**Mãos à obra – Titanic – Machine Learning from Disaster**

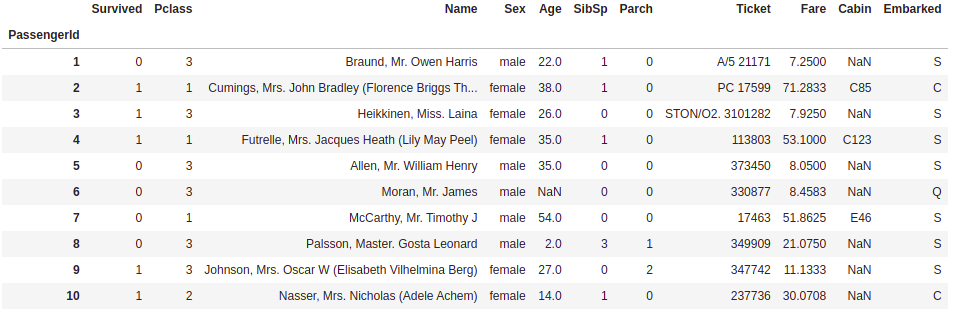
Este documento tem como objetivo principal descrever as fases realizadas ao longo do projeto, detalhando as informações obtidas a partir da exploração dos dados e relatando o passo a passo da manipulação das *features* e criação dos modelos.

**1. Análise exploratória, limpeza e manipulação dos dados**

A meta nessa fase é extrair o máximo de *insights* possíveis sobre os dados, buscando um maior entendimento sobre como se comportam de acordo com algum tipo de padrão e como se interinfluenciam.

**1.1. Visão geral**

Na figura abaixo estão as primeiras linhas do *Data Frame* do conjunto de treino, que foi carregado utilizando a coluna “*PassengerId*” como *index*. A primeira coluna contém a variável *target*, sendo que 1 indica que o passageiro sobreviveu ao desastre e 0 que veio a falecer. Os demais campos são os atributos relacionados a cada passageiro, com um total de 891 registros. Já à primeira vista, percebe-se que existem campos nulos na coluna *Cabin*. Verificando todas as colunas, também existem valores nulos para os atributos *Age* (idade) e *Embarked*, já trazendo a ideia que possivelmente serão necessárias técnicas para preenchimento.



O pandas também permite gerar um resumo estatístico básico sobre os dados, o que pode nos proporcionar um panorama geral sobre a distribuição e características dos valores.



**1.2. Correlação**

Como um problema de classificação binária, a partir da matriz de correlação é possível saber como um atributo está relacionado ao outro, a partir de um valor que varia de -1 a 1. Valores mais próximos de 1 indicam uma relação positiva (conforme um cresce o outro também), enquanto valores próximos de -1 indicam uma relação negativa (conforme um cresce o outro diminui). Já valores próximos de 0 indicam pouco relacionamento entre ambos.

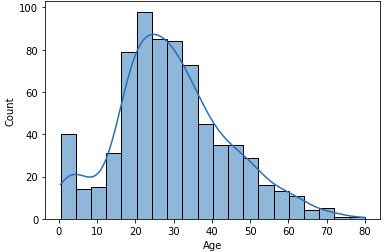


Pela matriz gerada, percebemos três correlações principais:

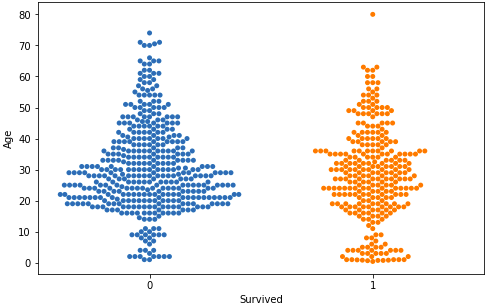
* *Pclass* e *Fare*: conforme o valor da classe aumenta o valor da tarifa tende a diminuir, o que faz sentido, já que a primeira classe tende a ser a mais cara e a terceira a mais barata;
* *Pclass* e *Age*: conforme a classe aumenta a idade do passageiro diminui, indicando que as pessoas de primeira classe tendem a ser mais velhas (mais estabelecidas financeiramente, possivelmente);
* *SibSp* e *Parch*: estes atributos são referentes à parceiros e família, portanto faz sentido que estejam relacionados, levando em conta que a viagem possivelmente era em família.

**1.3. Idade dos passageiros**

O histograma abaixo mostra a distribuição geral da idade dos passageiros. A maior parte dos passageiros possui idade entre 19 e 35 anos aproximadamente, além de uma boa quantidade de crianças com cerca de menos de 5 anos, mas com poucas pessoas acima dos 60 anos. No geral, observa-se que há uma distribuição aproximadamente normal das idades.

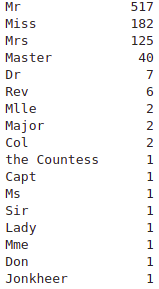


Olhando para a idade como fator de sobrevivência, a maior parte das pessoas que morreram estão justamente onde a distribuição se concentra. Porém, entre os sobreviventes já há um maior equilíbrio, indicando também que a maior parte das crianças foi salva.



**1.4. Título dos passageiros e porto de embarque**

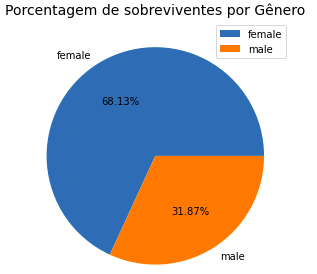
Quando olhamos para o campo nome, à primeira vista, podemos ter a impressão de que ele não é de grande importância para determinar a sobrevivência de um passageiro. Entretanto, olhando detalhadamente, percebemos que todos possuem um título relacionado, o que pode determinar a importância de cada um no momento de ser salvo. Para extrair essa informação, foi criada uma função. Com o campo criado, é interessante ver os valores únicos e a quantidade de passageiros que os possuem.



Já olhando para os portos de embarque, a maioria disparada está concentrada no “S”.

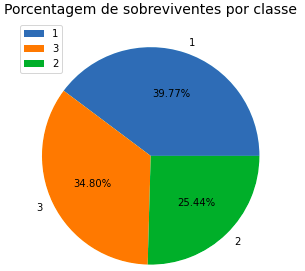
**1.5. Sobreviventes por gênero**

Olhando para a distribuição de sobreviventes por gênero, percebe-se que a maior parte foram mulheres, com uma boa diferença entre as quantidades. Porém, essa informação é ainda mais reforçada quando se calcula a proporção relativa, indicando que aproximadamente 74% das mulheres sobreviveram, enquanto para homens esse valor é de apenas 19%. Isso reforça a ideia de que durante o desastre foram priorizadas as mulheres e as crianças, o que é um procedimento padrão.



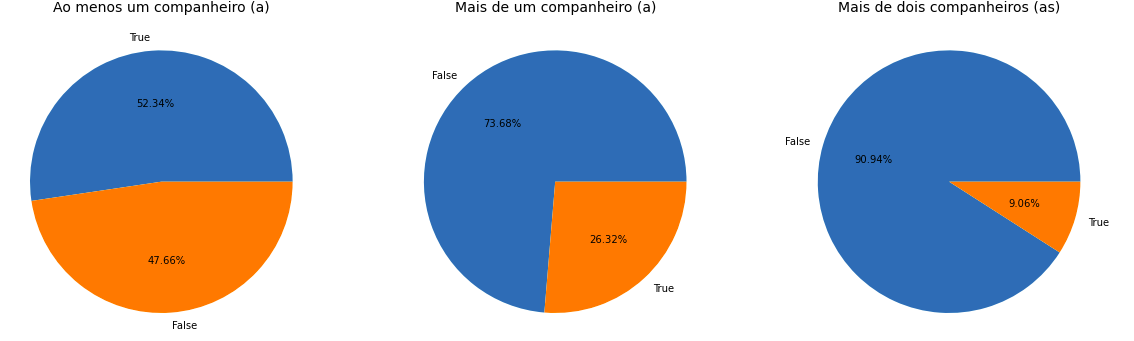
**1.6. Sobreviventes por classe**

Analisando-se o gráfico, existe um certo equilíbrio entre as classes dos sobreviventes. Contudo, mais uma vez olhando para proporções relativas, obtém-se uma grande discrepância, sendo que, proporcionalmente, para primeira, segunda e terceira classe, foram salvos, respectivamente, 63%, 47% e 24% dos passageiros, indicando que realmente a maior parte dos passageiros de primeira classe foram salvos.



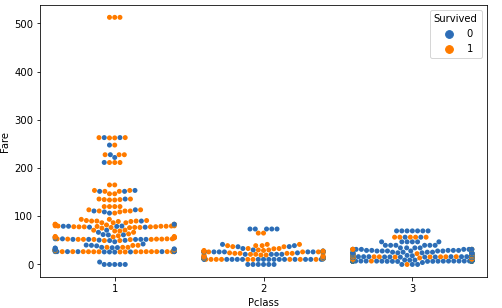
**1.7. Relacionamentos a bordo**

Para facilitar o processo, as colunas que indicam a quantidade de pessoas com a qual um passageiro está relacionado foram somadas, partindo do pressuposto que no momento do desastre qualquer nível de relacionamento impactaria nas atitudes. Para a construção do gráfico foi levado em conta se possuíam ao menos um companheiro, mais de um companheiro, e mais de dois companheiros. Pelas proporções, conforme o número de companheiros sobe a chance de sobrevivência diminui.



**1.8. Sobrevivência por taxa paga**

O gráfico mostra uma combinação entre a taxa paga, a classe e se sobreviveu ou não. Como era de se esperar, a maior parte dos passageiros com taxa alta sobreviveram, estando concentrados principalmente na primeira classe.



**2. Feature Engineering**

Nesta fase, o objetivo é que, após a análise dos dados, sejam feitas manipulações para criação de novos campos, ou a partir da manipulação dos já existentes, ou criando novos a partir de combinações. Para isso, foram desenvolvidas 3 etapas, evoluindo de um processamento inicial para um mais avançado, sendo cada caso testado separadamente na fase de criação dos modelos, que será tratada mais à frente.

**2.1. Etapa 1**

Nesta etapa, foi definido um pré-processamento inicial dos dados, que também foi utilizado nas demais, que envolve a criação definitiva de uma coluna título para treinamento e previsão, uma coluna de relações também já descrita anteriormente, e uma nova coluna, que busca sumarizar a classe do passageiro e a taxa paga, mais especificamente é a a razão entre taxa e classe.

Após isso, foram removidas as colunas que originam esses novos atributos: *Name, Pclass, SibSp, Parch e Fare*. Também foi removida a coluna *Ticket*, que em primeiro momento também não parece ser de muita ajuda, bem como a coluna *Cabin*, que possui quase 80% dos valores nulos.

Também foi criado um *pipeline*, que ficou encarregado de resolver problemas de valores faltantes e transformação dos dados. Para valores numéricos foi definido um *Imputer*, que preenche valores nulos a partir de um valor, sendo a mediana a escolhida para o caso. Já para valores categóricos, o *Imputer* utiliza o valor mais frequente, sendo aplicada posteriormente a técnica de *One-Hot-Encoding*.

**2.2 Etapa 2**

Nesta etapa, foi feito um aprimoramento do que já havia sido feito na anterior. Para a coluna título, foi feito um mapeamento, atribuindo valores mais generalistas para cada um. Para relações foram criadas 4 classes de acordo com o número de companheiros: 0 para nenhum, 1 para um, 2 para dois e 3 para três ou mais passageiros relacionados.

Já no *pipeline*, também foi inclusa uma técnica de *scaling* para valores numéricos, permitindo que valores em escalas diferentes possam ser melhor interpretados pelos modelos. Além disso, a coluna de relações também foi considerada como sendo categórica.

**2.3. Etapa 3**

Na fase de análise observou-se que o conjunto estava com um certo nível de desbalanceamento entre as classes. Nesta etapa, foi feito um balanceamento utilizando *oversampling*, gerando valores randômicos com base nos existentes para equilibrar o número de registros da classe minoritária em relação à majoritária.

**3. Criação dos modelos**

Para previsão dos valores, foram utilizados os modelos: *Decision Tree, Random Forest, Naive Bayes* e *Support Vector Machines*. Para cada modelo foi criado um *pipeline* combinado ao que foi desenvolvido nas etapas de Feature Engineering. Para validação, foi utilizado o método de validação cruzada com cv = 5 e acurácia média como métrica de avaliação de desempenho. Os resultados de validação para cada modelo em cada etapa estão na tabela abaixo.

|  |  |  |  |
| --- | --- | --- | --- |
| **Modelo** | **Etapa 1** | **Etapa 2** | **Etapa 3** |
| **Decision Tree** | 0.7587 | 0.7598 | 0.8588 |
| **Random Forest** | 0.7890 | 0.7811 | 0.8706 |
| **Naive Bayes** | 0.7497 | 0.7811 | 0.7823 |
| **Support Vector Machines** | 0.6836 | 0.8136 | 0.7887 |

Pelos valores, percebemos que na Etapa 1 todos se saíram muito bem, com exceção do SVM. Porém, na Etapa 2, enquanto os outros se mantiveram ou melhoraram apenas um pouco, o SVM obteve um ganho representativo. Isso ocorreu, possivelmente, pelo processo de *scaling*, já que, no caso, o modelo faz o uso de hiperplanos que levam em conta a distância dos atributos, e, portanto, o processo de diminuir e padronizar as escalas permite que ele identifique melhor os padrões. Já para a Etapa 3 o destaque fica para os modelos baseados em árvores, que aparentemente se comportam melhor com conjunto mais balanceados.

**4. Submissão das previsões**

Juntamente com os dados de treino, foram disponibilizados também dados de teste, que são o que de fato servem para avaliar a eficiência das previsões dos modelos. Na Etapa 1, foi escolhido o modelo de *Random Forest* para gerar as previsões, já que obteve a maior acurácia na validação. Como resultado, obteve-se 75.119% de acerto das previsões.

Com a melhora dos modelos na Etapa 2, desta vez foram utilizadas as previsões geradas com o modelo de SVM, atingindo acurácia de 76.076%. Na Etapa 3, porém, apesar do modelo de *Random Forest* atingir uma acurácia de 87% na validação, na submissão esse valor foi de 74.162%.

**5. Conclusões**

Ao longo do processo, o principal objetivo era colocar em prática algumas das lições aprendidas nos cursos disponibilizadas pelo próprio *Kaggle*, envolvendo análise básica de dados a partir de visualização gráfica e manipulação com pandas, bem como desenvolver novas *features* e criar modelos com *pipeline* de processamento dados.

No geral, os dados estavam bem organizados, o que exige menos de quem está participando no que diz respeito à limpeza e estruturação, o que já havia sido dito, já que se trata de uma competição inicial.

Sobre o processo, acredito que foi possível identificar bons *insights* sobre os dados e resumir bem algumas informações combinando os atributos, mas ainda deve ser possível extrair mais um pouco, utilizando por exemplo os campos *Ticket* e *Cabin*.

Em relação aos modelos, foi interessante ver como cada um se comporta de acordo como os atributos são fornecidos para treinamento e predição, dando destaque ao SVM, que teve uma melhora significativa após o processo de *scaling* nos dados numéricos. Porém, também se percebe que muitas vezes, apesar da validação apresentar um valor relativamente alto, esse pode não ser o comportamento com dados exclusivamente novos.