Escuela de Ingeniería y Ciencias, Campus Monterrey

Inteligencia artificial avanzada para la ciencia de datos I (TC3006C.102)

Momento de Retroalimentación: Reto Limpieza del Conjunto de Datos

Equipo 4:

Karla Andrea Palma Villanueva (A01754270)

Viviana Alanis Fraige (A01236316)

David Fernando Armendariz Torres (A01570813)

Alan Alberto Mota Yescas (A01753924)

Adrián Chávez Morales (A01568679)

Jose Manuel Armendáriz Mena (A01197583)

Docentes:

Alfredo Esquivel Jaramillo

Mauricio Gonzalez Soto

Frumencio Olivas Alvarez

Antonio Carlos Bento

Hugo Terashima Marín

Monterrey, Nuevo León, México. 9 de septiembre de 2024

Índice

Abstract

Este documento presenta un análisis exhaustivo del conjunto de datos del Titanic con el objetivo de desarrollar un modelo de aprendizaje automático capaz de predecir la supervivencia de los pasajeros. Se llevó a cabo un proceso meticuloso de limpieza y preprocesamiento de datos, que incluyó la imputación de valores ausentes y la eliminación de variables irrelevantes. Se exploraron diversas características del conjunto de datos, como la clase social, el género y la edad, y se seleccionaron variables clave para el modelado. Se probaron tres modelos de aprendizaje automático: Support Vector Machine (SVM), Random Forest y Multilayer Perceptron (MLP), evaluando su rendimiento en función de métricas como precisión, recall y F1-score. Finalmente, se eligió el modelo Random Forest por su robustez y capacidad de manejar características irrelevantes, logrando un rendimiento óptimo en la clasificación de supervivientes.

Keywords— Support Vector Machines (SVM), Multilayer Perceptron (MLP), Random Forest, Cross Validation, Grid Search, Accuracy, F1 Score.

1. Introducción

La inteligencia artifical se ha convertido en una herramienta fundamental en la industria, debido a su alto impacto en la toma de decisiones, principalmente en el ánalisis de datos. Entre sus ramas, el Aprendizaje Automático, también conocido como Machine Learning, destaca por su capacidad para mejorar continuamente su desempeño al aprender de los datos, en lugar de depender únicamente de la programación explícita, esta disciplina se basa en la aplicación de modelos estadísticos que buscan hacer predicciones precisas a partir de datos históricos.

El Aprendizaje Automático puede abordarse de diferentes maneras, y uno de sus enfoques más comunes es el Aprendizaje Supervisado. Se sabe que para realizar las predicciones se requieren de datos de entrada y de salida, por lo que para este tipo de modelos se requieren de datos etiquetados que sirven para el entrenamiento del mismo y así cumplir con su función de ejecutar las predicciones sobre los nuevos datos.

En este sentido, uno de los desafíos más comunes en la comunidad de las ciencia de datos es el "Titanic - Machine Learning from Disaster". Dicho desafío además de aplicar las

técnicas de aprendizaje automático, posee información realista mediante un análisis de datos del Titanic y permite a los algoritmos aprender de datos históricos para hacer predicciones sobre eventos futuros

2. Objetivo

El objetivo planteado corresponde a la creación de un modelo de aprendizaje automático con la capacidad de predecir la supervivencia de un individuo ante la tragedia del hundimiento del Titanic. Se busca generar un modelo de buen desempeño dada una entrada de caracterísitcas de un pasajero hipotetico.

Para cumplir este objetivo, se entrenarán multiples modelos de aprendizaje supervisado para clasificar el estatus del pasajero (sobrevivió, pereció). Los modelos serán entrenados mediante un set de datos de características de pasajeros pasados como su clase social, edad, sexo y tarifa, entre otras cosas.

Ante este objetivo, primeramente se deben preparar los datos, este es el objetivo puntual de esta primera etapa. Algo sumamente importante para poder aplicar cualquier tipo de modelo son las actividades de limpieza y organización de datos, en las cuales, se trata de limpiar y suavizar cualquier inconsistencia que impida que el conjunto de datos sea coherente. El proceso de limpieza que se llevará a cabo se compone de manejo de datos nulos, caracteres problemáticos, inconsitencias en los datos, etcétera.

Este proceso es muy importante porque establecerá las bases para un análisis más extenso, permitiendo que los modelos de aprendizaje automático trabajen con un dataset ordenado y correctamente estructurado.de tal forma que el modelo sea más preciso y su rendimiento de resultados sea eficiente y viable.

3. Manejo del Set de Datos

La exploración, limpieza y preprocesamiento de los datos se realizó a la par complementando cada una de estas partes del proces entre si. A continuación se presenta el proceso referente al manejo de los datos como se llevó a cabo.

3.1. Exploración | Exploración Inicial del Set de Datos

Los datos de los pasajeros del Titanic se encuentran por defecto dividios en dos conjuntos distintos: un conjunto de entrenamiento con 891 instancias y un conjunto de prueba con 418. El conjunto de entrenamiento cuenta con 12 variables, 10 características, 1 llave y 1 variable objetivo. El conjunto de prueba cuenta con las misma variables, excluyendo la variable objetivo. El cuadro 1 muestra una descripción de las variables del set de datos proveniente de Kaggle (Cukierski, 2012). El cuadro 2 muestra los primeros 5 registros del set de datos. Con base en la siguiente información, se descarta la variable *Ticket* dada su dificultad de manejo como dato tipo texto.

Variable	**Tipo**	**Descripción**	**Especificaciones**
PassangerId	Llave	Identificador de pasajeros.	Inicia en 1.
Survival	Categórica: binaria	Variable Objetivo: Supervivencia del pasajero	0 = No, 1 = Sí
Pclass	Categórica: ordinal	Estatus socioeconómico	1 = 1ª, 2 = 2ª y 3 = 3ª
Name	Texto	Nombre y título del pasajero	-
Sex	Categórica: binaria	Sexo del pasajero	-
Age	Numérica	Edad del pasajero	La edad puede ser fraccionaria si es menor de 1 año, de lo contrario es entera.
SibSp	Numérica	Número de hermanos o cónyuges a bordo	-
Parch	Numérica	Número de padres o hijos a bordo	Si los niños viajaron solo con una niñera el valor es cero en esos casos.
Ticket	Texto	Número de ticket del pasajero	-
Fare	Numérica	Costo del ticket	-
Cabin	Categórica: nominal	Número de cabina	-
Embarked	Categórica: nominal	Puerto de embarque	C = Cherbourg, Q = Queenstown, S = Southampton

Cuadro 1: Descripción de datos.

PassengerId	Survived	pclass	Name	sex	Age	sibsp	parch	ticket	fare	cabin	embarked
1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	female	38.0	1	0	PC 17599	71.2833	C85	C
3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

Cuadro 2: Datos en crudo.

3.1.1. Exploración | Balance de Clases

Las clases del set de datos de entrenamiento se encuentran desbalanceadas, los difuntos comprenden el $\sim 62\,\%$ de los registros (549) mientras que los sobrevientes el $\sim 38\,\%$ (342). Este desbalance indica que *F1-Score* es la métrica adecuada para medir el desempeño general de los modelos.

3.1.2. Exploración | Registros Nulos

El cuadro 3 muestra la cantidad de registros nulos por variable. Con base en esta información, se descarta Cabin debido a la cantidad exhorbitante de de registros ausentes.

Variable	Registros Nulos
PassengerId	0
Survived	0
Pclass	0
Name	0
Sex	0
Age	177
SibSp	0
Parch	0
Ticket	0
Fare	0
Cabin	687
Embarked	2

Cuadro 3: Cantidad de registros nulos por variable.

3.2. Pre-Procesamiento | Títulos de los Pasajeros

Se creó una variable adicional *Titles* la cual contiene los títulos de cada pasjero extraidos de la variable *Name*. Para facilitar el manejo de estos datos, se agruparon los títulos en las

siguientes categorías, títulos comunes (CT), títulos profesionales (PT), títulos militares (MT) y títutlos de nobleza (NT). La agrupación se realizó en una columna *Titles_ Grouped*.

Adicionalmente, algunos títulos se presentan como sinonimos de otros, estos se reasignaron con su sinónimo. "Mlle' se transformó en "Miss" y "Mme" en "Mrs'

Por último, el título "Ms" presenta ambiguedad ya que podría ser considerado tanto "Miss" como "Mrs'. Para solucionar esta encrucijada, ya que solo un pasejaro cuenta con dicho título, se investigó su estatus marital al momento de los eventos del titanic. La pasajera Ms. Encarnacion Reynaldo no se encontraba casada a la fecha del accidente por lo cual se reasignó su título a "Miss" (*Encarnación Reynaldo: Titanic Survivor*, 1996). El cuadro 4 muestra cada título, su cantidad de registros y la agrupación que se le asigno.

Título	**Cantidad de Registros**	**Agrupación**
Mr	517	CT
Miss	182	CT
Mrs	125	CT
Master	40	CT
Dr	7	PT
Rev	6	PT
Mlle	2	CT
Major	2	MT
Col	2	MT
the Countess	1	NT
Capt	1	MT
Ms	1	CT
Sir	1	NT
Lady	1	NT
Mme	1	CT
Don	1	NT
Jonkheer	1	NT

Cuadro 4: Tabla de Títulos, Cantidad de Registros y Agrupación.

3.3. Pre-Procesamiento | Manejo de Datos Categóricos

Los datos categóricos se manejaron con base en su clasificación. Para los datos binarios y ordinales (*Sex, Pclass)* se utilzó *Label Encoding*. En el caso de *Pclass*, se invirtió el orden de módo que la clase socieconomica mas alta tenga el valor mas alto: alta - 3, media - 2, baja -

1.

Para los datos ordinales (*Embarked, Titles_ Grouped*) se empleó *One Hot Encoding*

Para eliminar redundancias, las columnas originales *Embarked* y *Titles_ Grouped* se eliminaron. *Titles* se utilizará mas adelanta tras lo cual se eliminará de igual manera. El cuadro 5, muestra la estructura de un registro tras la transormaciones mencioandas.

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Fare	Titles	C	Q	S	CT	MT	NT	PT
1	0	1	Braund, Mr. Owen Harris	1	22.0	1	0	7.25	Mr	0	0	1	1	0	0	0

Cuadro 5: Registro tras manejo de datos categóricos.

3.4. Exploración | Identificar Variables Relevantes

A continuación se realiza un análisis de la relevancia de cada variable para la predicción de los modelos. En primera instancia se seleccionan la variables que aparentan ser relevantes para la supervivencia de un pasajero. Estas variables son: *Sex, Pclass, Embarked, Parch, Age, Titles* y se grafica la tasa de supervivencia respecto a cada una de estas varaibles en la figura 1.

La diferencia entre tasas de supervivencia por cada categoría de las varaibles denota que estas tienen algun impacto sobre la supervivencia de un pasajero. Para validar esta aseveración, se entrena un modelo de regresión logística y se ordenan las variables con base en el peso asignado dentro de la función del modelo. El orden por relevancia generado se presenta en el cuadro 6.

Con base en esta información, la variable *Fare* se descarta por poca relevancia y las variables *S*, *CT* se descartan para evitar multicolinealidad. Por último la variable *Parch* se utiliza para generar una nueva variable compuesta $Family\_Size = SibSp + Parch + 1$, que corresponde al tamaño de la familia de un individuo a bordo del titanic. Tras esto se eliminan *Parch* y *SibSp* para eliminar redundancias. El resto de las variables se consideran relevantes para la predicción de los modelos.

3.5. Limpieza | Manejo de Datos Ausentes

De las variables relevantes, solo 2 contienen datos asuentes: *Embarked* y *Age*.

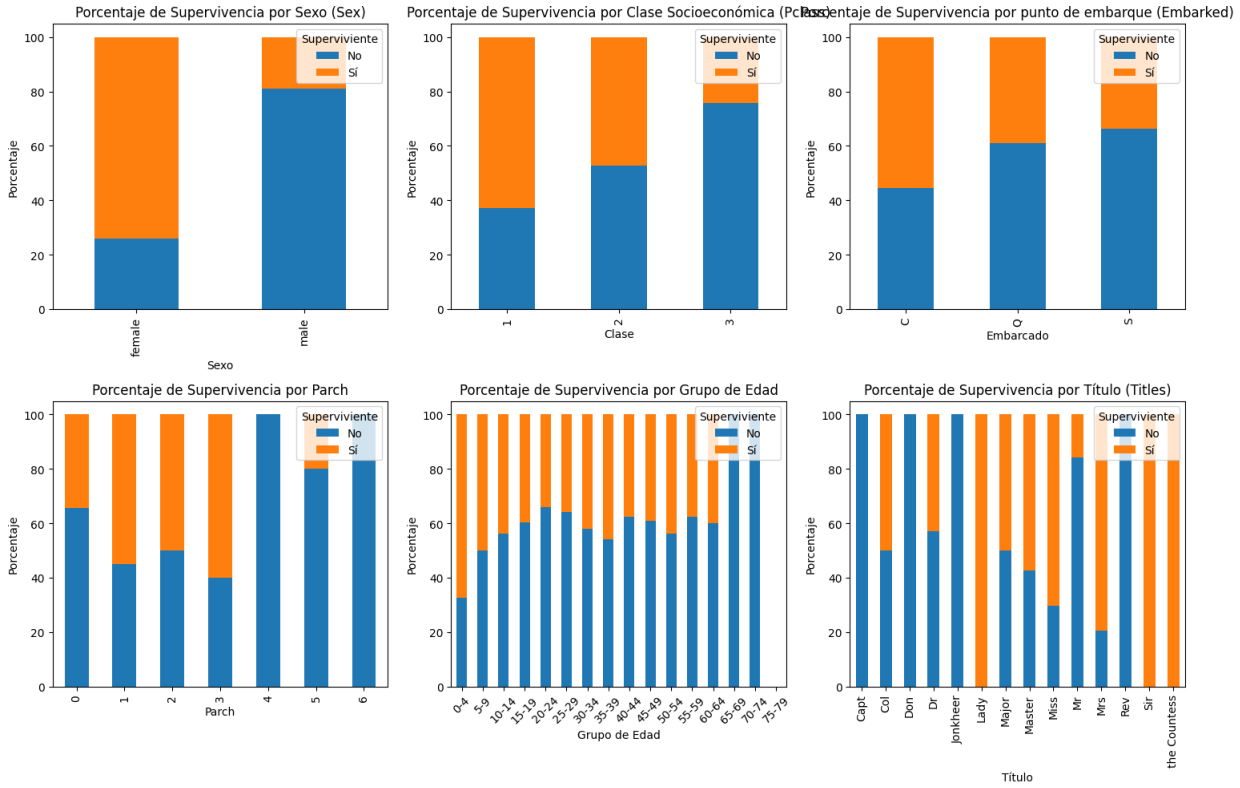

Figura 1: Tasas de supervivencia por variable de interés.

Característica	**Coeficiente**
Sex	-2.317112
Pclass	1.059715
Age	-0.641752
SibSp	-0.401230
Q	-0.336596
MT	0.303799
PT	-0.280902
C	0.237739
NT	-0.236133
CT	0.208747
S	0.094367
Parch	0.052758
Fare	0.003009

Cuadro 6: Tabla de Características y Coeficientes.

3.5.1. *Embarked*

Embarked cuenta con solo dos registros ausentes por lo cual se optó por investigar el puerto de embarque de ambos pasajeros, este se identificó como *Southampton* (*Martha Evelyn*

Stone: Titanic Survivor, 1996), (*Rose Amélie Icard: Titanic Survivor*, 1996) y se ingresó manualmente.

3.5.2. *Age*

Para la imputación de las edades ausentes se utilizaron medidas de tendencia central por título. Primeramente se identificaron los títulos con datos ausentes en edad, el resultado de esto se puede observar en el cuadro 7.

Título	**Edades Ausentes**
Dr	1
Master	4
Miss	36
Mr	119
Mrs	17

Cuadro 7: Tabla de Títulos y Edades Ausentes.

Para el título *Dr* al contener solo un registro ausente, este se ingresó manualmente con base en investigación, el Dr. Arthur Jackson tenía una edad a la fecha de la tragedia (*Arthur Jackson Brewe: Titanic Victim*, 1996).

El resto de las edades es rellenaron con medidas de tendencia central referentes al respectivo título. *Mr* y *Miss*, al contener outliers en edad, como se muestra en la figura utilizaron la mediana de 30.0 y 21.0 respectivament. En cambio *Master* y *Mrs* emplearon la media de 4.6 y 35.8 respectivamente. Estas tecnicas se conocen como imputación media incondicional y imputación mediana (Arteaga y Ferrer-Riquelme, 2009). Tras esto se eliminó la columna *Titles* para eliminar redundancia.

3.6. Pre-Procesamiento | Estandarización

Con la intención de mejorar el desempeño de los modelos a emplear, asi como los recuros computacionales empleados, las variables numericas (*Age, Familiy _Size*) se estandarizaron.

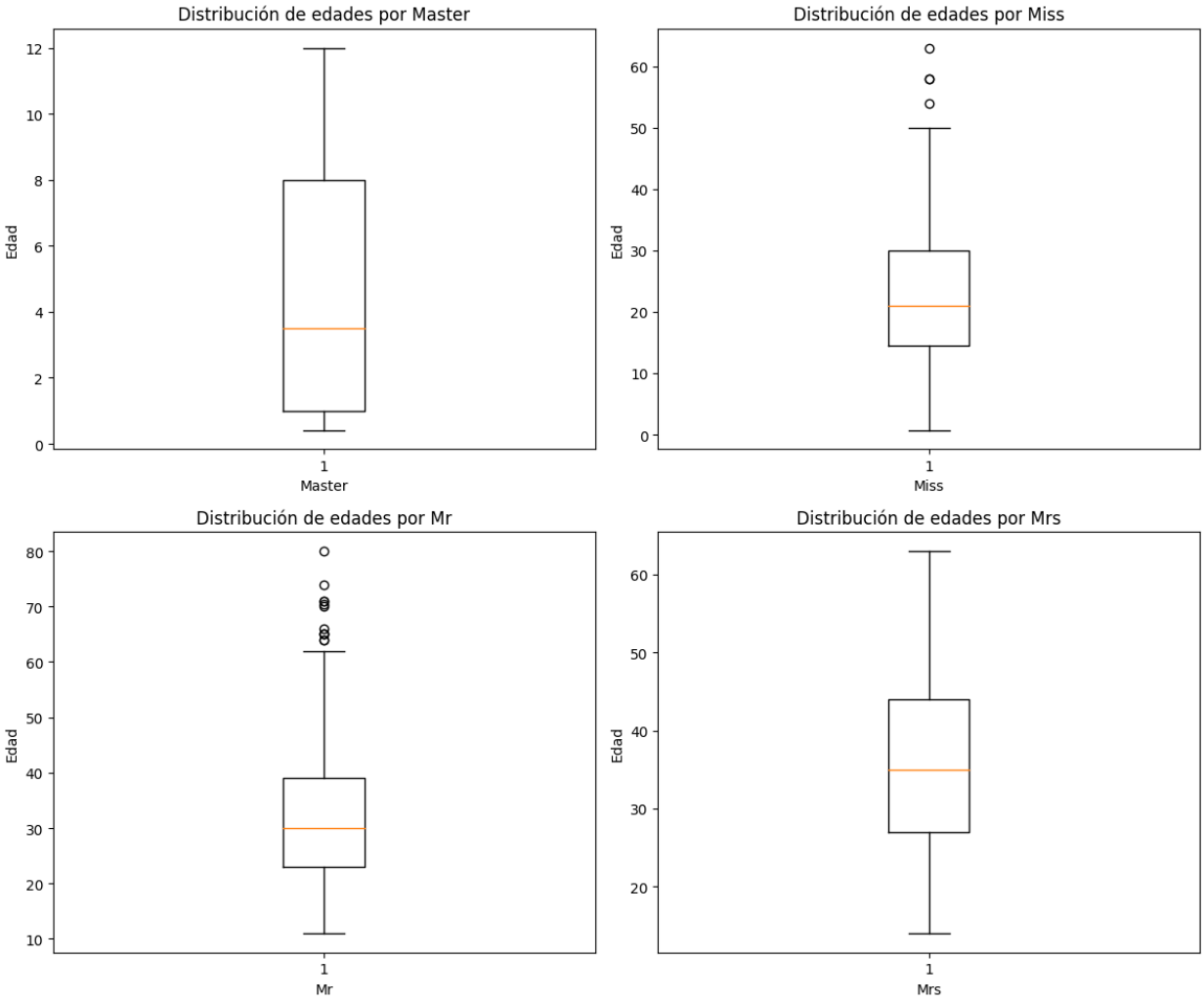

Figura 2: Gráficos Caja y Bigotes de Edad por Título.

3.7. Set de Datos Final

A continuación se presenta la estructura final del set de datos asi como un resumen de lo que se realizó sobre cada variable.

- ***PassengerId*:** Se utilizó como el índice de los datos.
- ***Survived*:** Esta variable es el 'target', será aislada en otro dataset.
- ***Pclass*:** Esta variable describe el estatus socioeconómico del pasajero. Tiene 216 registros con '1', 184 con '2', y 491 con '3'. Se transformó mediante *Label Encoding*.
- ***Name*:** Se utilizó para extraer los títulos de los pasajeros tras lo cual se descartó por

falta de relevancia.

- ***Sex*** Tiene 577 registros con *male* y 314 con *female*. Se transformó utilizando *Label Encoding*.
- ***Age*:** Se utilizaron métodos manuales e imputación media y mediana para rellenar los datos ausentes. Se estandarizó.
- ***Sibsp*:** Se combinó con *Parch* para generar *Family_ Size* tras lo cual se descartó para eliminar redundancia.
- ***Parch*:** Se combinó con *SibSp* para generar *Family_ Size* tras lo cual se descartó para eliminar redundancia. Se estandariza.
- **Ticket:** Esta variable se descarta debido a la dificultad de su manejo al ser tipo texto.
- ***Fare*:** El costo del ticket se descarta dada su baja relevancia para la predicción de supervivencia.
- ***Cabin*:** Se descarta debido a la cantida exhaustiva de datos ausentes.
- ***Embarked*:** Se utilizan téncincas manuales para rellenar los datos ausentes. Se transforma mediante *One Hot Encoding*, las columnas resultantes de esto son Q, C y S, la ultima se descarta para evitar multicolinealidad. La columna original se elimina para evitar redundancia.
- ***Titles*:** Se extrae de *Name* para la imputación de edades y la creación de *Title_ Groups*. Tras esto se descarta.
- ***Title_ Groups*:** Se transforma mediante *One Hot Encoding*, las columnas resultantes de est son CT, PT, MT, NT, la primera se descarta para evitar multicolinealidad, la original se descarta para evitar redundancia.

PassengerId	Pclass	Sex	Age	C	Q	MT	NT	PT	Family_Size
1	1	1	-0.559038	0	0	0	0	0	0.059160
2	3	0	0.648087	1	0	0	0	0	0.059160
3	1	0	-0.257257	0	0	0	0	0	-0.560975
4	3	0	0.421751	0	0	0	0	0	0.059160
5	1	1	0.421751	0	0	0	0	0	-0.560975

Cuadro 8: Estructura Final del Set de Datos, 5 Registros.

4. Selección, Configuración y Entrenamiento de los Modelos

4.1. Identificación del modelo

Para abordar el problema de predecir quién sobrevivirá al naufragio del Titanic, se implementarán tres modelos de Aprendizaje Automático: MLP, SVM y Random Forest. Se emplearon estos tres modelos debido a su capacidad para generar clasificación binaria y sus variantes grados de complejidad, así como diversas fortalezas y debilidades. Esto con la intención de generar un variado repertorio de modelos de los cuales poder seleccionar el mejor. De estos 3 modelos se busca seleccionar uno para refinar.

4.2. Modelos a analizar

- **SVM**: Una Máquina de Vectores de Soporte, SVM por sus siglas en inglés, es un modelo de aprendizaje supervisado utilizado para la clasificación binaria. Este modelo clasifica datos generando un hiperplano de $n-1$ dimensiones en un espacio de n dimensiones, donde n es la cantidad de features empleados. El hiperplano generado segmenta el espacio optimizando el margen entre los datos más cercanos de ambas clases, los datos empleados se denominan vectores de soporte. Dicha separación se utiliza para diferenciar entre ambas clases (IBM, s.f.).

  El modelo de clasificación busca generar una separación lineal entre ambas clases, en dado caso que los datos no sean linealmente separables se utilizar una función kernel para proyectar los datos. Esta proyecta los datos, sin transformarlos, en una dimensión superior en la cual sean linealmente separables (Wilimitis, 2019). Adicionalmente, en

aquellos casos en los cuales los datos no sean perfectamente linealmente separables, el margen generado se clasifica como suave, lo que permite ciertos errores de clasificación, lo que a su vez, en datos reales, hace del modelo de clasificación más robusto ante datos atípicos.

- **Random Forest**: El Random Forest es un modelo de aprendizaje supervisado que combina la salida de múltiples árboles de decisión para alcanzar un solo resultado. Su facilidad de uso y flexibilidad han impulsado su adopción, ya que maneja problemas de clasificación y regresión (IBM, s.f.).

  Para su funcionamiento, este modelo elige una muestra aleatoria del conjunto de datos. Cada uno de estos se entrena de distinta manera para que el algoritmo combine todos los resultados posibles y así genere resultados basados en la votación. Para su procedimiento se inicia seleccionando aleatoriamente un número 'n"de registros del conjunto de datos original. A partir de cada muestra seleccionada, se construye un árbol de decisión que genera un resultado que se basa en los datos que le fueron asignados. Finalmente, para su clasificación, el resultado final se determina por la mayoría de votos de estos árboles generados (*Random forest, la gran técnica de Machine Learning*, 2023).

- **MLP**: Clasificador Perceptrón Multicapa es un modelo de aprendizaje supervisado que optimiza la función de descenso de gradiente estocástico, a través de redes neuronales. Se compone de múltiples capas de neuronas interconectadas, en las que las salidas de las neuronas de una capa se convierten en entradas para la siguiente capa. La primera capa se llama capa de entrada, la última capa se llama capa de salida y las capas intermedias se llaman capas ocultas, en las cuales cada conexión tiene un peso asociado que se ajusta durante el proceso de entrenamiento (*Qué es Perceptrón Multicapa - MLP / Concepto y definición. Glosario*, s.f.).

  Este modelo es uno de los más utilizados de redes neuronales, debido a su capacidad para modelar tareas de clasificación y regresiones no lineales; además, la arquitectura que se implementa, permite que el modelo aprenda representaciones complejas de los datos.

  La elección principal de este modelo se realizó, puesto que es apto para el manejo

de relaciones no lineales entre variables complejas como las que posee el Titanic, como ejemplo, se tiene la edad, el género, la clase de boleto, entre otras. Dichas características pueden influir en la probabilidad de supervivencia y por consiguiente, el modelo se encargará de capturar estas interacciones y patrones en los datos, permitiendo una predicción más precisa entre los pasajeros que sobrevivieron y los que no.

4.3. Configuración y entrenamiento

Dado que cada modelo cuenta con diferentes parámetros e hiperparámetros, la configuración empleada varía. Es pertinente destacar que estas implementaciones son preliminares para observar el desempeño, tras lo cual se seleccionará y refinará un modelo. Todos los modelos se entrenaron con el 80 % de los datos del set de entrenamiento proporcionado por Kaggle y se evaluaron con el 20 %.

4.4. SVM

Para el modelo de *Support Vector Machine* (SVM), se empleó *Grid Search Cross-Validation* para seleccionar el mejor kernel. El proceso evaluó diferentes valores para el hiperparámetro `kernel`, los cuales definen la transformación de los datos.

Los kernels evaluados fueron:

- `linear`
- `poly`
- `rbf`
- `sigmoid`

El proceso de búsqueda se llevó a cabo usando validación cruzada estratificada de 5 particiones (*5-Fold Stratified Cross-Validation*), lo que permite asegurar que la proporción de clases se mantenga constante en cada partición, mejorando la robustez del modelo.

Para el resto de los hiperparámetros del SVM, se utilizaron las configuraciones por defecto de `sklearn`, tales como:

- `C=1.0` (parámetro de regularización)
- `gamma='scale'` (para kernels no lineales)

Los mejores parámetros preliminares encontrados fueron:

```
{'kernel': 'linear'}
```

El desempeño del modelo, medido en el conjunto de prueba, fue:

- ***Accuracy***: 0.83
- ***F1-Score*** (promedio ponderado): 0.83

4.5. Random Forest

El modelo de *Random Forest* fue configurado con hiperparámetros básicos, sin embargo, se ajustó el criterio para la medida de la calidad de la división (`criterion`), utilizando los valores:

- `gini`
- `entropy`
- `log_loss`

Se utilizó una validación cruzada estratificada de 5 particiones para garantizar la robustez del modelo y evitar sobreajuste. 'overfitting'

Los mejores parámetros preliminares encontrados fueron:

```
{'criterion': 'gini'}
```

El desempeño del modelo, medido en el conjunto de prueba, fue:

- ***Accuracy***: 0.82
- ***F1-Score*** (promedio ponderado): 0.82

4.6. MLP

Para el modelo de *Multi-Layer Perceptron* (MLP), se entrenó de igual forma utilizando el proceso de *Grid Search Cross-Validation* para ajustar los hiperparametros. Los valores evaluados fueron:

- `hidden_layer_sizes`: (50,), (100,), (50,50)
- `activation`: `relu`, `tanh`
- `solver`: `adam`, `sgd`
- `alpha`: 0.0001, 0.05

El proceso de búsqueda se llevó a cabo utilizando validación cruzada estratificada de 5 particiones (*5-Fold Stratified Cross-Validation*), lo que garantiza que la proporción de clases se mantenga constante en cada partición, mejorando la robustez del modelo.

Los mejores parámetros preliminares encontrados fueron:

```
{'activation': 'tanh', 'alpha': 0.05, 'hidden_layer_sizes': (50,),
'learning_rate': 'constant', 'solver': 'adam'}
```

El desempeño del modelo, medido en el conjunto de prueba, fue:

- **Accuracy**: 0.85
- **F1-Score** (promedio ponderado): 0.85

4.7. Comparación de Modelos

Para evaluar los resultados de los modelos, se seleccionaron las siguientes métricas:

- ***F1-Score*:** Ayuda a encontrar un equilibrio óptimo entre identificar sobrevivientes correctamente (*recall*) y evitar predecir falsamente demasiados (precisión). Dado el desbalance de las clases, se emplea el promedio ponderado.
- **Exactitud (*Accuracy*):** Mide qué tan bien el modelo clasifica tanto los positivos como los negativos. Puede ser engañosa si las clases están desequilibradas. Este no sería útil, puesto que se da un peso a las variables, más la fuente "Kaggle" trabaja con el resultado de exactitud, así que es considerado para verificar la exactitud del modelo obtenido.

Los modelos también muestran los resultados de las métricas "Precisión", "Recall" y "Soporte". Sin embargo, estas no nos son relevantes en el contexto actual de los datos de sobrevivientes del Titanic, ya que no tendrían un impacto significativo en la actualidad.

Utilizando los valores obtenidos por el entrenamiento de cada modelo, se estarán comparando en el Cuadro 7:

Modelo	***F1-Score* (Promedio Ponderado)**	**Exactitud (*Accuracy*)**
MLP	85 %	85 %
SVM	83 %	83 %
Random Forest	82 %	82 %

Cuadro 9: Comparación de Modelos.

5. Refinamiento del Modelo

Con base en el desempeño observado en las implementaciones preliminares, el modelo seleccionado a refinar es *Support Vector Machine* (SVM). Este algoritmo fue elegido debido a su simplicidad en comparación con el *Multi-Layer Perceptron* (MLP) y su capacidad para manejar de manera eficiente problemas de clasificación, además de ofrecer un amplio margen para la mejora en futuras iteraciones.

A pesar de haberse realizado una evaluación exhaustiva con redes neuronales como MLP, que mostraron un mejor rendimiento en algunos casos, se decidió escoger SVM por su sim-

plicidad y equilibrio entre rendimiento y facilidad de ajuste. Esto facilita su implementación y ajuste en esta fase inicial, permitiendo desarrollar un modelo eficiente y fácilmente interpretable.

5.1. Optimización de hiperparámetros utilizando *Grid search*

Nuevamente se utilizaron téncicas combinadas de busqueda en cuadrícula y validación cruzada sobre el 80 % del set de datos de entrenamiento para identificar la configuración ideal. En esta ocasión se realizó una busqueda mas exhaustiva para identificar alguna configuración que mejore el *F1-Score* (promedio ponderado), 0.83, del modelo preliminar.

La estrategia de validación cruzada fue de 5-segmentos estratificados evaluados con la métrica *F1-Score*. Los elementos de los cuales se compone la cuadrícula de busqueda son:

- C: 0.01, 0.1, 1, 10, 100
- *kenrel*: lineal, polinómico, función de base radial, sigmoide.
- *gamma*: scale, auto
- *degree*: 2, 3, 4
- *coef0* 0.0, 0.1, 0.5, 1.0

Este proceso identificó el *kernel* de función de base radial(rbf) con $C = 1$ y *gamma* "auto" como los hiperparámetros ideales con un F1-Score de 0.88 (0.91 para difuntos y 0.81 para sobrevivientes).

Para refinar incluso mas el modelo, se repitió el proceso, ahora con una cuadrícula más exhaustiva para los hiperparámetros del kernel *rbf*. También se agregó el hiperparamétro *class_ weight* dado el desbalance de las clases.

- C: 0.1, 1, 10, 100, 1000
- *kenrel*: función de base radial
- *gamma*: 1, 0.1, 0.01, 0.001, 0.0001, scale, auto
- *class_ weight*: balanced, None

Finalmente, la configuración ideal no cambió respecto a la iteración anterior, es decir el modelo no se pudo mejorar mas. La configuración final del modelo es la siguiente:

- *kernel*: rbf
- *C*: 1
- *gamma*: scale
- *class_ weight*: None

El resto de los hiperparámetros se mantuvieron con la configuración por defecto de la clase sklearn.svm.SVC() de scickit-learn (Pedregosa y cols., 2011).

5.2. Evaluación Final del Modelo

Como ya se mencionó con anterioridad, las metricas empleadas para evaluar el modelo fueron *F1-Score* (Promedio Ponderado) y *Accuracy*. La primera de estas es la métrica mas acertada dada el balance de las clases presentes, la segunda se considera debido a la evaluación de por parte de Kaggle. El modelo tuvo un desempeño preliminar de 0.83 para ambas métricas, tras refinar el modelo se logró un incremento considerable a 0.88 en ámbas métricas.

En la figura 3 se puede observar cómo el modelo es mas propenso a equivocarse con falsos negativos que con falsos positivos. Dada la antigüedad del suceso y el contexto actual, no hay necesidad de dar mayor peso a alguno de estos tipos de errores. No obstante se intentó reducir el desbalance de estos errores mediante *weight_ class* sin mucho exito.

El conjunto de las técnicas de validación cruzada y segmentación de datos (prueba, entrenamiento) ayudó a reducir el riesgo a tanto sobreajuste como subajuste.

Para finalizar, el resultado de las predicciones generadas sobre el set de prueba designado por Kaggle fue evaluado con un accuracy de 0.79.

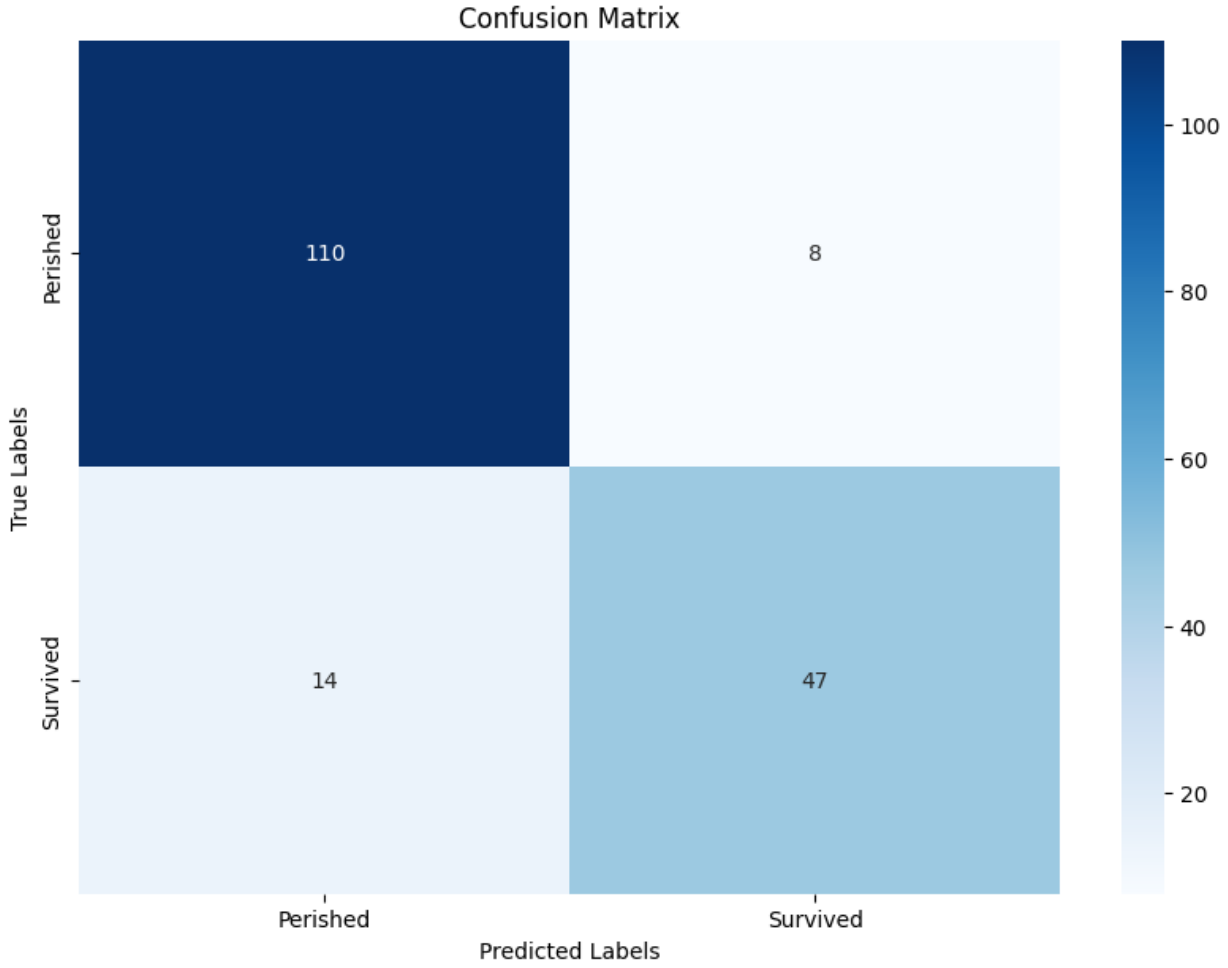

Figura 3: Matriz de Confusión, SVM Refinado.

6. Conclusión

El análisis y modelado del conjunto de datos del Titanic permitió identificar las variables más influyentes en la supervivencia de los pasajeros y desarrollar modelos predictivos eficaces. Se exploraron varios enfoques, como *Support Vector Machine* (SVM), *Random Forest* y *Multi-Layer Perceptron* (MLP), optimizando sus hiperparámetros mediante búsqueda por cuadrícula (*Grid Search*) y validación cruzada estratificada.

Entre los modelos, el SVM destacó por su capacidad para clasificar de manera precisa y eficiente, logrando una precisión del 0.83 y un *f1-score* de 0.81 con un kernel rbf. Aunque otros modelos, como *Random Forest* y *MLP*, mostraron un rendimiento competitivo, el SVM

demostró ser particularmente eficaz en este caso al evitar el sobreajuste y manejar con precisión las relaciones lineales entre las características. Adicional a esto, como ya se mencionó, su simple implementación preliminar en comparación al modelo MLP presentó el potencial de superar su desempeño.

Los resultados del modelo generado fueron satisfactorios, sin embargo el desempeño obtenido sobre el set designado y evaluado por Kaggle, pone un poco en duda tanto el desempeño del modelo como el muestreo original de los datos previo al tratamiento que se le dió en este reporte. Sería ideal ante esta situación emplear téncicas mas avanzada como la validación cruzada anidada para generar una evaluación mas robusta del modelo generado y asi generar predicciones con un desempeño mas acorde a lo evaluado en el reporte.

Referencias

Arteaga, F., y Ferrer-Riquelme, A. (2009). 3.06 - missing data. En S. D. Brown, R. Tauler, y B. Walczak (Eds.), *Comprehensive chemometrics* (p. 285-314). Oxford: Elsevier. Descargado de https://www.sciencedirect.com/science/article/pii/B9780444527011001253 doi: https://doi.org/10.1016/B978-044452701-1.00125-3

Arthur jackson brewe: Titanic victim. (1996, Sep). Descargado de https://www.encyclopedia-titanica.org/titanic-victim/arthur-jackson-brewe.html

Cukierski, W. (2012). *Titanic - machine learning from disaster.* Kaggle. Descargado de https://kaggle.com/competitions/titanic

Encarnación reynaldo: Titanic survivor. (1996, Sep). Descargado de https://www.encyclopedia-titanica.org/titanic-survivor/encarnacion-reynaldo.html

IBM. (s.f.). *Support vector machine.* Descargado de https://www.ibm.com/topics/support-vector-machine#:~:text=What%20are%20SVMs%3F,in%20an%20N%2Ddimensional%20space.

Martha evelyn stone: Titanic survivor. (1996, Sep). Descargado de https://www.encyclopedia-titanica.org/titanic-survivor/martha-evelyn-stone.html

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Qué es Perceptrón Multicapa - MLP | Concepto y definición. Glosario. (s.f.). Descargado de https://gamco.es/glosario/percepatron-multicapa-mlp/#:~:text=El%20Perceptr%C3%B3n%20Multicapa%20(MLP%2C%20por,el%20campo%20del%20aprendizaje%20autom%C3%A1tico

Random forest, la gran técnica de Machine Learning. (2023, 1). Descargado de https://www.inesdi.com/blog/random-forest-que-es/

Rose amélie icard: Titanic survivor. (1996, Sep). Descargado de https://www.encyclopedia-titanica.org/titanic-survivor/amelia-icard.html

Wilimitis, D. (2019, Feb). The kernel trick in support vector classification. *Towards Data Science*. Descargado de https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f