Tecnológico de Monterrey

Inteligencia Artificial Avanzada para la Ciencia de Datos (Gpo 102)

Curso:
TC3006C.102

Campus:
Monterrey

Limpieza del Conjunto de Datos

Equipo 2:

Rafhael Eduardo Chavez Ramirez | A00832228

Jose David de la Garza Salas | A00834760

Pablo Andrés Martínez Sánchez | A01252489

Andre Sebastian Galindo Posadas | A00833376

Daniel Sánchez Villarreal | A01197699

Lugar y Fecha:

Monterrey, Nuevo León

19 de Agosto del 2024

Índice

1. Introducción

El hundimiento del RMS Titanic es uno de los desastres marítimos más famosos de la historia, ocurrido en 1912 durante su viaje inaugural. Este evento trágico, en el que más de 1500 personas perdieron la vida, ha sido objeto de numerosos estudios e investigaciones. En el contexto del análisis de datos y el aprendizaje automático, el Titanic ofrece el desafío de predecir la supervivencia de los pasajeros basándose en diversas características personales y del viaje.

2. Definición del Problema

El objetivo de este reto es construir un modelo predictivo capaz de determinar si un pasajero sobrevivió al hundimiento del Titanic. Este desafío se basa en un conjunto de datos históricos que incluye diversas características de los pasajeros, como su edad, sexo, clase en la que viajaban, entre otros. Para esto, necesitamos crear un modelo de aprendizaje automático que prediga la supervivencia de los pasajeros utilizando las características disponibles en el dataset.

3. Origen del Dataset

El dataset utilizado en este proyecto proviene de Kaggle y está basado en registros reales del Titanic. Este conjunto de datos es un estándar en la comunidad de ciencia de datos para practicar técnicas de limpieza de datos, análisis exploratorio y modelado predictivo.

El dataset se divide en tres archivos:

- train.csv: Contiene los datos de entrenamiento con características y la variable objetivo (Survived).
- test.csv: Contiene características similares pero sin la variable objetivo, utilizado para probar el modelo.
- gender_submission.csv: Ejemplo de cómo se deben presentar los resultados de la predicción.

4. Exploración del Dataset Crudo

Contenido del Dataset:

- PassengerId: Identificador único de cada pasajero.
- Survived: Indica si el pasajero sobrevivió (1) o no (0).
- Pclass: Clase del boleto del pasajero (1 = 1ra clase, 2 = 2da clase, 3 = 3ra clase).

- Name: Nombre completo del pasajero.
- Sex: Sexo del pasajero.
- Age: Edad del pasajero.
- SibSp: Número de hermanos o cónyuges a bordo.
- Parch: Número de padres o hijos a bordo.
- Ticket: Número del boleto.
- Fare: Tarifa pagada por el boleto.
- Cabin: Número de cabina.
- Embarked: Puerto de embarque (C = Cherburgo, Q = Queenstown, S = Southampton).

Valores Crudos:

- En el dataset de entrenamiento (train.csv), se observaron valores faltantes en las columnas Age y Cabin.
- En la columna Age, hay 177 valores faltantes (19.86% de los datos). En Cabin, faltan datos en un 77.1% de los registros.

5. Análisis de los Valores

Distribución de Edad:

- La columna Age presenta una distribución sesgada hacia edades más jóvenes, con un promedio de alrededor de 29 años. Dada la cantidad de valores faltantes, se plantearon dos enfoques para abordar el problema: eliminar las instancias con valores faltantes o imputar los datos faltantes.

Distribución de Cabina:

- Dado el alto porcentaje de valores faltantes en la columna Cabin, y su posible irrelevancia comparada con otras características como la edad o la clase del boleto, se decidió eliminar esta columna.

Promedios y Estadísticas:

- Se consideraron las estadísticas básicas como promedios y medianas, particularmente en la columna Age, para determinar la mejor estrategia de imputación.

6. Transformación de los Datos

Imputación de Edad:

- Los valores faltantes de la columna Age se imputaron utilizando la mediana de grupos definidos por la clase del boleto (Pclass) y el sexo (Sex). Este enfoque permitió una imputación más informada, reduciendo el sesgo en los datos.

Eliminación de Columnas:

- Se eliminaron las columnas Cabin, Ticket, PassengerId y Fare debido a su alta proporción de valores faltantes o baja relevancia para el modelo predictivo.

Conversión de Categóricos:

- La columna Sex se transformó en una variable binaria (Sex bool), donde 0 representa a los hombres y 1 a las mujeres.
- Las columnas categóricas como Embarked fueron codificadas usando pd.get_dummies para convertirlas en variables numéricas más adecuadas para el modelo.

7. Limpieza de los Datos

Eliminación de Instancias con Valores Faltantes:

- En un primer enfoque, se eliminaron todas las instancias con valores faltantes en la columna Age, resultando en una reducción del número total de registros en el dataset.

Imputación de Valores Faltantes:

- En un segundo enfoque, se imputaron los valores faltantes en la columna Age, conservando así el total de instancias y evaluando si esto mejora la precisión del modelo.

Estado Final del Dataset:

- Después de la limpieza y transformación, el dataset final de entrenamiento tiene un número reducido de columnas y, dependiendo del enfoque, un número diferente de registros.

8. Modelado

Se probaron diferentes modelos de aprendizaje automático, como regresión logística y árboles de decisión, en ambos enfoques (eliminación de valores faltantes e imputación). Los resultados de precisión se compararon para seleccionar el mejor modelo a utilizar en la predicción final.

Extracción de Datos

```
print(df_gender.shape)
df_gender.head()
```

```
(418, 2)
```

	PassengerId	Survived
0	892	0
1	893	1
2	894	0
3	895	0
4	896	1

Tabla 1: En esta es la tabla del archivo gender_submission.csv y se enseña por cuantas columnas y filas está hecha

```
print(df_train.shape)
df_train.head()
```

```
(891, 12)
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

Tabla 2: En esta es la tabla del archivo train.csv y se enseña por cuantas columnas y filas está hecha

```
print(df_test.shape)
df_test.head()
```

```
(418, 11)
```

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	Q
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	S
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	Q
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	S
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	S

Tabla 3: En esta es la tabla del archivo test.csv y se enseña por cuantas columnas y filas está hecha

Lo que se hizo primero fue leer los archivos que se descargaron del reto del Titanic y luego de entender porque información estaba compuesta, se empezó a hacer el análisis correspondiente para saber qué datos nos podría servir para el modelo. La primera forma en la que limpiamos los datos fue ver cuáles eran las columnas que más datos vacíos tenían (NaN).

```
count_na = df_train.isna().sum()
print('Cantidad de valores NaN en datos de entrenamiento:\n', count_na)
```

```
Cantidad de valores NaN en datos de entrenamiento:
 PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

Tabla 4: Se muestran todos los datos vacíos (NaN), que tiene cada columna del archivo train.csv

En la tabla se observó que los datos de la cabina y de la edad son los que presentan más información faltante. En el caso de la cabina son 687 datos faltantes de 891 datos, luego de realizar una sencilla regla de tres, nos percatamos de que cerca del 77% de los datos estaban ausentes. Debido a esta alta proporción de datos faltantes, decidimos eliminar esa columna.

En la columna de edad, hay 177 valores faltantes de un total de 891, lo que equivale al 19.86% de los datos en esa columna. Se consideraron dos cursos de acción para

abordar este problema: imputación de datos o eliminación. Tras evaluar el modelo en ambos escenarios, se escogerá el que tenga mejor precisión. Para la imputación, utilizamos la mediana, agrupando los datos por clase y sexo, y así obtener la mediana de cada grupo. Esta se usó para llenar los datos faltantes, con cerca de seis combinaciones diferentes. Este método se consideró adecuado debido a la menor proporción de información faltante en esta columna.

Además, identificamos que hay información que no aporta mucho al modelo, como el PassengerID, que sólo indica cómo se almacenó al pasajero en la base de datos, pero no ofrece información relevante para el modelo, ya que los pasajeros ya están ordenados con un número respectivo. Otro dato eliminado fue el Ticket, ya que no proporciona información sobre la supervivencia de los pasajeros. El último dato que se eliminó fue el Fare, dado que tampoco brinda información relevante sobre los pasajeros que sobrevivieron o no al accidente del Titanic.

```
[ ] df_train = df_train.drop(['Ticket', 'Cabin', 'PassengerId', 'Fare'], axis=1)
```

```
print(df_train.shape)
df_train.head()
```

```
(891, 8)
```

	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Embarked
0	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	S
1	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	C
2	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	S
3	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	S
4	0	3	Allen, Mr. William Henry	male	35.0	0	0	S

Tabla 5: Se muestran los datos luego de la limpieza en el archivo train.cvs y test.csv

Una pequeña modificación que se hizo al archivo fue el hecho de cambiar los géneros por valores booleanos, esto para que sea más sencillo de analizar para el modelo que queremos crear, por lo que hicimos una nueva asignación donde los valores de 0 es para el sexo masculino, mientras que los valores de 1 son para las mujeres y eliminamos la columna de sexo normal, de esta manera la mayoría de datos que analiza el sistema son númericos. Luego de todas las modificaciones la tabla final queda de la siguiente manera

```
df_train.head()
```

	Survived	Pclass	Name	Age	SibSp	Parch	Embarked	Sex bool
0	0	3	Braund, Mr. Owen Harris	22.0	1	0	S	0
1	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	38.0	1	0	C	1
2	1	3	Heikkinen, Miss. Laina	26.0	0	0	S	1
3	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	35.0	1	0	S	1
4	0	3	Allen, Mr. William Henry	35.0	0	0	S	0

Tabla 6: En esta tabla se ven los datos finales y sin nada redundante en los archivos de train.csv y test.csv

Procesamiento de datos :

Como muchos algoritmos de aprendizaje automático no pueden trabajar directamente con variables categóricas porque esperan que las entradas sean numéricas. One-Hot Encoding convierte estas variables en un formato adecuado. Y a diferencia de asignar números enteros a las categorías (por ejemplo, "S" = 1, "Q" = 2, "C" = 3), el One-Hot Encoding evita que el modelo asuma que hay algún orden o jerarquía entre las categorías. En nuestro ejemplo la columna de ‘Embarked’ clasifica nuestras instancias en 3 categorías dependiendo de donde partió la persona dichos como: S, C o Q. Decidimos utilizar One-Hot Encoding en este feature por su practicidad y su facilidad de procesamiento de datos.

```
[40] df_train_encoded = pd.get_dummies(df_train, columns=['Embarked'], dtype=int)
     df_test_encoded = pd.get_dummies(df_test, columns=['Embarked'], dtype=int)
     df_test_encoded
```

	Pclass	Name	Sex	Age	SibSp	Parch	Sex bool	Survived	Embarked_C	Embarked_Q	Embarked_S
0	3	Kelly, Mr. James	male	34.5	0	0	0	0	0	1	0
1	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	1	0	0	0	1
2	2	Myles, Mr. Thomas Francis	male	62.0	0	0	0	0	0	1	0
3	3	Wirz, Mr. Albert	male	27.0	0	0	0	0	0	0	1
4	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	1	0	0	0	1
...	...	...	...	...	...	...	...	...	...	...	...
413	3	Spector, Mr. Woolf	male	24.0	0	0	0	0	0	0	1
414	1	Oliva y Ocana, Dona. Fermina	female	39.0	0	0	1	0	1	0	0
415	3	Saether, Mr. Simon Sivertsen	male	38.5	0	0	0	0	0	0	1
416	3	Ware, Mr. Frederick	male	24.0	0	0	0	0	0	0	1
417	3	Peter, Master. Michael J	male	24.0	1	1	0	0	1	0	0