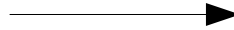


Primera Clase de Análisis de Datos

Prof: Boris Panes
Universidad Del Desarrollo

Agosto 24, 2024

Flujo de Cursos



Flujo general de Ciencia de Datos
Herramientas básicas, estimadores
centrales y dispersión
Estadísticos muestrales y test de hipótesis
Datos estructurados y sucintos

Insertión del curso en el flujo general
Revisión de herramientas más sofisticadas,
como modelos predictivos lineales y
series de tiempo
Exploración de datos mas realistas

Ciencia de Datos

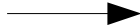
Descripción

Conjunto de herramientas y metodologías basadas en conceptos estadísticos y probabilísticos que permiten generar un marco conceptual general para el uso de datos para la toma de decisiones

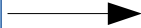
Objetivo

En el mediano a largo plazo es argumentable suponer que uno de los objetivos de la ciencia de datos es establecer un conjunto de etapas de análisis secuenciales y cíclicas que permiten la automatización del proceso de análisis de datos

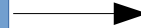
Preparación de
Datos



Análisis
Estadísticos



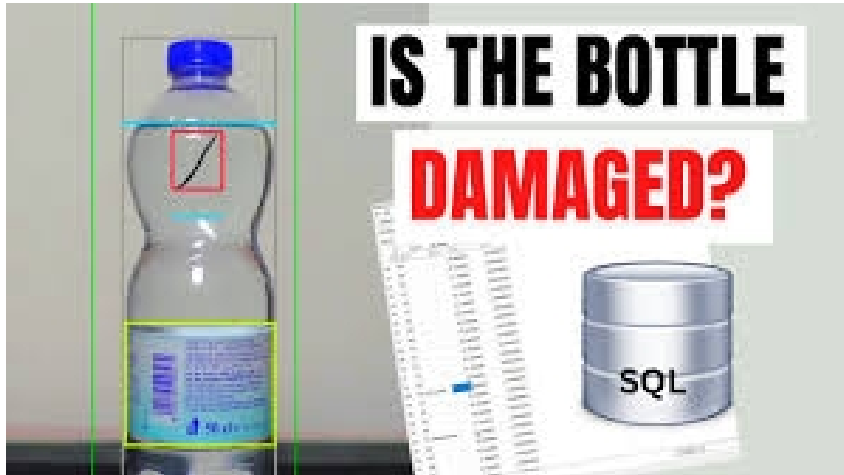
Resultados
Empíricos



Decisiones

Ventajas de la automatización

La automatización no solo mejora la eficiencia, sino que también democratiza el acceso a herramientas avanzadas, permitiendo que tanto expertos como no expertos puedan extraer valor de los datos de manera más rápida y precisa. Al dominar estas técnicas, estarán mejor preparados para enfrentar los desafíos del futuro en un mundo cada vez más impulsado por datos.



Presentación de los contenidos del curso Análisis de Datos

Perspectiva contextual dentro de la Ciencia de Datos

Calendario de Clases y Estilo del CANVAS para este curso

Referencias

Información General

Clases en Video y Presentación

Ejercicios de Selección Múltiple

Evaluaciones en Formato Grupal con Presentación

Contenido

Test de hipótesis, tamaño de la muestra, p -value, poder estadístico

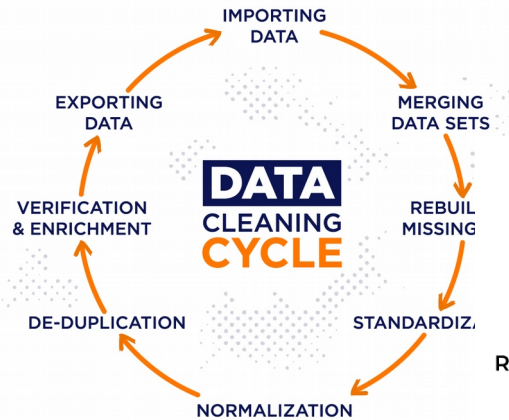
Limpieza y estructuración de Datos

Definición conceptual

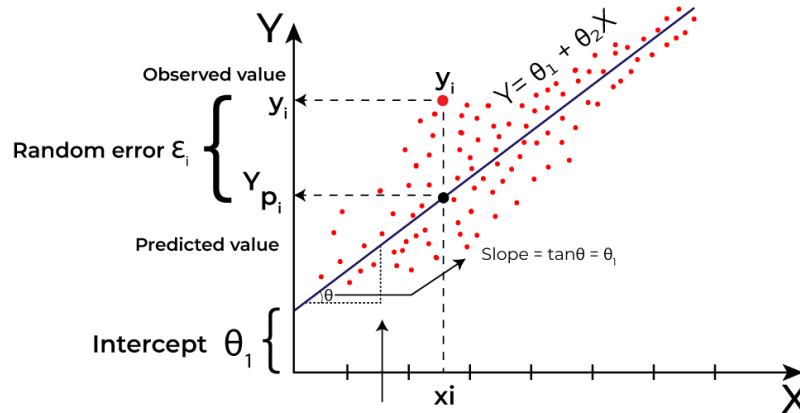
Análisis práctico

Contenidos del Curso

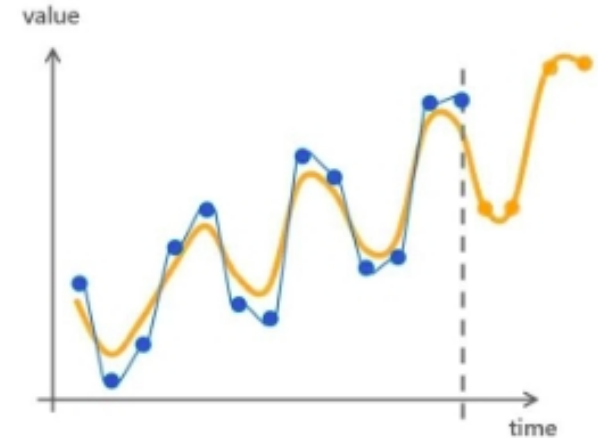
1- Limpieza y estructuración de datos



2- Regresión Lineal



3- Series Temporales



El **primer módulo de Análisis de Datos** se puede considerar en términos de bases de datos de estructura nutrida, con múltiples columnas y seguramente muchas filas.

De hecho pretendemos limpiar, estandarizar, graficar, etc. Por lo tanto necesitamos muchos datos y de preferencia reales y contingentes. Enfatizaremos en visualización e interpretación de gráficos. Datos normales vs estandarizados

El **segundo modulo** es mas teórico y como tal se puede abordar con tablas mas reducidas y mas bien acompañado de formulas matemáticas y una proliferación de conceptos complementarios

Esta dinámica aparece bastante seguido a lo largo del curso, donde hemos intentado combinar la revisión de conceptos mas teóricos durante las clases con evaluaciones basadas en conjuntos de datos como Iris y Titanic

En el **tercer modulo** veremos un caso intermedio, dado que los datos de series de tiempo puede ser reducidos en cantidad pero sutiles en estructura. Requiere casos reales y visualización explícita de datos

El método de regresión **lineal** es el **primer modelo predictivo** que vamos a considerar

Ejemplos donde se puede utilizar la **Regresión Lineal**

1) **Predicción de precios de viviendas:**

Modelos de regresión lineal se utilizan para predecir el precio de las viviendas en función de características como el tamaño, el número de habitaciones, la ubicación, etc.

Ayuda a compradores, vendedores e inmobiliarias a tomar decisiones más informadas

2) **Evaluación de riesgo crediticio:**

Bancos y entidades financieras utilizan modelos de regresión para evaluar la probabilidad de incumplimiento de los clientes en función de su historial crediticio y otros factores.

Mejora la gestión de riesgos y la toma de decisiones de crédito.

Ejemplos donde se puede utilizar **Series de Tiempo**

1- Predicción de la Demanda de Energía

Las empresas energéticas, como Enel, utilizan series temporales para predecir la demanda de electricidad a lo largo del tiempo. Analizan datos históricos de consumo eléctrico, tomando en cuenta **patrones estacionales, tendencias, y eventos externos** (como cambios climáticos). Esto permite planificar la generación y distribución de energía de manera eficiente.

Evita cortes de energía, optimiza la generación y reduce costos operativos

2- Previsión de Mercados Financieros

Inversores y firmas financieras como Goldman Sachs o JP Morgan utilizan modelos de series temporales para analizar y predecir el comportamiento de los mercados financieros. Se estudian datos históricos de precios de acciones, tasas de interés y volúmenes de transacciones para identificar tendencias, ciclos económicos y volatilidad. Esto forma la base para la toma de decisiones en inversiones.

Influye en la asignación de capital, la gestión de riesgos y las estrategias de trading

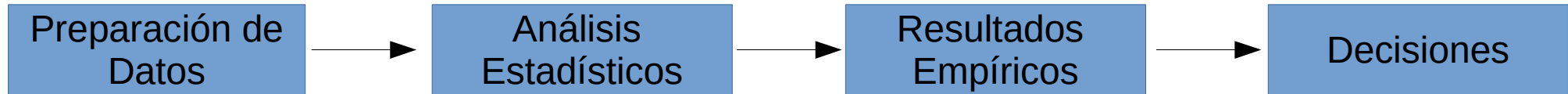
Localización de estos conceptos en el flujo
general de la Ciencia de Datos

Flujos Conceptuales Generales

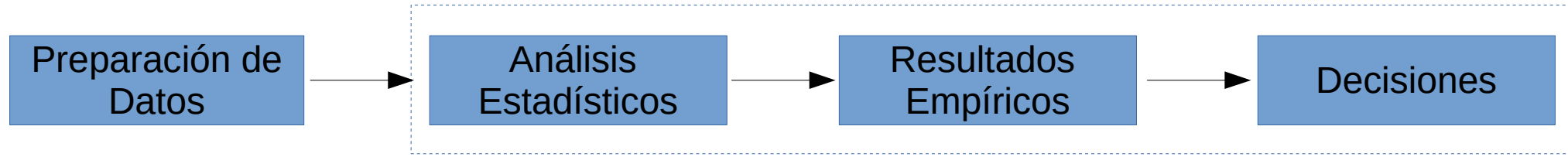
Históricamente y conceptualmente hablando, la disciplina de Ciencia de Datos es desarrollada con herramientas de Estadística y Probabilidades, así como otros tipos de Ciencias y disciplinas empíricas en general (Física, Química, Biología, Economía, Ciencias Sociales)



Cada Ciencia o disciplina empírica intenta resolver cierto tipo de problemas considerando una metodología, normalmente basada en el método científico. En Ciencia de Datos tenemos



Durante el primer curso del diplomado el énfasis está hecho en el aprendizaje de herramientas para realizar Análisis **Estadísticos (Estadística para Ciencia de Datos)**



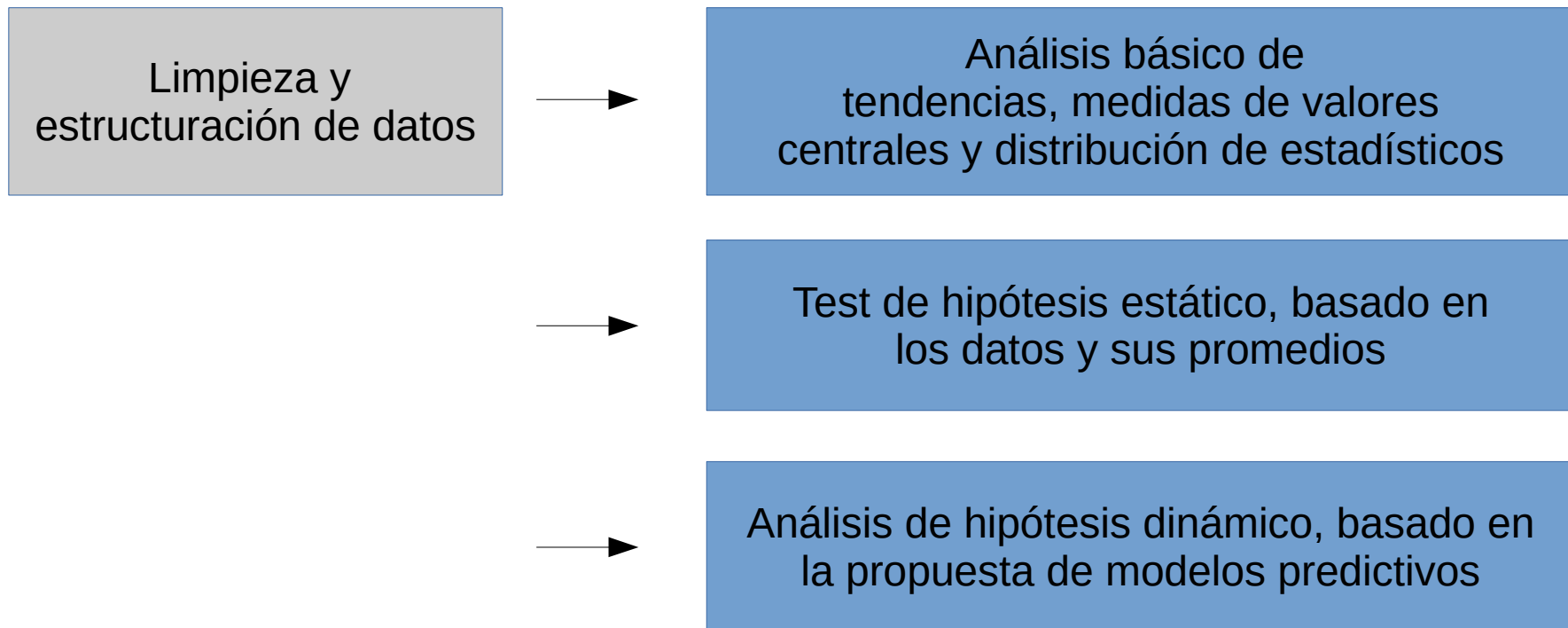
Durante el primer curso el énfasis está hecho en el aprendizaje de herramientas para realizar Análisis **Estadísticos (Estadística para Ciencia de Datos)**

Durante el segundo curso el énfasis estará en la Preparación **de los Datos**, así como en la introducción de nuevas herramientas de análisis estadísticos, como la Regresión **Lineal** y las **Series de Tiempo**

Las evaluaciones estarán enfocadas en **ejercicios prácticos de selección múltiple** considerando cada ítem revisado, pero también incluiremos **trabajos basados en el análisis estadístico de alguna base de datos en particular, realista y validada.**

El mejor ejemplo que puedo pensar por el momento es la plataforma **Kaggle**, la cual contiene múltiples desafíos basados en el análisis de datos. Muchos de los intentos en esta plataforma incluyen notebooks con desarrollos de análisis estadísticos utilizando notebooks de python. Los notebooks vienen en diferentes niveles dependiendo de la sofisticación de las herramientas utilizadas y la precisión de sus cuantificaciones

Flujo del trabajo de análisis de datos



¿Por qué durante el primer curso de Estadística para Ciencia de Datos fue posible esquivar la limpieza y estructuración de datos? simpleza, simulación, implícito (titanic)

Calendario de Clases, Evaluaciones y Canvas

Calendario y Evaluaciones

TRIM.	FECHA	HORA
TRIM.2	sábado, 24 de agosto de 2024	11.20 - 12.30 12.30 - 13.40
TRIM.2	sábado, 31 de agosto de 2024	11.20 - 12.30 12.30 - 13.40
TRIM.2	sábado, 7 de septiembre de 2024	11.20 - 12.30 12.30 - 13.40
TRIM.2	sábado, 14 de septiembre de 2024	11.20 - 12.30 12.30 - 13.40
TRIM.2	sábado, 28 de septiembre de 2024	11.20 - 12.30 12.30 - 13.40
TRIM.2	sábado, 5 de octubre de 2024	11.20 - 12.30 12.30 - 13.40
TRIM.2	sábado, 19 de octubre de 2024	11.20 - 12.30 12.30 - 13.40

Publicación T1: Limpieza y Estructura de Datos

Publicación T2: Regresión Lineal

Publicación T3: Series Temporales

Más ejercicios con múltiples alternativas

Con cuatro notas tenemos suficiente para evaluar todos los items y mantenerse en constante actividad

Estructura de los equipos de trabajo

- 1- Responsable de búsqueda de datos y descripción del objetivo usando Kaggle
- 2- Responsable de realizar el análisis de datos en Notebook de python
- 3- Responsable de armar la presentación en un PPT
- 4- Responsable de exponer el trabajo en formato video de ZOOM

Etapas del trabajo T1

- 1- Formación de equipos de trabajo y elección primer set de datos, próxima semana
- 2- Entrega de primeros test T1, tercera semana

Información General

Calendario de clases y evaluaciones

Notas parciales

Referencias

Libros, videos

Unidad 1: Limpieza y Estructura de Datos

Clases en Video y Presentación

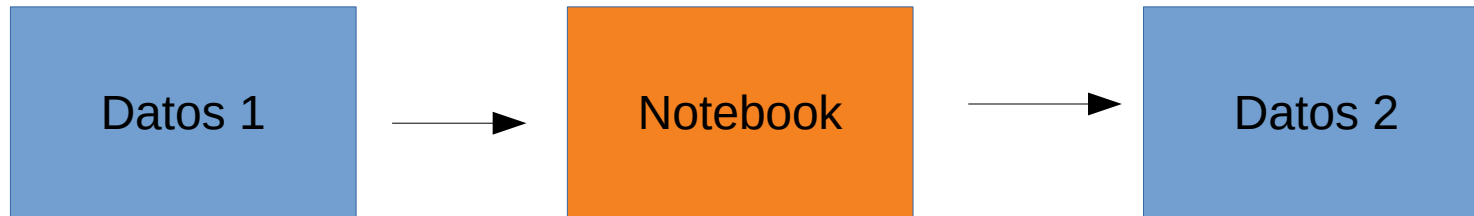
Evaluaciones de Selección Múltiple

Evaluaciones en Formato Grupal con Presentación

Unidad 2: Regresión Lineal

Unidad 3: Series Temporales

Exploración y estructuración de datos



Limpieza y estructuración de datos

Calidad de los datos

Compleitud: todas las entradas del dataset contienen todos sus campos

Consistencia de formato: consistencia y coherencia de los formatos (fechas por ejemplo)

Limpieza: eliminación de errores, duplicados o datos irrelevantes

Precisión de cada dato: exactitud y veracidad de cada dato

Representatividad: muestreo aleatorio, estratificado y por conglomerado. Durante este curso vamos a suponer que los datos considerados son representativos

Limpieza y estructuración de datos

Ejemplo práctico utilizando la base de datos del **Titanic: Aprendiendo desde el Desastre**

Aplicación directa de los siguientes conceptos

Datos Faltantes (Compleitud)

Inconsistencias en los datos (Consistencia de Formato)


Outliers (Muestras lejanas al promedio)


Datos duplicados (Precisión de cada dato)


Evaluación de los aspectos positivos y restricciones de cada concepto


Notebook time: Comienzo: Agosto 22, 2024, 18:15,
Término: Agosto 23, 2024, 22:00


Limpieza y estructuración de datos: selección del set de datos de estudio





 Create


 Home


 Competitions


 Datasets


 Models


 Code


 Discussions

 Learn


 More

 Your Work


 VIEWED


 Search

Titanic Dataset

 243

New Notebook

 Download (23 kB)






Data Card

Code (178)

Discussion (1)

Suggestions (0)

Titanic-Dataset.csv (61.19 kB)



Download


10 of 12 columns

Detail

Compact

Column

About this file

 Add Suggestion

The sinking of the Titanic is one of the most infamous shipwrecks in history.


On April 15, 1912, during her maiden voyage, the widely considered “unsinkable” RMS Titanic sank after colliding with an iceberg. Unfortunately, there weren’t enough lifeboats for everyone on board, resulting in the death of 1502 out of 2224 passengers and crew.

While there was some element of luck involved in surviving, it seems some groups of people were more likely to survive than others.

In this challenge, we ask you to build a predictive model that answers the question: “what sorts

Data Explorer

Version 1 (61.19 kB)

 Titanic-Dataset.csv

Contexto sobre el set de datos

Titanic-Dataset.csv (61.19 kB)



Detail Compact Column

10 of 12 columns ▾

About this file

The sinking of the Titanic is one of the most infamous shipwrecks in history.

On April 15, 1912, during her maiden voyage, the widely considered “unsinkable” RMS Titanic sank after colliding with an iceberg. Unfortunately, there weren’t enough lifeboats for everyone on board, resulting in the death of 1502 out of 2224 passengers and crew.

While there was some element of luck involved in surviving, it seems some groups of people were more likely to survive than others.

In this challenge, we ask you to build a predictive model that answers the question: “what sorts of people were more likely to survive?” using passenger data (ie name, age, gender, socio-economic class, etc).

Entender si existe alguna correlación entre los datos demográficos de los pasajeros y su probabilidad de sobrevivir

1502 fallecidos de 2224 pasajeros

Limpieza y estructuración de datos: visualización inicial de los datos

```
In [16]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```
In [19]: df = pd.read_csv("../datos/Titanic-Dataset.csv")
```

```
In [20]: df
```

Out[20]:

	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
...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C

```
In [28]: df["Age"].mean()
```

```
Out[28]: 29.69911764705882
```

```
In [29]: from sklearn.linear_model import LinearRegression
```

```
In [30]: # Intentar ajustar un modelo de regresión lineal sin manejar NaN
X = df[['Age']]
y = df['Fare']

# Crear el modelo
model = LinearRegression()

# Intentar ajustar el modelo (esto generará un error)
model.fit(X, y)
```

```
-----
ValueError                                Traceback (most recent call last)
--> 60                                msg_dtype if msg_dtype is not None else X.dtype)
    61                                )
    62                                # for object dtype data, we only check for NaNs (GH-13254)

ValueError: Input contains NaN, infinity or a value too large for dtype('float64').
```

La función `mean()` por defecto ignora la presencia de valores NaN, lo que puede ser confuso con respecto al número de eventos N, pero se puede sobrellevar

Sin embargo, cuando intentemos hacer una regresión lineal obtendremos un **ValueError**

¿Qué hacemos con los valores NaN?

Primero, intentemos explorar todas las columnas

```
In [32]: print(df.info())
print("****40)

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 13 columns):
#   Column          Non-Null Count  Dtype
---  -
0   PassengerId     891 non-null   int64
1   Survived        891 non-null   int64
2   Pclass          891 non-null   int64
3   Name            891 non-null   object
4   Sex             891 non-null   object
5   Age             714 non-null   float64
6   SibSp           891 non-null   int64
7   Parch           891 non-null   int64
8   Ticket          891 non-null   object
9   Fare            891 non-null   float64
10  Cabin           204 non-null   object
11  Embarked        889 non-null   object
12  Age_imputed     891 non-null   float64
dtypes: float64(3), int64(5), object(5)
memory usage: 90.6+ KB
None
****40)
```

Podemos notar que Age, Cabin y Embarked tienen valores nulos

En particular, podemos notar que **Cabin** contiene valores nulos cerca de un 80% de las veces

Esta columna no entrega información relevante de la mayoría de los datos podemos borrar esta columna por el momento

Al borrar la columna Cabin hemos hecho una limpieza del set de datos, porque ahora contiene menos valores nulos

```
In [36]: df.drop('Cabin', axis=1, inplace=True)
```

```
In [37]: df
```

```
Out[37]:
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embarked	Age_imputed
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	S	22.0
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C	38.0
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	S	26.0
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	S	35.0
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	S	35.0
...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	S	27.0
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	S	19.0
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	S	28.0
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C	26.0
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	Q	32.0

891 rows × 12 columns

Las variables Age y Embarked tienen un numero minoritario de valores nulos

$$\text{Age} = 1 - 714/891 = 0.2$$

$$\text{Embarked} = 1 - 889/891 = 0.002$$

En general se plantea que en estos casos se procede con un **reemplazo utilizando valores centrales como la media o la moda**

Ventajas: se mantiene el tamaño de la muestra, lo cual es especialmente útil si estamos trabajando con set de datos pequeños

Bajo la suposición que los datos faltantes son aleatorios, la inclusión de estos eventos no debería modificar el calculo de valores centrales de las variables afectadas. Por esta razón tiene sentido utilizar el valor de la media como valor para el reemplazo

Cuando el numero de casos que debemos retirar es alto (20% o más) y por alguna razón los datos faltantes no son aleatorios, **el cálculo de estimadores de los otros parámetros podría** incurrir en un sesgo con respecto a una muestra representativa

Al final del día, la estrategia utilizada debe ser contrastada con el objetivo del análisis

Luego del reemplazo de los valores nulos repetimos el conteo para verificar

```
In [42]: df.fillna({'Age': df['Age'].mean()}, inplace=True)  
df.fillna({'Embarked': df['Embarked'].mode()[0]}, inplace=True)
```

```
In [43]: print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 891 entries, 0 to 890  
Data columns (total 12 columns):  
#   Column          Non-Null Count  Dtype  
---  -  
0   PassengerId     891 non-null   int64  
1   Survived        891 non-null   int64  
2   Pclass         891 non-null   int64  
3   Name            891 non-null   object  
4   Sex             891 non-null   object  
5   Age             891 non-null   float64  
6   SibSp           891 non-null   int64  
7   Parch          891 non-null   int64  
8   Ticket          891 non-null   object  
9   Fare            891 non-null   float64  
10  Embarked        891 non-null   object  
11  Age imputed     891 non-null   float64  
dtypes: float64(3), int64(5), object(4)  
memory usage: 83.7+ KB  
None
```

En este punto aun somos agnósticos sobre cuales pueden ser los factores que están mas o menos relacionados con la sobrevivencia

Por ese motivo es una buena práctica mantener la mayor cantidad de datos posible

Para cuantificar las correlaciones entre las variables necesitamos identificar cuales son los datos numéricos (correlación de pearson) y categóricos (boxplots)

```
In [45]: numerical_variables = df.dtypes[df.dtypes != 'object'].index
print('The number of numerical features is: ', len(numerical_variables))
print('The numerical features are:', numerical_variables)
```

```
The number of numerical features is: 8
The numerical features are: Index(['PassengerId', 'Survived', 'Pclass', 'Age', 'SibSp', 'Parch', 'Fare',
                                  'Age_imputed'],
                                  dtype='object')
```

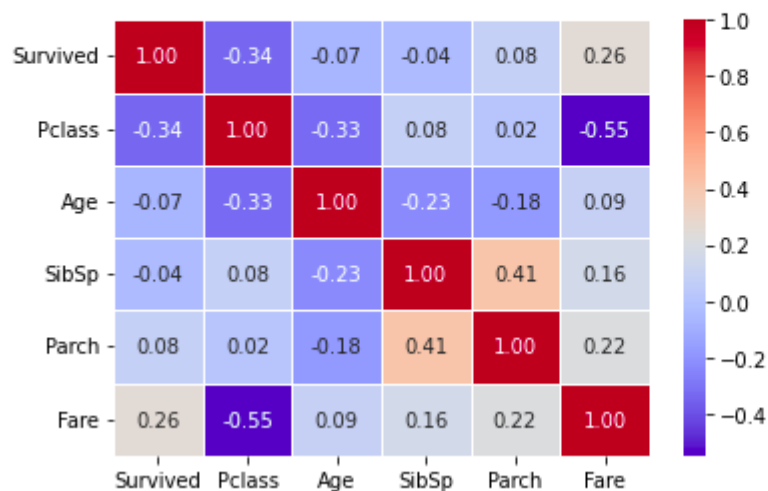
```
In [46]: categorical_variables = df.dtypes[df.dtypes == 'object'].index
print('The number of categorical features is: ', len(categorical_variables))
print('The categorical features are: ', categorical_variables)
```

```
The number of categorical features is: 4
The categorical features are: Index(['Name', 'Sex', 'Ticket', 'Embarked'], dtype='object')
```

La variable **Survived** aparece como numérica, por lo tanto nos permite calcular directamente coeficientes de correlación con las otras variables

Correlaciones entre las variables numéricas

```
In [66]: # Remove 'PassengerId' from the list of numerical variables
numerical_variables = [col for col in numerical_variables if col != 'PassengerId']
matrix_corr = df[numerical_variables].corr()
sns.heatmap(matrix_corr, annot=True, fmt=".2f", cmap="coolwarm", linewidths=0.5);
```



Pclass y Fare tiene una fuerte correlación negativa (-0.55)

SibSp y Parch tienen una correlación positiva importante (0.46)

Survived y PClass tienen una correlación negativa (-0.34)

El resto de las variables presentan un menor nivel de correlación

Analicemos como se comportan estas variables respecto a la tasa de supervivencia

```
In [67]: survival_by_class = df.groupby(['Pclass'])['Survived'].agg(['sum', 'size'])

# Rename columns for clarity
survival_by_class.columns = ['Survived', 'Total']

# Calculate survival rate
survival_by_class['Survival Rate'] = survival_by_class['Survived'] / survival_by_class['Total']

survival_by_class
```

Out[67]:

	Survived	Total	Survival Rate
Pclass			
1	136	216	0.629630
2	87	184	0.472826
3	119	491	0.242363

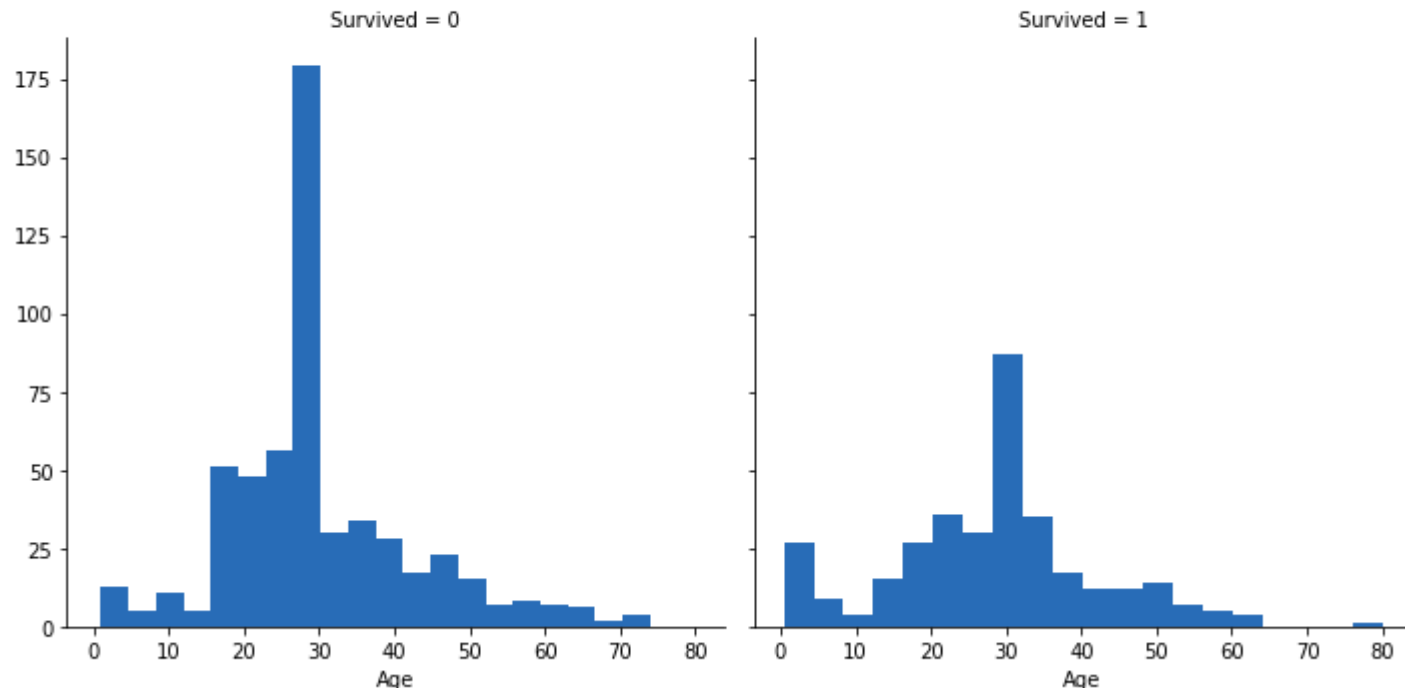
test de chi2?

el modelo nulo tendría
Survival Rate = 0.38

La correlación mostrada anteriormente
es consistente con los valores de esta
tabla

Mientras mas alto el valor de Pclass
(ticket mas económico) menor es
la tasa de supervivencia

```
In [70]: g = sns.FacetGrid(df, col='Survived', height=5)  
g.map(plt.hist, 'Age', bins=20);
```



La variable Age parece distribuir de manera similar entre Survived = 0 y 1. Con una alta concentración en torno a una edad media de 25 años

Podemos proyectar que un análisis tipo **test AB** entregaría un **p-value alto**, dado que la distancia entre las medias es muy pequeña

Análisis de variables categóricas

```
In [76]: categorical_variables
```

```
Out[76]: Index(['Name', 'Sex', 'Ticket', 'Embarked'], dtype='object')
```

```
In [73]: df.Sex.value_counts(normalize=True)
```

```
Out[73]: male      0.647587  
female    0.352413  
Name: Sex, dtype: float64
```

```
In [75]: survival_by_gender = df.groupby(df.Sex)['Survived'].agg(['sum', 'size'])  
  
# Rename columns for clarity  
survival_by_gender.columns = ['Survived', 'Total']  
  
# Calculate survival rate  
survival_by_gender['Survival Rate'] = survival_by_gender['Survived'] / survival_by_gender['Total']  
  
survival_by_gender
```

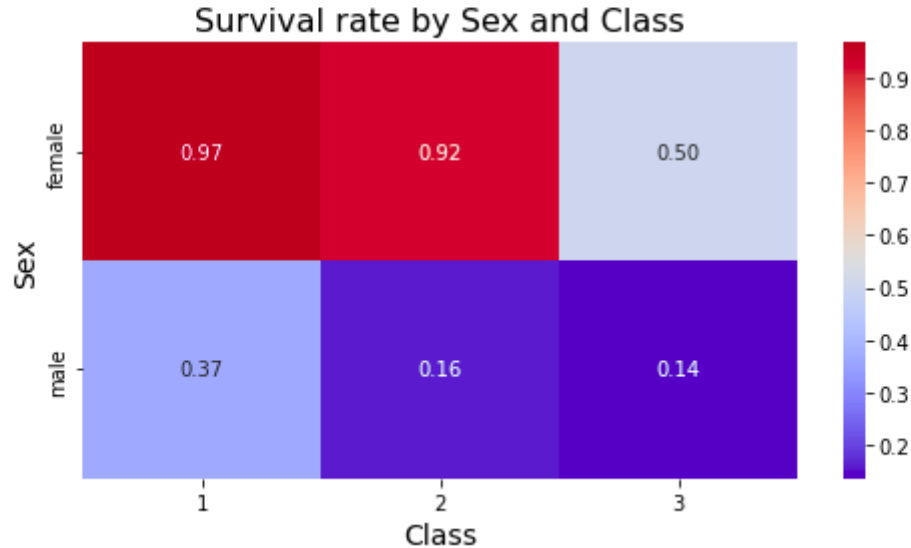
```
Out[75]:
```

	Survived	Total	Survival Rate
Sex			
female	233	314	0.742038
male	109	577	0.188908

Female representan un 35% de la población total y un 74% sobrevive al accidente

Que tal si juntamos las variables mas representativas hasta el momento (Sex y PClass)

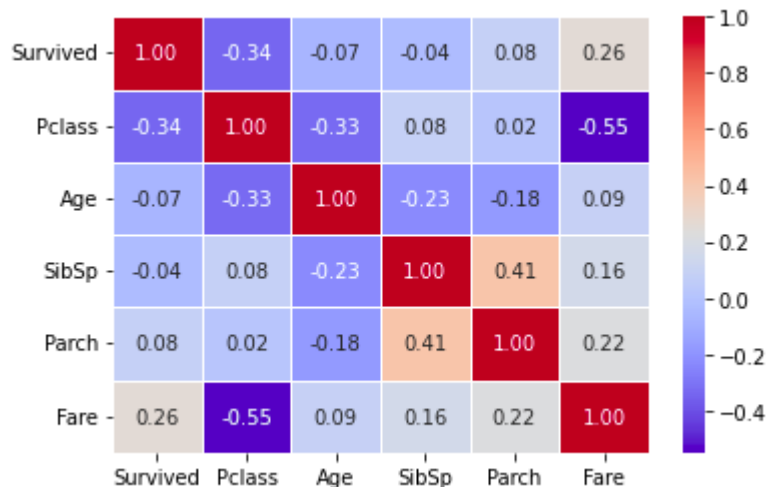
```
In [78]: sex_class_survival = df.groupby(['Sex', 'Pclass'])['Survived'].mean().unstack()
# plot
plt.figure(figsize=(8,4))
sns.heatmap(sex_class_survival, annot=True, fmt='.2f', cmap='coolwarm')
plt.title('Survival rate by Sex and Class', fontsize=16)
plt.ylabel('Sex', fontsize=14)
plt.xlabel('Class', fontsize=14)
plt.show()
```



Podemos ver que la combinación entre PClass=1 y Sex=Female muestra la tasa mas alta de sobrevivencia con un 97%

Por otro lado, PClass=3 y Sex=Male obtiene la menor tasa de sobrevivencia con 14%

Pasando de dos columnas a una sola



Existe una moderada correlación positiva entre las variables SibSp y Parch, donde

SibSp = hermanos y parejas abordo
Parch = padres e hijos

SibSp + Parch = Relatives

Este proceso en general se denomina como ingeniería de datos, dado que se esta creando un nuevo tipo de característica a partir de los tipos nativos

```
In [28]: df["Familysize"] = df["SibSp"] + df['Parch']
```

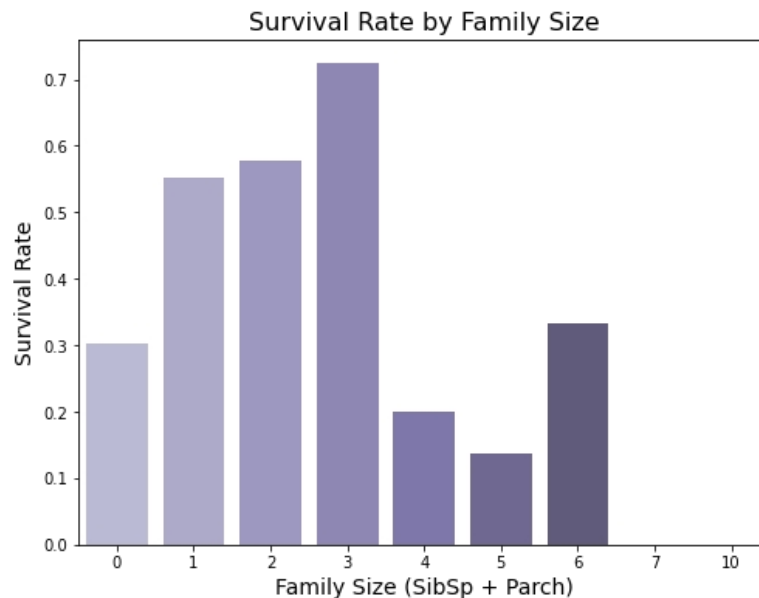
```
In [29]: family_surr = df.groupby('Familysize')['Survived'].mean()
```

```
In [30]: family_surr
```

```
Out[30]: Familysize
0      0.303538
1      0.552795
2      0.578431
3      0.724138
4      0.200000
5      0.136364
6      0.333333
7      0.000000
10     0.000000
Name: Survived, dtype: float64
```

```
In [31]: #plot
plt.figure(figsize=(8,6))
sns.barplot(x=family_surr.index, y=family_surr.values,
            palette='Purples_d')
plt.title('Survival Rate by Family Size', fontsize=16)
plt.xlabel('Family Size (SibSp + Parch)', fontsize=14)
plt.ylabel('Survival Rate', fontsize=14)
plt.show()
```

Apoyo más facilidad de movimiento:
Los pasajeros con familias medianas presentan una mayor probabilidad de sobrevivir



Conclusiones

Durante la etapa inicial de un análisis es necesario poner atención en la calidad de los datos

Muchas veces nos encontraremos con valores nulos o indefinidos, la decisión sobre como proceder depende de la situación en particular

En particular, en el contexto del análisis del dataset Titanic de Kaggle decidimos eliminar la columna Cabin por tener una mayoría de valores nulos. Dado que las columnas Age y Embarked contienen una minoría de valores nulos procedemos solo a reemplazar estos valores por la media y la moda cuando corresponde

A partir de un análisis simple de las correlaciones entre las variables, tablas de contingencia e histogramas logramos concluir que la mayor tasa de sobrevivencia se encuentra para los pasajeros con Sex=Female y PClass=1, mientras que los que presentan menos tasa de sobrevivencia están los pasajeros con Sex=Male y PClass=3

El final de la película es consistente estadísticamente hablando