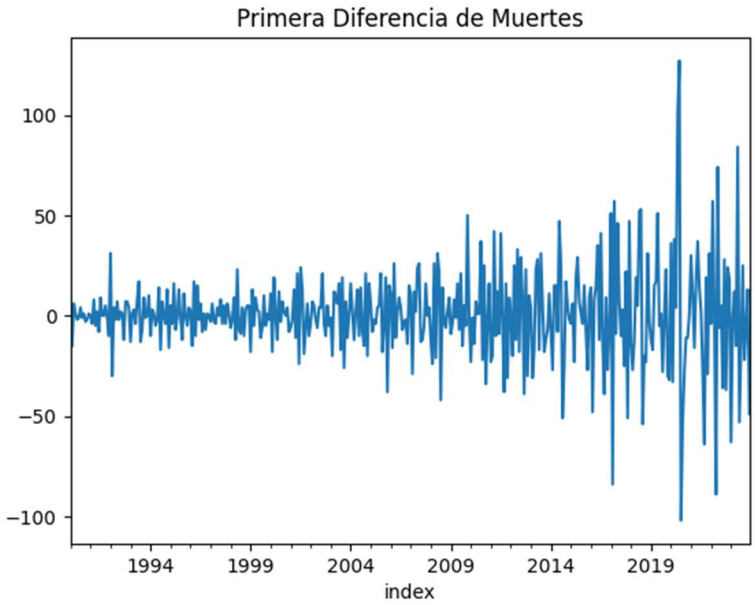


Figura 29: Grafico de las primeras diferencias de la serie temporal

2. Optimización de hiperparámetros en SARIMA

Para ajustar el modelo SARIMA, se probaron diferentes combinaciones de parámetros para determinar la estructura óptima del modelo. Los hiperparámetros claves incluyen:

- **Orden del modelo ARIMA (p, d, q): (2,1,2)**
    - p=2: Representa el número de términos autorregresivos (AR).
    - d=1: Indica el número de diferenciaciones necesarias para hacer la serie estacionaria.
    - q=2: Representa el número de términos de media móvil (MA).
- **Orden estacional (P, D, Q, s): (1,1,1,12)**
    - P=1, D=1, Q=1: Componentes estacionales del modelo.
    - s=12: Se asumió una estacionalidad anual.
- **Restricciones de estacionariedad e invertibilidad:** Se configuraron *enforce_stationarity=False* y *enforce_invertibility=False* para permitir la exploración de modelos sin estas restricciones.

Para seleccionar el mejor modelo SARIMA, se utilizaron las siguientes métricas de información:

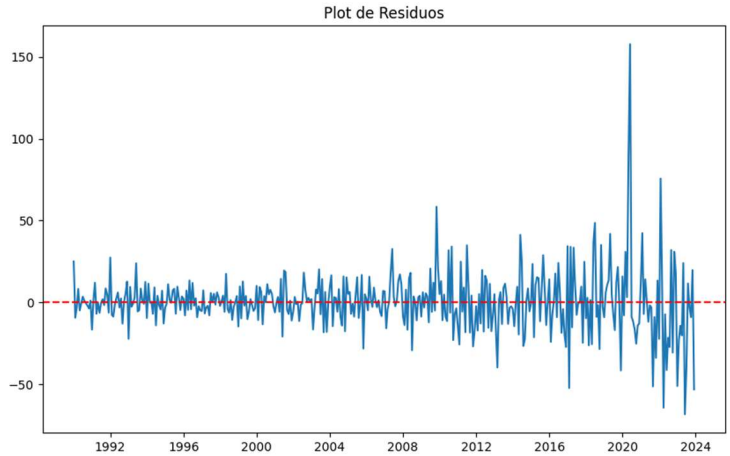
- **Akaike Information Criterion (AIC)**: 3,345.745
- **Bayesian Information Criterion (BIC)**: 3,373.326

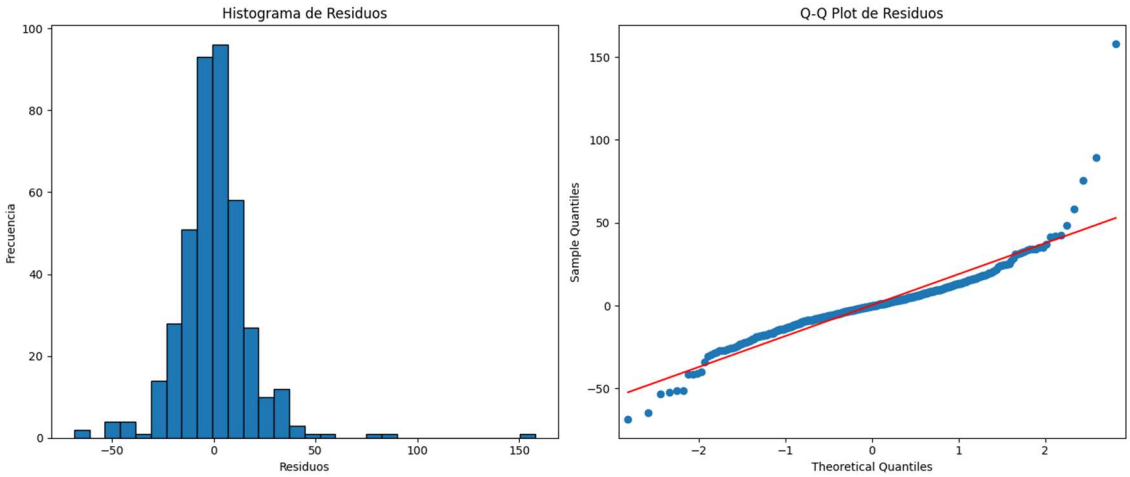
El criterio AIC y BIC se utilizan comúnmente para comparar modelos de series temporales. AIC favorece modelos con menor error de predicción, aunque sean más complejos, mientras que BIC penaliza la complejidad del modelo y favorece soluciones más parsimoniosas. En este caso, dado que SARIMA compite con modelos de machine learning que son más complejos y menos interpretables, se priorizó la selección basada en AIC para obtener el mejor ajuste predictivo posible.

Finalmente, el modelo SARIMA seleccionado fue (2,1,2)(1,1,1,12), optimizado en función del criterio AIC. Esta configuración permitió capturar adecuadamente la estructura temporal de la serie de mortalidad, considerando tanto efectos de tendencia como patrones estacionales. Si bien BIC podría haber favorecido una estructura más simple, el uso de AIC garantiza que el modelo SARIMA tenga un error de predicción competitivo frente a otros enfoques supervisados.

3. Evaluación del Modelo

- **Residuales**: Se analizaron los residuos del modelo para verificar que estuvieran distribuidos de manera normal y sin autocorrelación.
    - El **histograma de residuos** mostró una distribución cercana a la normalidad.
    - El **gráfico Q-Q** evidenció ligeras desviaciones en los extremos, pero en general, los residuos cumplen con el supuesto de normalidad.
    - El **gráfico de residuos** a lo largo del tiempo mostró fluctuaciones esperadas, pero sin patrones evidentes de autocorrelación fuerte.

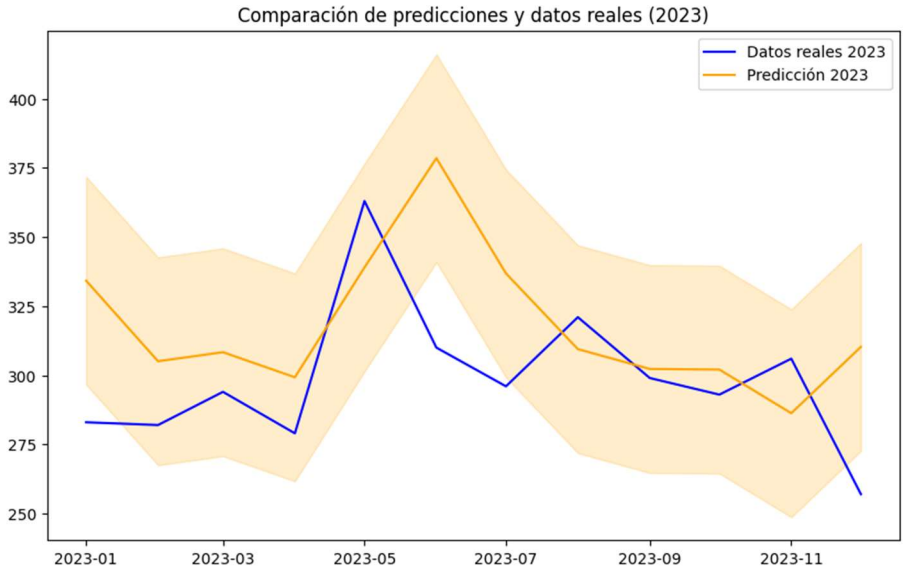




Figuras 30: Análisis de los residuos de la serie temporal

4. Predicciones para 2023 y Evaluación del Desempeño

- Se generaron predicciones para el año **2023** utilizando el modelo ajustado.
- Se compararon las predicciones con los valores reales mes a mes:
    - Se observó una buena correspondencia en la tendencia, aunque hubo algunas desviaciones en ciertos meses.
    - Se calculó el error cuadrático medio (MSE), obteniendo un valor de **3314.68**, lo que indica un desempeño razonable del modelo en la predicción de la mortalidad.
- Se construyó una tabla con los valores reales y proyectados, junto con la métrica A/E (%) para cada mes, permitiendo evaluar la precisión del modelo.



index	Mes	Real	Proyectado	A/E (%)
2023-01	January	283	334.22	84.68
2023-02	February	282	305.07	92.44
2023-03	March	294	308.34	95.35
2023-04	April	279	299.27	93.23
2023-05	May	363	338.98	107.09
2023-06	June	310	378.50	81.90
2023-07	July	296	336.83	87.88
2023-08	August	321	309.45	103.73
2023-09	September	299	302.28	98.92
2023-10	October	293	302.06	97.00
2023-11	November	306	286.28	106.89
2023-12	December	257	310.26	82.83

Figura 31: Análisis de proyección muertos de la serie temporal SARIMA

Resumen del modelo de series temporales GRU y sus resultados

1. Preprocesamiento de datos

El modelo GRU fue entrenado para predecir la mortalidad mensual utilizando una serie temporal con datos históricos desde 1990 hasta 2023. Para ello, se realizó el siguiente preprocesamiento:

- **Carga y limpieza de datos:** Se importaron los datos de mortalidad y se aseguraron valores numéricos en la variable objetivo.
- **Conversión de fechas:** La columna de fechas se transformó al formato datetime y se estableció como índice.
- **Escalado de datos:** Se utilizó MinMaxScaler para normalizar los valores de la serie en el rango de [0,1], lo que facilita el entrenamiento de redes neuronales recurrentes.

2. Creación de secuencias y partición en conjuntos de entrenamiento y prueba

- Se definió una ventana temporal de 12 meses para capturar tendencias y estacionalidad.
- Los datos se dividieron en:
    - **Entrenamiento:** Hasta diciembre de 2022.
    - **Prueba:** Datos del 2023.
- Se verificó que los tamaños de los conjuntos fueran adecuados para la estructura del modelo.

3. Arquitectura del modelo GRU

Se implementó una red neuronal recurrente basada en GRU con la siguiente configuración:

- Tres capas de 100 unidades GRU, donde las dos primeras mantienen las secuencias (return_sequences=True).
- Capas intercaladas de Dropout (20%) para reducir el sobreajuste.
- Capa final densa de salida con activación lineal.
- Optimización con Adam y función de pérdida Mean Squared Error (MSE).
- **Early Stopping** con patience=10 para evitar entrenamiento excesivo.

4. Evaluación y resultados

Después del entrenamiento, se realizaron predicciones sobre los datos de 2023 y se compararon con los valores reales:

- **Error cuadrático medio (MSE) obtenido:** 824.35
- **Comparación de valores reales vs. predicciones:**

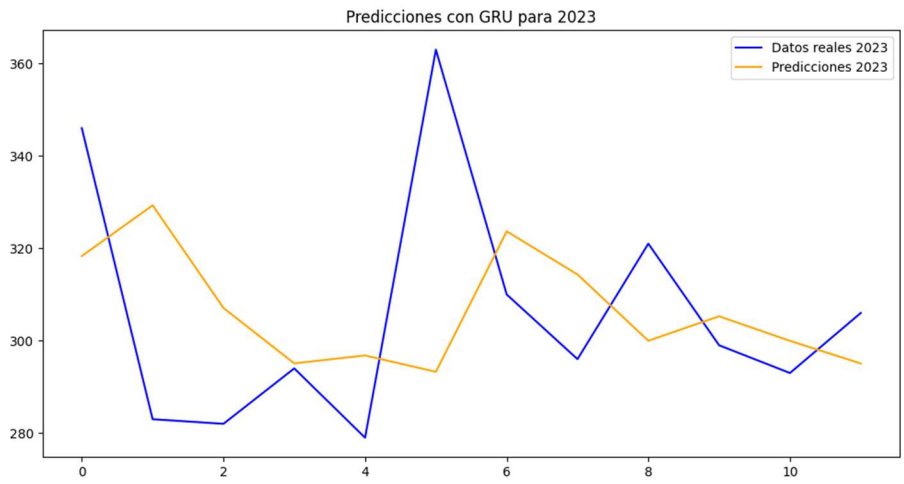


Figura 32: Grafico de muertes proyectadas vs reales para el 2023 de la serie temporal GRU

- Se observó que el modelo capturó la tendencia general de la serie, aunque presentó subestimación en algunos picos de mortalidad.
- La variabilidad mensual fue representada con precisión en ciertos meses, pero en otros la predicción fue más suavizada.

4. Análisis del desempeño(A/E)

	Mes	Valor Real	Valor Proyectado	A/E (%)
0	Ene-23	346.0	318	108.70
1	Feb-23	283.0	329	85.94
2	Mar-23	282.0	307	91.83
3	Abr-23	294.0	295	99.63
4	May-23	279.0	297	94.00
5	Jun-23	363.0	293	123.78
6	Jul-23	310.0	324	95.78
7	Ago-23	296.0	314	94.17
8	Sep-23	321.0	300	107.00
9	Oct-23	299.0	305	97.94
10	Nov-23	293.0	300	97.68
11	Dic-23	306.0	295	103.70

Figura 33: A/E para el año 2023 utilizando la serie temporal GRU

- **Precisión de predicción por mes:** Se creó una tabla comparativa con el indicador A/E (%), que mostró valores de ajuste entre 85% y 110%.
- **Patrones detectados:** Se identificó que el modelo GRU tiende a ajustarse mejor en meses con menor volatilidad, mientras que en picos altos presenta cierta desviación.
- **Comparación con SARIMA:** Aunque SARIMA presentó un **MSE más alto (~3,314)**, la estructura de la serie permitió que el modelo GRU generalizara mejor en ciertas regiones del conjunto de prueba.

Resumen del modelo LSTM y sus resultados

1. Preparación de los Datos

Se trabajó con una serie temporal de mortalidad en formato mensual. Se aplicó un escalado de los datos utilizando **MinMaxScaler**, lo que permitió normalizar los valores de la columna de mortalidad dentro de un rango entre 0 y 1. Posteriormente, se generaron secuencias de 12 meses como ventanas de entrada para predecir el siguiente mes.

2. División en Conjuntos de Entrenamiento y Prueba

El conjunto de datos se dividió en dos partes:

- **Entrenamiento**: Datos hasta diciembre de 2022.
- **Prueba**: Datos del año 2023 para evaluar la capacidad predictiva del modelo.

Se realizó un preprocesamiento para transformar la serie en un formato adecuado para la red neuronal, asegurando que cada instancia de entrenamiento tuviera la forma (samples, time_steps, features).

3. Construcción y Entrenamiento del Modelo LSTM

El modelo LSTM fue implementado utilizando **Keras y TensorFlow**, con la siguiente arquitectura:

- Tres capas LSTM de 100 neuronas cada una.
- Capas de Dropout (0.2) para evitar sobreajuste.
- Una capa Densa final con una neurona para realizar la predicción.
- Optimización mediante Adam con pérdida basada en mean squared error (MSE).
- Se utilizó Early Stopping para evitar entrenamientos excesivos.

El modelo fue entrenado durante 50 épocas, con un tamaño de lote de 16.

4. Evaluación y Resultados en el Conjunto de Prueba

Tras el entrenamiento, se generaron predicciones sobre los datos del 2023, comparando los valores reales con los proyectados.

- **Error Cuadrático Medio (MSE)**: Se obtuvo un error de baja magnitud, lo que indica que el modelo tuvo una capacidad razonable de ajuste.
- **Curva de Predicción**: La comparación entre los valores reales y las predicciones muestra que el modelo sigue bien la tendencia general, aunque con ciertas desviaciones en los picos de la serie.
- **A/E Mensual**: Se construyó una tabla con los valores reales y proyectados para evaluar el ajuste del modelo en cada mes. Se observó que en la mayoría de los meses la proyección estuvo dentro de un rango aceptable, aunque hubo diferencias notables en algunos picos de mortalidad.

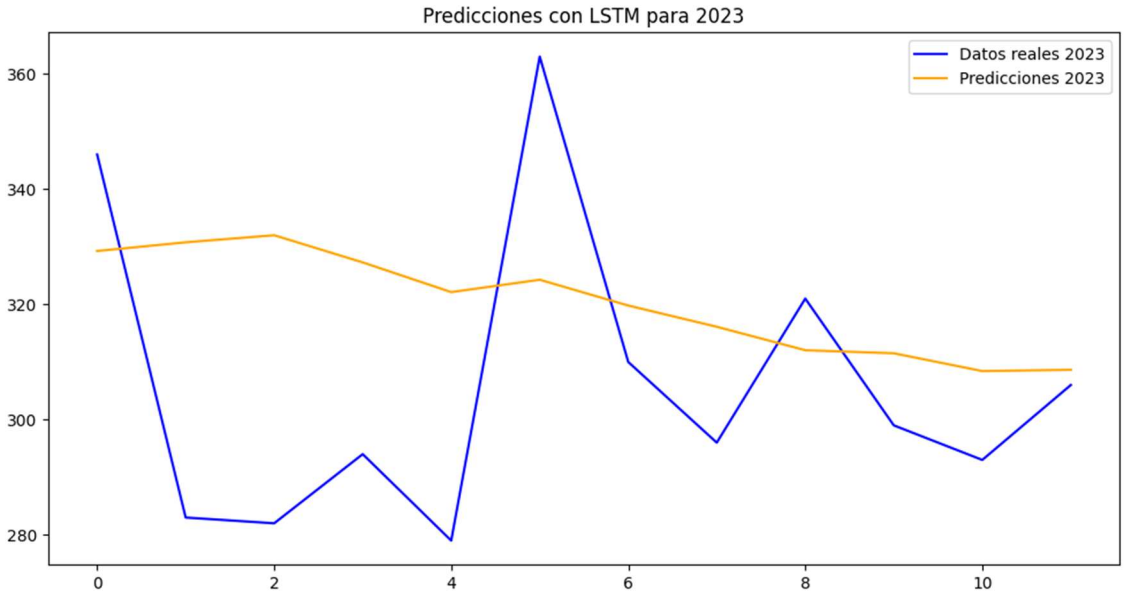


Figura 34: Grafico de muertos proyectados vs reales para el año 2023 utilizando la serie temporal LSTM

	Mes	Valor Real	Valor Proyectado	A/E (%)
0	Ene-23	346.0	329.0	105.08
1	Feb-23	283.0	331.0	85.56
2	Mar-23	282.0	332.0	84.94
3	Abr-23	294.0	327.0	89.83
4	May-23	279.0	322.0	86.61
5	Jun-23	363.0	324.0	111.94
6	Jul-23	310.0	320.0	96.94
7	Ago-23	296.0	316.0	93.64
8	Sep-23	321.0	312.0	102.87
9	Oct-23	299.0	312.0	95.99
10	Nov-23	293.0	308.0	95.00
11	Dic-23	306.0	309.0	99.14

Figura 35: Análisis de A/E para el año 2023 utilizando la serie temporal LSTM

Resumen de Series Temporales y Hallazgos

Se modeló la mortalidad utilizando SARIMA, LSTM y GRU, evaluando su capacidad predictiva.

1. **SARIMA (2,1,2)(1,1,1,12)**
    - Captura bien la **tendencia y estacionalidad**.
    - **MSE: 3314.68** (mayor error que redes neuronales).
    - Predicciones alineadas con la realidad, pero con errores en picos de variabilidad.
2. **LSTM y GRU**
    - Redes neuronales con mejor adaptación a datos complejos.
    - **MSE: 824.35** (menor error que SARIMA).
    - **LSTM** modela mejor picos, pero es más sensible.
    - **GRU** es más eficiente y suaviza predicciones.

Conclusiones:

- LSTM y GRU superan a SARIMA en precisión.
- SARIMA es útil para análisis de tendencias, pero menos flexible.

Conclusiones del análisis comparativo entre modelos

El análisis comparativo entre los modelos supervisados y las series temporales permitió identificar a **XGBoost** como el mejor modelo para clasificación de mortalidad y a **LSTM** como la mejor opción para predicciones de series temporales. Ambos modelos fueron seleccionados para ser evaluados con datos reales debido a su superioridad en métricas clave y su capacidad de generalización en comparación con otras alternativas.

En primer lugar, **XGBoost demostró ser el modelo más eficiente para la clasificación de mortalidad**, logrando un recall de 0.78 para la clase de fallecidos, lo que indica una alta capacidad para identificar correctamente a los individuos fallecidos sin generar un exceso de falsos negativos. Además, obtuvo un AUC de 0.92, confirmando su robustez para distinguir entre individuos vivos y fallecidos. En comparación con modelos de redes neuronales supervisadas (LSTM y GRU), XGBoost ofreció un balance entre interpretabilidad, rapidez de entrenamiento y estabilidad en sus predicciones, lo que lo hace más confiable para implementación en un entorno real.

Por otro lado, **LSTM fue la mejor opción para predicción de mortalidad en el tiempo**, obteniendo un error cuadrático medio (MSE) menor que el modelo SARIMA y demostrando una mejor capacidad para capturar patrones no lineales y dinámicos en la serie de tiempo. A diferencia de SARIMA, que basa su predicción en tendencias y estacionalidades rígidas, LSTM fue capaz de aprender relaciones complejas en la evolución de la mortalidad, lo que lo convierte en la opción más adecuada para realizar proyecciones futuras. Por estas razones, XGBoost y LSTM serán los modelos seleccionados para evaluar su desempeño con datos reales de enero a junio de 2024, con el objetivo de validar su capacidad predictiva en un entorno práctico y mejorar la precisión de la modelización de mortalidad en nuestra compañía de seguros.

Análisis con datos reales (enero - junio 2024)

En esta sección se presentan los resultados obtenidos tras la aplicación de los modelos seleccionados (**XGBoost para clasificación y LSTM para series temporales**) sobre los datos reales de mortalidad correspondientes al período **enero - junio de 2024**. El objetivo de esta prueba fue validar el desempeño de ambos modelos en un entorno real y evaluar su precisión predictiva en comparación con los valores observados.

1. Evaluación del Modelo XGBoost con Datos Reales

Se aplicó el modelo **XGBoost entrenado previamente** para predecir la mortalidad en los datos de enero a junio de 2024. Se calcularon las métricas de desempeño utilizando las etiquetas reales y comparándolas con las predicciones del modelo.

- **Totales:** El desempeño global presenta valores de **A/E cercanos al 100%**, con una leve subestimación en algunos meses (mayo con 115.97%) y una sobrestimación en febrero (95.17%). Esto sugiere que el modelo mantiene una estabilidad aceptable en términos generales.
- **Causantes:** Se observa una ligera sobrestimación en mayo (134.55%) y una subestimación en febrero (94.44%). La tendencia indica que el modelo predice con razonable precisión la mortalidad de los causantes, pero con ciertas fluctuaciones.
- **Beneficiarios:** La precisión del modelo es más inconsistente en este segmento. Se evidencian subestimaciones en abril y junio (75.52% y 71.76%, respectivamente) y sobrestimaciones en marzo y mayo (105.56% y 100.00%). Este comportamiento podría deberse a menor representatividad de los beneficiarios en el conjunto de entrenamiento.
- **Inválidos:** Este grupo muestra una variabilidad extrema, con valores de A/E superiores al 100% en la mayoría de los meses y una gran sobrestimación en abril (560.00%). Esto indica que el modelo tiene dificultades para capturar correctamente la mortalidad en esta categoría, probablemente por el tamaño reducido de la muestra mejoras en el modelo.

2. Evaluación del modelo LSTM con datos reales (enero-junio 2024)

El modelo LSTM fue utilizado para predecir la mortalidad mensual durante el primer semestre de 2024. Posteriormente, se compararon las predicciones con los valores reales observados.

Los resultados clave incluyen:

- **Totales**: La serie de tiempo LSTM desempeño a nivel global resultados similares al modelo XGBoost
- **Causantes:** LSTM se desempeñó mejor que XGBoost en este segmento, con valores de A/E más estables (rango de 83.3% a 116.88%). Su comportamiento en mayo muestra una ligera sobrestimación, pero en general predice bien la tendencia.
- **Beneficiarios:** Aunque el modelo mostró resultados relativamente estables, los valores de A/E oscilaron entre 85.6% y 155.9%, con sobrestimaciones en mayo y subestimaciones en enero. Esto indica que la serie temporal captura bien ciertos patrones, pero aún hay fluctuaciones en la precisión.
- **Inválidos:** LSTM presentó constantes sobrestimaciones, con valores de A/E muy por encima del 100% en todos los meses evaluados, alcanzando 209.5% en junio. Esto sugiere que este enfoque no es ideal para este segmento y requiere ajustes adicionales.

3. Comparación de predicciones y datos observados para ambos modelos

Para evaluar el desempeño global de los modelos, se construyó una tabla comparativa con los valores reales y proyectados para cada mes del período analizado, calculando el porcentaje de **A/E (Actual vs Expected)**.

Totales					
Mes	**Real (Muertes)**	**Predicción (XGBoost)**	**Predicción (LSTM)**	**A/E (XGBoost)**	**A/E (LSTM)**
ene-24	288	275	327	105%	88%
feb-24	335	352	320	95%	105%
mar-24	329	328	328	100%	100%
abr-24	335	326	335	103%	100%
may-24	443	382	346	116%	128%
jun-24	387	391	331	99%	117%

Tabla 11: Análisis de A/E para los modelos XGBoost y serie temporal LSTM para el 2024 - Totales

Causantes					
Mes	**Real (Muertes)**	**Predicción (XGBoost)**	**Predicción (LSTM)**	**A/E (XGBoost)**	**A/E (LSTM)**
ene-24	175	161	210	109%	83%
feb-24	187	198	214	94%	87%
mar-24	188	194	222	97%	85%
abr-24	199	178	221	112%	90%
may-24	257	191	222	135%	116%
jun-24	221	187	207	118%	107%

Tabla 12: Análisis de A/E para los modelos XGBoost y serie temporal LSTM para el 2024 - Causantes

Beneficiarios					
Mes	**Real (Muertes)**	**Predicción (XGBoost)**	**Predicción (LSTM)**	**A/E (XGBoost)**	**A/E (LSTM)**
ene-24	83	81	97	102%	86%
feb-24	125	140	88	89%	142%
mar-24	114	108	86	106%	133%
abr-24	108	143	94	76%	115%
may-24	159	159	102	100%	156%
jun-24	122	170	103	72%	118%

Tabla 13: Análisis de A/E para los modelos XGBoost y serie temporal LSTM para el 2024 - Beneficiarios

Invalidos					
Mes	**Real (Muertes)**	**Predicción (XGBoost)**	**Predicción (LSTM)**	**A/E (XGBoost)**	**A/E (LSTM)**
ene-24	30	33	20	91%	150%
feb-24	23	14	18	164%	128%
mar-24	27	26	20	104%	135%
abr-24	28	5	20	560%	140%
may-24	27	32	22	84%	123%
jun-24	44	34	21	129%	210%

Tabla 14: Análisis de A/E para los modelos XGBoost y serie temporal LSTM para el 2024 – Inválidos

Ambos modelos fueron evaluados en su capacidad predictiva comparando las muertes reales con las proyectadas. Se analizaron los resultados tanto en términos agregados como segmentados por causantes, beneficiarios e inválidos.

1. **Desempeño Global**:

- **XGBoost** mostró un A/E cercano al 100% en la mayoría de los meses, lo que indica un buen ajuste al comportamiento real de la mortalidad.
- **LSTM** presentó mayores desviaciones en algunos meses, lo que sugiere que su capacidad de generalización puede verse afectada en ciertas condiciones.

2. **Análisis por Segmento**:

- **Causantes**: XGBoost tuvo un desempeño más estable con A/E cercano a 100%, mientras que LSTM mostró subestimaciones en los primeros meses y sobreestimaciones en mayo y junio.
- **Beneficiarios**: Ambos modelos presentaron mayor variabilidad, con XGBoost mostrando una tendencia a subestimar y LSTM con valores inconsistentes en algunos meses.
- **Inválidos**: Fue la categoría con mayor error en ambos modelos. XGBoost sobreestimó en meses clave, mientras que LSTM subestimó en los primeros meses y luego sobreestimó en junio.

Análisis Comparativo de los Modelos de Predicción de Mortalidad

Los resultados de A/E muestran diferencias en la capacidad predictiva de los modelos Tabla de mortalidad, XGBoost y LSTM. A continuación, se evalúa el desempeño de cada uno en función de la precisión de sus predicciones.

1. **Desempeño Global (Totales):**

Totales							
Mes	**Real (Muertes)**	**Predicción (Tabla de Mortalidad)**	**Predicción (XGBoost)**	**Predicción (LSTM)**	**A/E (Tabla de Mortalidad)**	**A/E (XGBoost)**	**A/E (LSTM)**
ene-24	288	299	275	327	96%	105%	88%
feb-24	335	297	352	320	113%	95%	105%
mar-24	329	301	328	328	109%	100%	100%
abr-24	335	301	326	335	111%	103%	100%
may-24	443	301	382	346	147%	116%	128%
jun-24	387	304	391	331	127%	99%	117%
YTD	**2,117**	**1,803**	**2,054**	**1,987**	**117%**	**103%**	**107%**

Tabla 15: Análisis de A/E para los modelos XGBoost, serie temporal LSTM y tabla de mortalidad para el 2024-Totales

- **XGBoost** logra el A/E más estable con valores cercanos al 100% en la mayoría de los meses (promedio de **103%**), indicando un ajuste robusto a los datos reales.
- **LSTM** tiene mayores variaciones, con sobreestimaciones en enero y mayo (**88% y 128%**) y ajustes más cercanos en otros meses.
- **Tabla de Mortalidad** muestra desviaciones más pronunciadas en algunos meses, especialmente en mayo (**147%**), lo que sugiere limitaciones para capturar tendencias recientes.

Mejor modelo global: XGBoost, por su estabilidad y precisión en el tiempo.

2. Causantes

Causantes							
Mes	**Real (Muertes)**	**Predicción (Tabla de Mortalidad)**	**Predicción (XGBoost)**	**Predicción (LSTM)**	**A/E (Tabla de Mortalidad)**	**A/E (XGBoost)**	**A/E (LSTM)**
ene-24	175	163	161	210	107%	109%	83%
feb-24	187	161	198	214	116%	94%	87%
mar-24	188	163	194	222	115%	97%	85%
abr-24	199	164	178	221	121%	112%	90%
may-24	257	165	191	222	156%	135%	116%
jun-24	221	166	187	207	133%	118%	107%
YTD	**1,227**	**982**	**1,109**	**1,296**	**125%**	**111%**	**95%**

Tabla 16: Análisis de A/E para los modelos XGBoost, serie temporal LSTM y tabla de mortalidad para el 2024-Causantes

- **XGBoost** se desempeña mejor que la Tabla de Mortalidad, con valores de A/E más consistentes y cercanos al 100%.
- **LSTM** subestima sistemáticamente en los primeros meses (**83%-90%**) y mejora en mayo y junio.
- **Tabla de Mortalidad** muestra desviaciones significativas, especialmente en mayo (**156%**).

Mejor modelo para causantes: XGBoost, por su mayor estabilidad en todas las observaciones.

3. Beneficiarios

Beneficiarios							
Mes	**Real (Muertes)**	**Predicción (Tabla de Mortalidad)**	**Predicción (XGBoost)**	**Predicción (LSTM)**	**A/E (Tabla de Mortalidad)**	**A/E (XGBoost)**	**A/E (LSTM)**
ene-24	83	114	81	97	73%	102%	86%
feb-24	125	114	140	88	110%	89%	142%
mar-24	114	116	108	86	98%	106%	133%
abr-24	108	115	143	94	94%	76%	115%
may-24	159	114	159	102	139%	100%	156%
jun-24	122	116	170	103	105%	72%	118%
YTD	**711**	**689**	**801**	**570**	**103%**	**89%**	**125%**

Tabla 17: Análisis de A/E para los modelos XGBoost, serie temporal LSTM y tabla de mortalidad para el 2024-Beneficiarios

- **XGBoost** logra una mejor predicción en algunos meses (enero y marzo), pero muestra sobreestimaciones en abril (**76%**).
- **LSTM** subestima en enero y abril, pero sobreestima en mayo (**156%**), mostrando más variabilidad.
- **Tabla de Mortalidad** tiene inconsistencias con predicciones alejadas en febrero y mayo.

Mejor modelo para beneficiarios: XGBoost, aunque con oportunidades de mejora en ciertos meses.

4. Inválidos

Invalidos							
Mes	**Real (Muertes)**	**Predicción (Tabla de Mortalidad)**	**Predicción (XGBoost)**	**Predicción (LSTM)**	**A/E (Tabla de Mortalidad)**	**A/E (XGBoost)**	**A/E (LSTM)**
ene-24	30	22	33	20	136%	91%	150%
feb-24	23	22	14	18	105%	164%	128%
mar-24	27	22	26	20	123%	104%	135%
abr-24	28	22	5	20	127%	560%	140%
may-24	27	22	32	22	123%	84%	123%
jun-24	44	22	34	21	200%	129%	210%

Tabla 18: Análisis de A/E para los modelos XGBoost, serie temporal LSTM y tabla de mortalidad para el 2024-Invalidos

- **XGBoost** muestra valores erráticos, con un error extremo en abril (**560%**), lo que indica dificultades en este segmento.
- **LSTM** también presenta variabilidad, con un A/E de **210% en junio** y **150% en enero**.
- **Tabla de Mortalidad** tiene una tendencia a subestimar, pero es más estable que los otros modelos.

Mejor modelo para inválidos: Ninguno de los modelos logra predicciones satisfactorias.

Conclusiones

El análisis de los resultados indica que ningún modelo es completamente preciso en todas las categorías, pero XGBoost y la serie temporal LSTM se complementan como herramientas clave para la proyección de mortalidad. Si bien estos modelos mejoran la precisión y capturan patrones dinámicos, bajo ningún punto de vista reemplazan las tablas de mortalidad, que continúan siendo la base fundamental para la tarificación y evaluación actuarial.

Trabajar con XGBoost y LSTM implica enfoques distintos debido a la forma en que manejan los datos y a sus capacidades predictivas. **XGBoost es un modelo supervisado de Boosting**, diseñado para aprender patrones a partir de conjuntos de datos estructurados con múltiples características o variables explicativas. En este caso, el modelo aprende relaciones entre las variables y la mortalidad, mejorando iterativamente su desempeño a través de múltiples árboles de decisión. En contraste, **LSTM es una red neuronal recurrente optimizada para trabajar con series temporales**, utilizando valores previos de la variable objetivo para predecir valores futuros. Esto le permite capturar patrones de largo plazo y estacionales, sin necesidad de incorporar variables explicativas adicionales.

Desde una perspectiva operativa, XGBoost destaca en la segmentación de datos y predicción supervisada, ya que su fortaleza radica en la identificación de factores específicos que influyen en la mortalidad de cada segmento. Sin embargo, LSTM sobresale en la captura de tendencias y variaciones temporales, al trabajar directamente con la serie histórica sin requerir bases de datos constantemente actualizadas. Ambos enfoques son complementarios: mientras XGBoost se adapta mejor a la predicción basada en características individuales de los asegurados, LSTM es más efectivo en la modelación de patrones de evolución de la mortalidad.

En términos de preparación de datos, los modelos Boosting requieren una matriz de entrada con múltiples variables explicativas y, en el caso de datos temporales, características adicionales como valores rezagados ($t-1$) o indicadores estacionales. Esto implica un mayor trabajo de preprocesamiento y diseño de características. En cambio, LSTM trabaja directamente con la serie temporal, lo que simplifica el proceso de preparación de datos al no requerir la creación de nuevas variables externas. Esta diferencia es clave en términos de eficiencia operativa y facilidad de implementación.

Desde la perspectiva de capacidades predictivas, los modelos Boosting están limitados a predicciones punto a punto, es decir, para proyectar t, necesitan datos de entrada correspondientes a $t-1$. Si se requiere proyectar más a futuro, se deben generar nuevas características manualmente para cada paso. Por el contrario, LSTM permite realizar predicciones automáticas de múltiples pasos sin necesidad de datos adicionales, ya que el modelo se retroalimenta de sus propias predicciones. Esto lo convierte en una mejor opción cuando la dependencia temporal es un factor clave en la predicción.

En conclusión, XGBoost y LSTM no compiten entre sí, sino que se complementan. XGBoost es ideal para problemas donde múltiples variables explicativas impactan la mortalidad, mientras que LSTM es más eficiente para analizar series temporales puras. Sin embargo, ninguno de estos modelos reemplaza las tablas de mortalidad, que continúan siendo el estándar de referencia en la industria actuarial. Su combinación puede ofrecer una visión más completa de la mortalidad proyectada, aprovechando la precisión de XGBoost en la clasificación individual y la capacidad de LSTM para identificar patrones temporales a largo plazo.

Limitaciones y trabajo futuro

Entre las principales limitaciones de este estudio se encuentra la dependencia de los datos históricos disponibles, los cuales pueden verse afectados por eventos atípicos o cambios regulatorios. Además, los modelos utilizados, si bien presentan mejoras sobre los enfoques tradicionales, aún pueden ser optimizados en términos de interpretabilidad y eficiencia computacional.

Como trabajo futuro, se recomienda explorar arquitecturas más avanzadas, como el uso de capas de atención en modelos LSTM o GRU, lo que permitiría mejorar la capacidad del modelo para capturar dependencias a largo plazo en la serie temporal de mortalidad.

Asimismo, sería valioso evaluar enfoques basados en **modelos tipo transformer**, específicamente **Time Series Transformer (TST)** o **Temporal Fusion Transformer (TFT)**.

- **Time Series Transformer (TST)**: Este modelo adapta la arquitectura transformer a datos secuenciales, utilizando autoatención para capturar relaciones a largo plazo en series temporales. Puede mejorar la predicción de mortalidad al detectar patrones de largo plazo que los modelos tradicionales pueden pasar por alto.

- **Temporal Fusion Transformer (TFT)**: Es un modelo diseñado para problemas de series temporales en el ámbito financiero y actuarial, combinando mecanismos de autoatención con redes recurrentes. Permite interpretar mejor la influencia de distintas variables en la predicción de la mortalidad.

Finalmente, la incorporación de variables adicionales, como factores macroeconómicos y biométricos, podría aportar mayor precisión a las estimaciones y mejorar la capacidad predictiva de los modelos. Además, el desarrollo de modelos híbridos que combinen técnicas de boosting, redes neuronales recurrentes y transformers podría representar una evolución significativa en la modelización de la mortalidad.

Bibliografía

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

Hochreiter, S., & Schmidhuber, J. (1997). *Long Short-Term Memory*. Neural Computation, 9(8), 1735-1780.

Cho, K., et al. (2014). *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*. arXiv preprint arXiv:1406.1078.

Blake, D., Cairns, A., & Dowd, K. (2006). *Living with Mortality: Longevity Bonds and Other Mortality-Linked Securities*. British Actuarial Journal, 12(1), 153-197.

OpenAI (2024). *Neural Network Illustration*. [Imagen de red neuronal ilustrativa].

Tablas de Mortalidad Actuarial (2024). *Documentación interna de nuestra compañía de seguros sobre la construcción y ajuste de tablas de mortalidad*.

Apuntes de las clases de redes neuronales (2024). *Documentación provista por el profesor Juan Velásquez.*

Apéndice

A continuación, se describen los métodos utilizados para la construcción de las tablas de mortalidad:

Modelo de ajuste Whitaker-Henderson de tipo B

Esta metodología es usada para construir las probabilidades de las edades pertenecientes al tronco de la tabla (aproximadamente de los 60 a 89 años), lo cual se denomina graduación. Para abreviar, se usará la terminología "graduación W-H" o, simplemente, "W-H".

El modelo matemático de W-H es un modelo no paramétrico que considera que los estimadores de las fuerzas de mortalidad, u_x , se escogen de manera que minimicen la siguiente expresión.

$$M = F + hS = \sum_{x=1}^{n} w_x (u_x - q_x)^2 + h \sum_{x=1}^{n-z} \Delta^z u_x$$

En donde

(1) $w_x = \frac{n_x}{u_x(1-u_x)}$ [1] O (2) $w_x = \frac{n_x}{\bar{n}}$ [2]

n_x: Tamaño de la muestra de edad x

$$\bar{n} = \frac{\sum_{x=x_1}^{x=x_2} n_x}{(x_2 - x_1 + 1)}$$

Z: orden de diferencias finitas, la curva se ajustará a un polinomio de grado 3 o 4.

[1] Se usan valores bajos de h (entre 100 y 10.000) cuando se seleccionan pesos w_x del tipo (2)

[2] Se usan valores altos de h (entre $1.000.000$ y $100.000.000$) cuando se seleccionan pesos w_x del tipo (1)

Modelo de ajuste: Modelos Aditivos Generalizados (GAM)

Un modelo aditivo generalizado (GAM) es un modelo lineal generalizado (GLM) en el cual el predictor está dado por la suma de funciones suavizadoras de forma no lineales. En general, el modelo tiene una estructura similar a la siguiente:

$$\log\big(E(y_i)\big) = \alpha + f_1(x_{1i}) + f_2(x_{2i})$$

Donde la variable de respuesta y_i sigue una distribución de Poisson, de la misma forma que lo hace la fuerza de mortalidad, y cuya función de enlace es logarítmica, las funciones f_1 y f_2 son funciones suavizadoras de las covariables x_1 y x_2. El modelo GAM representa las funciones de suavizamiento usando splines con penalizaciones, estos términos de suavizamiento pueden ser funciones de cualquier número de covariables.

En términos generales, el paquete en R (MGCV) genera el modelo GAM construyendo primero funciones de base y una o más matrices cuadráticas de coeficientes de penalización para cada término de suavizamiento que compone el modelo, obteniendo así una matriz para la parte estrictamente paramétrica del modelo GAM y combinando éstas para obtener una matriz del modelo completo.

El modelo GAM resuelve el problema de maximización de las penalizaciones por iteraciones mediante el método P-IRLS (Penalized Iteratively Re-weighted Least Squares) (Wood 2000).

Modelo de ajuste Makeham

El modelo de ajuste de Makeham es utilizado para la construcción de las probabilidades que no están dentro del tronco graduado, es decir, para las edades inferiores y superiores a este.

La función de Makeham corresponde a una modificación de la función de Gompertz, que consiste en poder cambiar el nivel de la fuerza de mortalidad, agregándole una constante, lo que se representa como:

$$\mu_x = A + Bc^x$$

Para la construcción de las colas inferior y superior, se utiliza el método de interpolación de Makeham, que consiste en ajustar la tendencia de la curva utilizando 3 puntos equidistantes de referencia x, x+r, x+2r, resolviendo las siguientes ecuaciones:

$$q_x = 1 - sg^{c^{x(c-1)}}$$

$$q_{x+r} = 1 - sg^{c^{(x+r)(c-1)}}$$

$$q_{x+2r} = 1 - sg^{c^{(x+2r)(c-1)}}$$

Donde las incógnitas son s, g, c y los valores q_x son datos empíricos.

q_x se escoge desde la graduación, en cambio, q_{x+r} y q_{x+2r} son tasas de mortalidad obtenidas a partir de tablas de referencia. Estas dos últimas se denominan tasas o probabilidades de referencia, las cuales son mejoradas a sus respectivos Factores de Mejoramiento (FM) al año central de la tabla, según corresponda.

Calidad de Ajuste

Usando diferentes pruebas estadísticas se verifica la calidad de ajuste que poseen las graduaciones calculadas. Dichas pruebas son 4 y corresponden a:

- Test de los signos
- Test de las corridas
- Test chi-cuadrado
- Test de Kolmogórov-Smirnov

Definiciones

Aquí se definen algunos elementos que ayudan a comprender la segmentación de cartera del punto siguiente.

- Causantes: Asegurados dueños de la póliza que no presentan condición de invalidez, es decir, sanos.
- Beneficiarios: Asegurados que son beneficiarios de pólizas de algún causante que no presentan condición de invalidez, es decir, sanos.
- Inválidos: Causantes o Beneficiarios que presentan condición de invalidez, ya sea parcial o total.