

Crias de Dados

Membros: Leão Pereira Neto:22200115 / Riam Martinelli:22205569 Igor de Souza dos Santos:22203270 / Hézio Silva dos Santos:22204407

Titanic

Fizemos maior utilização da biblioteca Matplotlib.

```
In [2]: import pandas as pd
import numpy as np

df=pd.read_csv("train.csv",sep=",")
df
```

Out[2]:

	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
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows x 12 columns

```
In [10]: import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

```
In [15]: train_df = pd.read_csv("train.csv",sep=",")
```

```
In [16]: print(train_df.columns.values)

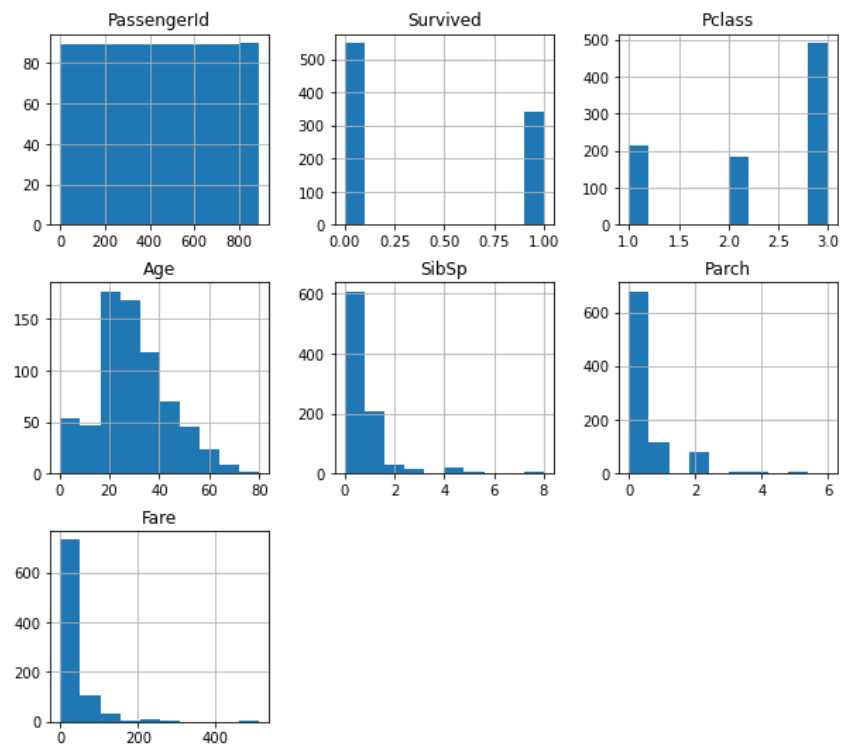
['PassengerId' 'Survived' 'Pclass' 'Name' 'Sex' 'Age' 'SibSp' 'Parch'
'Ticket' 'Fare' 'Cabin' 'Embarked']
```

```
In [20]: train_df.describe()
```

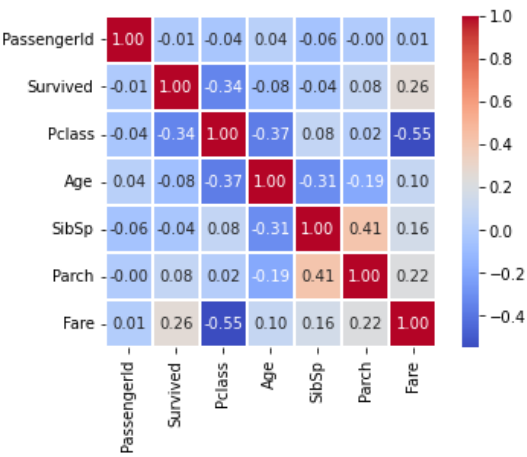
Out[20]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200

```
In [28]: train_df.hist(figsize=(10,9));
```



```
In [31]: sns.heatmap(train_df.corr(), cmap='coolwarm', fmt='.2f', linewidths=0.1,vmax=1.0,
square=True, linecolor='white', annot=True);
```



Predição com base de dados no enem 2019

Fizemos maior utilização da biblioteca NumPy

Descrição da idade do nosso banco de dados e o tipo

```
In [9]: idades_enem_2019.describe()

Out[9]: count    127380.000000
mean       22.067051
std        7.433180
min         2.000000
25%        18.000000
50%        19.000000
75%        24.000000
max        85.000000
Name: NU_IDADE, dtype: float64
```

Encontrando a média de idade por município

```
In [10]: media_idade_municipio = pd.DataFrame(dados_enem2019.groupby(['NO_MUNICIPIO_RESIDENCIA']).mean()['NU_IDADE'])
media_idade_municipio.loc[media_idade_municipio['NU_IDADE'] == media_idade_municipio['NU_IDADE'].max()]

Out[10]:
```

NO_MUNICIPIO_RESIDENCIA	NU_IDADE
Maria Helena	55.0

Quais municípios têm a média mais baixa

```
In [11]: media_idade_municipio.loc[media_idade_municipio['NU_IDADE'] == media_idade_municipio['NU_IDADE'].min()]

Out[11]:
```

NO_MUNICIPIO_RESIDENCIA	NU_IDADE
Abreulândia	15.0
Diamante do Sul	15.0
Fagundes Varela	15.0
Flor do Sertão	15.0
Gouvelândia	15.0
Itaipava do Grajaú	15.0
Jaborá	15.0
Lajeado Grande	15.0
Mansidão	15.0
Neves Paulista	15.0
Nova Crixás	15.0
Rio Branco do Ivaí	15.0
Rio Sono	15.0
Santa Rita do Novo Destino	15.0
Tejupá	15.0