Senac

Fecomércio Sesc

Data Science – Princípios e Técnicas

Março

2024

Data Science

Onde me encontrar:

https://www.linkedin.com/in/marco-mialaret-junior/

e

https://github.com/MatmJr

Vimos na aula passada

Data Science

O Ciclo de Vida da Ciência de Dados envolve várias etapas, desde a limpeza de dados até a modelagem e avaliação, para extrair insights e previsões úteis para objetivos comerciais.

As principais etapas são:

Data Science

Vamos voltar para o dataset do titanic

Data Science

Estamos trabalhando com 1309 observações de 12 variáveis. Para tornar as coisas um pouco mais explícitas, já que alguns dos nomes das variáveis não são totalmente claros, aqui está o que temos que lidar:

Data Science

Nome da Variável	Descrição
Survived	Sobreviveu (1) ou morreu (0)
Pclass	Classe do passageiro (1 = Upper Class, 2 = Middle Class, 3 = Lower Class)
Name	Nome do passageiro
Sex	Sexo do passageiro
Age	Idade do passageiro
SibSp	Número de irmãos/cônjuges a bordo
Parch	Número de pais/filhos a bordo
Ticket	Número do bilhete
Fare	Tarifa
Cabin	Cabine
Embarked	Porto de embarque C = Cherbourg, Q = Queenstown, S = Southampton

Data Science

```
# Criação do dataFrame dos alunos
# ID do arquivo no Google Drive
file_id = '1S5Nl793vcL5ZPTGjzKaIEbwbLaDplvIP'

# URL modificada para forçar o download do arquivo
url = f"https://drive.google.com/uc?id={file_id}"

# Tentando obter o arquivo com requests
try:
    response = requests.get(url)
    response.raise_for_status()  # Lança um erro para respostas não-sucedidas
    # Usando StringIO para converter o texto em um arquivo em memória e, então, lendo com o Pandas
    csv_raw = StringIO(response.text)
    data = pd.read_csv(csv_raw)
except requests.RequestException as e:
    print(f"Erro ao acessar o arquivo: {e}")
```

Data Science

Obter informações sobre o dataset:

```
> data.info()
```

Data Science

Estabelecer um atributo como índice:

```
> data = data.set_index("PassengerId")
```

Data Science

Os nomes completos dos passageiros por si só podem não fornecer informações úteis para análise, porém, os títulos presentes antes dos nomes podem ser valiosos. Estes títulos, que sempre terminam com um ponto final ('.'), oferecem pistas sobre o status social ou a classe dos indivíduos a bordo.

Data Science

Identificar e analisar esses títulos pode nos ajudar a entender as relações sociais existentes entre os passageiros e potencialmente correlacioná-las com outras variáveis, como taxas de sobrevivência.

Data Science

```
import re

def extract_title(name):
    title_search = re.search(' ([A-Za-z]+)\.', name)
    if title_search:
        return title_search.group(1)
    return ""

# Aplicando a função na coluna de nomes
data['Title'] = data['Name'].apply(extract_title)
data.head()
```

Tratando os Valores ausentes

Data Science

Vamos checar se existem valores ausentes ou NaN no dataset

```
> data.isnull().sum()

> data.isna().sum()

> data.duplicated()
```

Data Science

Vamos checar se existem valores ausentes ou NaN no dataset

```
> data.isnull().sum()

> data.isna().sum()

> data.duplicated()
```

Data Science

A quantidade de valores ausentes em Idade, Embarque e Tarifa é pequena em comparação com a amostra total, mas cerca de 80% das informações da Cabine estão ausentes. Os valores ausentes em Idade, Embarque e Tarifa podem ser preenchidos com medidas estatísticas descritivas, mas isso não funcionaria para Cabine.

Data Science

Valores ausentes na variável Idade são comumente preenchidos utilizando a mediana. Contudo, aplicar a mediana de todo o conjunto de dados pode não ser a abordagem mais eficaz, pois não considera variações dentro dos grupos de dados. Para refinar essa estimativa, é recomendável utilizar a estratificação por grupos relevantes, como as classes de passageiros (Pclass).

Data Science

A estratificação consiste em dividir o conjunto de dados em subgrupos mais homogêneos antes de calcular medidas estatísticas, como a mediana, para preenchimento de dados ausentes. Isso é especialmente útil quando há uma correlação significativa entre a variável com dados ausentes e a variável usada para estratificar.

Data Science

Agrupando as por sexo e classe:

```
> data.groupby(['Sex', 'Pclass'])['PassengerId'].count()
```

Data Science

Para encontrar a idade mediana por sexo e classe:

```
> data.groupby(['Sex', 'Pclass'])['Age'].median()
```

Data Science

Susbistituir os valores NaN pela mediana dos registos semelhantes.

```
> data['Age'] = data.groupby(['Sex',
'Pclass'])['Age'].transform(lambda x:
x.fillna(x.median()))
```

Data Science

Olhando outra vez os os NaN...

```
data.isna().sum()
```

```
Survived      0
Pclass        0
Name          0
Sex           0
Age           0
SibSp         0
Parch         0
Ticket        0
Fare          0
Cabin       687
Embarked      2
Title         0
dtype: int64
```

Data Science

O atributo "Embarked" só possui dois valores ausentes, vamos subistituir pela label que mais apareceu.

```
> data['Embarked'].value_counts()
```

```
data['Embarked'] = data['Embarked'].fillna('S')
```

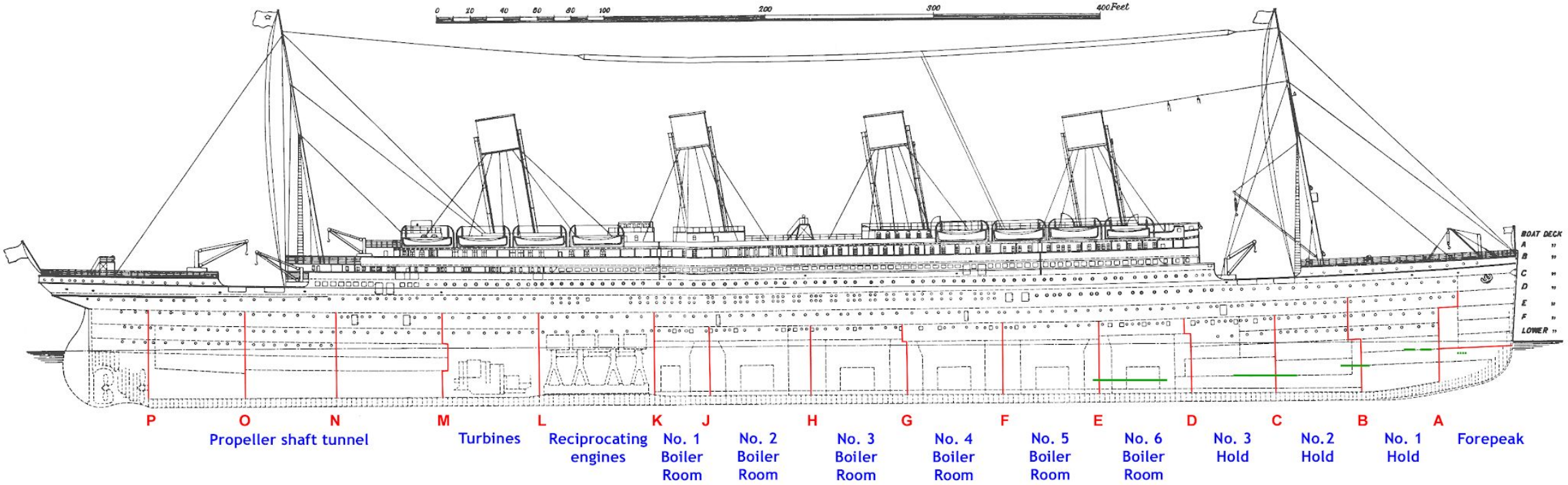
Data Science

A característica Cabine é um pouco complicada e precisa de mais exploração. Uma grande parte dos dados da Cabine está ausente e a característica em si não pode ser completamente ignorada, porque algumas cabines podem ter taxas de sobrevivência mais altas.

Data Science

Descobriu-se que a primeira letra dos valores de Cabine representa os decks onde as cabines estão localizadas. Esses decks eram principalmente separados por uma classe de passageiros, mas alguns deles eram usados por múltiplas classes de passageiros.

Data Science



Data Science

- No Convés do Barco havia 6 salas rotuladas como T, U, W, X, Y, Z, mas apenas a cabine T está presente no conjunto de dados.
- Os conveses A,B e C eram exclusivos para passageiros da 1ª classe.

Data Science

- Os conveses D e E eram para todas as classes.
- Os conveses F e G eram para passageiros da 2ª e 3ª classe.
- Ao ir de A para G, a distância até a escada aumenta, o que pode ser um fator de sobrevivência.

Data Science

```
for num in [1, 2, 3]:
    if num == 1:
        data.loc[data['Pclass'] == 1, 'Cabin'] = data.loc[data['Pclass'] == 1, 'Cabin'].fillna('ABC')
    elif num == 2:
        data.loc[data['Pclass'] == 2, 'Cabin'] = data.loc[data['Pclass'] == 2, 'Cabin'].fillna('DE')
    elif num == 3:
        data.loc[data['Pclass'] == 3, 'Cabin'] = data.loc[data['Pclass'] == 3, 'Cabin'].fillna('FG')
```

Data Science

```
data.isna().sum()
```

```
Survived    0
Pclass      0
Name        0
Sex         0
Age         0
SibSp       0
Parch       0
Ticket      0
Fare        0
Cabin       0
Embarked    0
Title       0
dtype: int64
```

Dúvidas?

Marco Mialaret, MSc

Telefone:
81 98160 7018

E-mail:
marco.junior@pe.senac.br