Reto‘Kaggle’ {

[Titanic - Machine Learning from Disaster]

< Andrea Piñeiro A01705681>,
< Antonio Galarza A00828688>,
< Carlos Contreras A01232543>,
< Eduardo Alvarado A01251534>,
< Felipe Yépez A01658002>

}

Tabla de ‘Contenidos’ {

}

01 {

[Resultados]

Accuracy: 79.66%

}

450	**Felipe Yepez**		0.79665	2	1s

Your Best Entry!
Your most recent submission scored 0.79665, which is an improvement of your previous score of 0.78947. Great job!

Tweet this

02 {

[Exploración y preparación]

}

Distribución en los datos

{

Registros

891 personas. (40% del número real)

Tarifas

Con pocos pasajeros variaron significativamente.

Survived

Variable categórica entre 0 y 1.

Pasajeros

Casi la mitad de los pasajeros viajaba acompañado (48%).

Edades

Menos del 1% de los pasajeros están entre los 65 y 80 años.

}

Datos faltantes < /df > {

< Se buscaron los datos faltantes y se calcularon los porcentajes que representan. >

< Se eliminó la variable “Cabin”
&& Se eliminaron registros sin “Embarked”. >
}

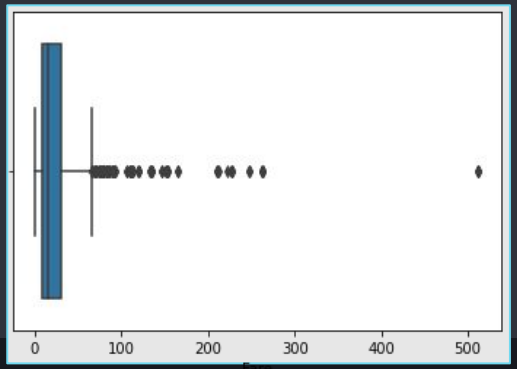
PassengerId	0.000000
Survived	0.000000
Pclass	0.000000
Name	0.000000
Sex	0.000000
Age	0.198653
SibSp	0.000000
Parch	0.000000
Ticket	0.000000
Fare	0.000000
Cabin	0.771044
Embarked	0.002245

dtype: float64

Datos atípicos < /da > {

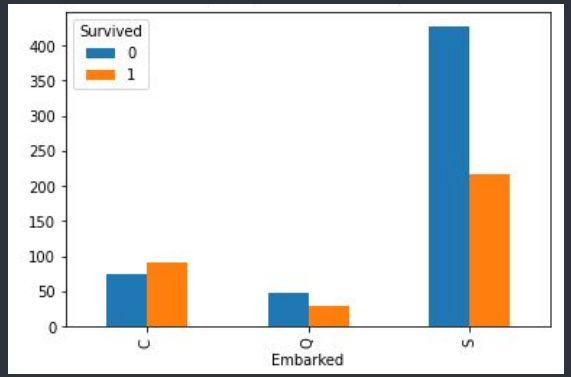
< Al analizar las todas las variables, sólo “Fare” tenía un dato atípico. >

< Se eliminaron los valores atípicos más alejados de la variable “Fare” >
}



Transformación; {

< Para su mejor aprovechamiento, a partir de la variable ‘Embarked’ se crearon 3 variables dummies: ‘C’, ‘Q’ y ‘S’ >



< Utilizando la columna ‘Name’, pudimos extraer los títulos de las personas, para utilizarlos más tarde en la predicción de edades. >

Title	
Master	40
Miss	184
Mr	536
Mrs	126

}

Transformación; {

< ¿Viaja acompañado? >

< Sexo → 0,1. >

< Clase 1,2,3 → 3,2,1. >

}

	Survived	Pclass	Name	Sex	Age	Fare	C	Q	Group
0	0	1	Braund, Mr. Owen Harris	0.0	22.0	7.2500	0	0	1.0
1	1	3	Cumings, Mrs. John Bradley (Florence Briggs Th...	1.0	38.0	71.2833	1	0	1.0
2	1	1	Heikkinen, Miss. Laina	1.0	26.0	7.9250	0	0	0.0
3	1	3	Futrelle, Mrs. Jacques Heath (Lily May Pe...)	1.0	35.0	53.1000	0	0	1.0
4	0	1	Allen, Mr. William Henry	0.0	35.0	8.0500	0	0	0.0

Predicción edades; {

< Ya que había datos faltantes en la variable ‘Age’, decidimos utilizar un modelo predictivo para obtener completitud en nuestra base. >

}

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 886 entries, 0 to 890
Data columns (total 13 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Survived  886 non-null    int64
 1   Pclass    886 non-null    int64
 2   Sex       886 non-null    float64
 3   Age       886 non-null    float64
 4   Fare      886 non-null    float64
 5   C         886 non-null    uint8
 6   Q         886 non-null    uint8
 7   S         886 non-null    uint8
 8   Group     886 non-null    float64
 9   Master    886 non-null    uint8
 10  Miss      886 non-null    uint8
 11  Mr        886 non-null    uint8
 12  Mrs       886 non-null    uint8
dtypes: float64(4), int64(2), uint8(7)
memory usage: 54.5 KB
```

Dataframe subset; {

	Pclass	Sex	Age	Fare	C	Q	S	Group	Master	Miss	Mr	Mrs
0	1	0.0	22.0	7.2500	0	0	1	1.0	0	0	1	0
1	3	1.0	38.0	71.2833	1	0	0	1.0	0	0	0	1
2	1	1.0	26.0	7.9250	0	0	1	0.0	0	1	0	0
3	3	1.0	35.0	53.1000	0	0	1	1.0	0	0	0	1
4	1	0.0	35.0	8.0500	0	0	1	0.0	0	0	1	0

}

03 {

[Análisis]

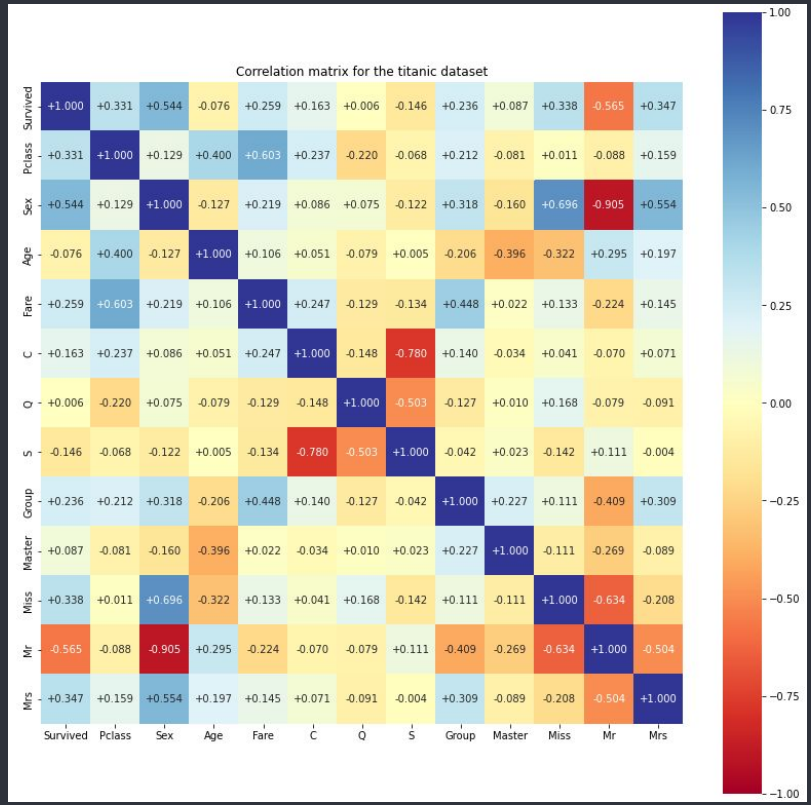
}

Correlación; {

< Como parte de nuestro análisis, se exploró la correlación entre las variables ya preparadas. >

#correlationMatrix

}

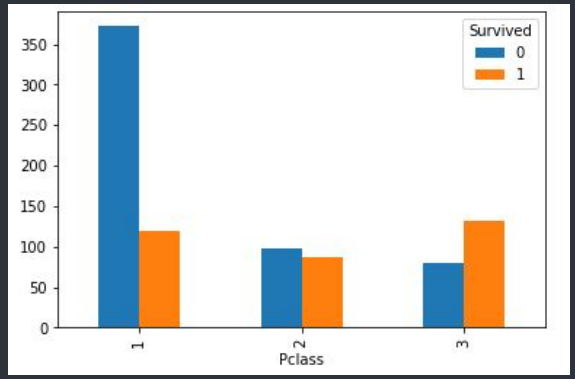


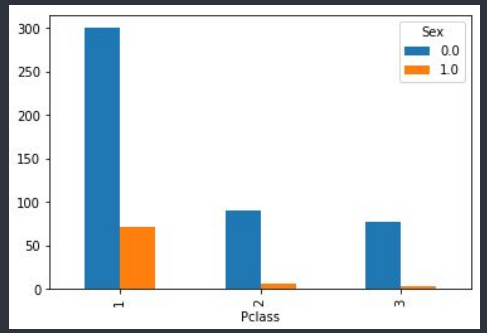
Clases; {

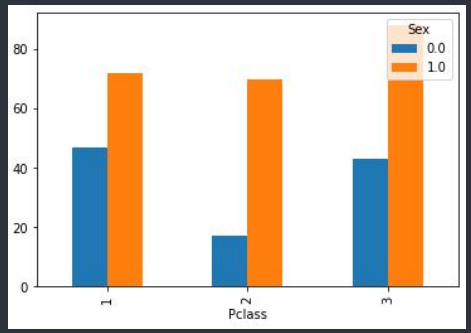
< Número de sobrevivientes y no sobrevivientes: por clases >

< Número de no sobrevivientes por sexo y clase. >

{ < Número de sobrevivientes por sexo y clase. >





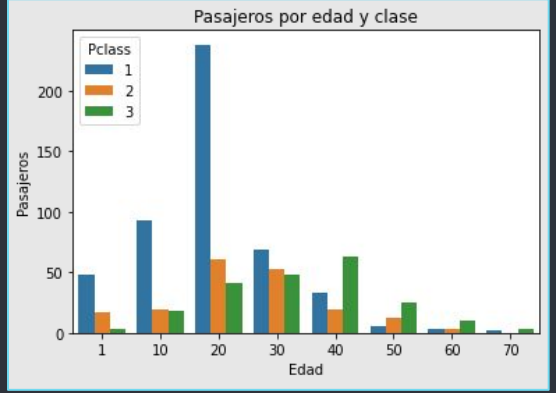


Edades; {

< Distribución de pasajeros con supervivencia por grupo de edad >

< Distribución de pasajeros y sus clases por grupo de edad >

}



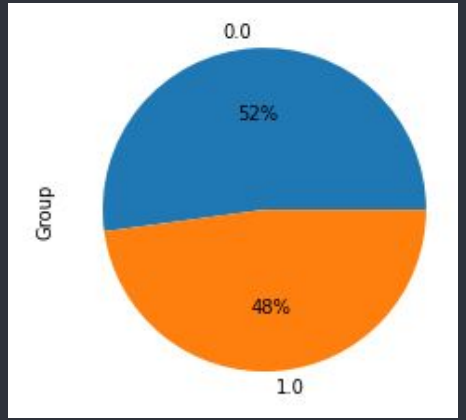
Acompañados </1>; {

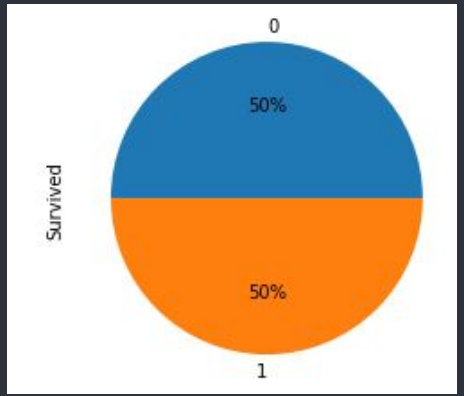
< Con la nueva variable “Group”, se pudo determinar que el 48% venía acompañado >

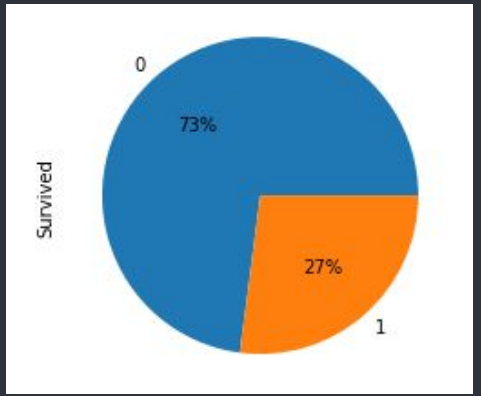
< Se observó que, de los acompañados, el 50% sobrevivió. >

< Por otro lado, los no acompañados sólo sobrevivieron en el 27% de los casos. >

}

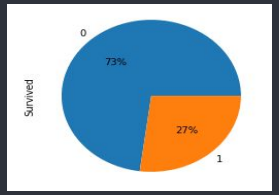


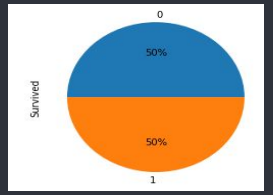




Acompañados </2>; {

Se realizó una prueba de hipótesis con la que se comprobó que el venir acompañado influye en la supervivencia.






```
Alone survivement mean value: 0.27056277056277056
Friends survivement mean value: 0.5
Alone survivement std value: 0.4447321299590394
Friends survivement std value: 0.5005906676601786
p-value 1.093814468879732e-12
Se rechaza la hipótesis nula (las medias son diferentes estadisticamente con un 95% de confianza)
```

}

04 {

[Modelo & Resultados]

}

Steps ‘Model’ {

- Step 01 Escalar los datos
- Step 02 Observar con qué variables quedarnos
- Step 03 Modelos con Scikit-Learn
- Step 04 Modelo Final con Tensorflow

}

Escalar < /datos > {

Se probó con:
- StandardScaler
- MinMaxScaler

}

Elegir < /variables >

{

< Analizamos la correlación obtenida con nuestra variable objetivo de supervivencia y detectamos que con nuestro procesamiento generamos correlaciones más altas que las iniciales. >

‘Fare’ no tiene influencia para el modelo

}

Modelos ‘Scikit-Learn’{

Elegir < /modelos >

Utilizando la librería de ‘Scikit-Learn’ tratamos con varios modelos tanto para los datos originales como los normalizados, para ver cómo se comportan modelos establecidos con los datos que procesamos.

Decision tree:

Sin normalizar		Normalizados	
Accuracy	0.741627	Accuracy	0.744019
Precision	0.650602	Precision	0.650888
Recall	0.683544	Recall	0.696203
F1 Score	0.666667	F1 Score	0.672783

}

Modelos 'Scikit-Learn' {

Random forest:

Sin normalizar		Normalizados	
Accuracy	0.758373	Accuracy	0.746411
Precision	0.691275	Precision	0.668831
Recall	0.651899	Recall	0.651899
F1 Score	0.671010	F1 Score	0.660256

SVM (Support vector machine):

Sin normalizar		Normalizados*	
Accuracy	0.777512	Accuracy	0.777512
Precision	0.748092	Precision	0.748092
Recall	0.620253	Recall	0.620253
F1 Score	0.678201	F1 Score	0.678201

}

Modelos 'Scikit-Learn' {

Logistic regression:

Sin normalizar		Normalizados	
Accuracy	0.782297	Accuracy	0.782297
Precision	0.708075	Precision	0.705521
Recall	0.721519	Recall	0.727848
F1 Score	0.714734	F1 Score	0.716511

XGB (Extreme Gradient Boosting):

Sin normalizar		Normalizados	
Accuracy	0.760766	Accuracy	0.763158
Precision	0.693333	Precision	0.697987
Recall	0.658228	Recall	0.658228
F1 Score	0.675325	F1 Score	0.677524

}

Tensorflow; {

<Hiperparámetros >

Capa Oculta:

<8 unidades>

<Relu>

<Dropout 0.1>

Capa Salida:

<Softmax>

<2 unidades>

} </Hiperparámetros>

```
model.summary()
```

```
Model: "sequential_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_8 (Dense)             (None, 8)                 104

 dropout_4 (Dropout)         (None, 8)                 0

 dense_9 (Dense)             (None, 2)                 18

=================================================================
Total params: 122
Trainable params: 122
Non-trainable params: 0
_________________________________________________________________
```

Tensorflow; {

‘MODELO FINAL’

<ACCURACY 79.66%>

Optimizer: RMSprop

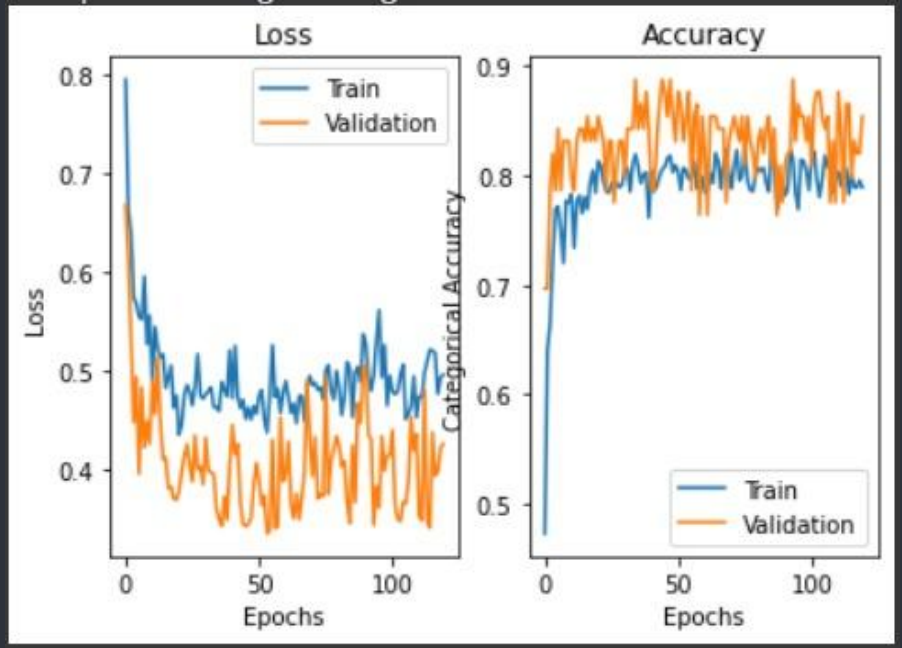
Epochs: 120

Batch_size = 32

Learning_rate = 0.001

Momentum = 0.999

}



```
Gracias {

    'Momento para Preguntas'

}
```