Copy of Smarkio

February 28, 2021

Este notebook foi desenvolvido com o objetivo de aplicação para o processo seletivo de estágio em DataScience na Smarkio Itajubá.

Neste notebook serão respondidas as 5 questões solicitadas no teste da vaga para Data Science, sendo elas:

1. Análise exploratória dos dados utilizando estatística descritiva e inferencial, considerando uma, duas e/ou mais variáveis;
2. Calcule o desempenho do modelo de classificação utilizando pelo menos três métricas;
3. Crie um classificador que tenha como output se os dados com status igual a revision estão corretos ou não (Sugestão : Técnica de cross-validation K-fold);
4. Compare três métricas de avaliação aplicadas ao modelo e descreva sobre a diferença;
5. Crie um classificador, a partir da segunda aba - NLP do arquivo de dados, que permita identificar qual trecho de música corresponde às respectivas artistas listadas (Sugestão: Naive Bayes Classifier).

#1. Análise exploratória

O primeiro ponto a ser abordado, será a análise exploratória dos dados, de forma que tenhamos uma conclusão geral dos dados, quantidades e formatos.

0.0.1 Importando bibliotecas

Primeiramente, antes de começar toda a análise, serão importadas as biliotecas necessárias para se fazer a análise. A bibliotecas usada será:

- Pandas para manipulação dos dados

[ ]:
```
#importando a biblioteca pandas
import pandas as pd
```

0.0.2 Importando dados

Agora que as bibliotecas já estão importadas, é possível importar os dados para que sejam manipulados e análisados. Iremos fazer isso através da biblioteca pandas.

[ ]:
```
data = pd.read_excel("https://s3.amazonaws.com/gupy5/production/companies/634/
↪emails/1614302730414/7b0075b0-77d1-11eb-9933-2573999db431/teste_smarkio_lbs.
↪xls", None)
```

0.0.3 Análise dos dados

Primeiramente iremos visualizar os primeiros dados do registro, para verificar a quantidade de atributos existentes, quantidade de dados e tipos.

[ ]:
```
data['Análise_ML'].head()
```

[ ]:

	Pred_class	probabilidade	status	True_class
0	2	0.079892	approved	0.0
1	2	0.379377	approved	74.0
2	2	0.379377	approved	74.0
3	2	0.420930	approved	74.0
4	2	0.607437	approved	NaN

Dicionário de variáveis

Este dicionário tem como objetivo deixar mais claro cada uma das colunas, para que facilite a análise futura.

Pred_class - Corresponde a classe que foi identificada pelo modelo

probabilidade - Corresponde a probabilidade da classe, identificada pelo modelo

status - Status da classificação, de acordo com um especialista

True_class - Corresponde a classe verdadeira

0.0.4 Quantidade de atributos e entradas e o tipo de cada um

Agora é necessário sabermos a quantidade de linhasXcolunas que o nosso arquivo possui

[ ]:
```
#Identificando volume dos dados
print("Entradas:\t {}".format(data['Análise_ML'].shape[0]))
print("Variáveis:\t {}".format(data['Análise_ML'].shape[1]))

#verificando os tipos de cada coluna
display(data['Análise_ML'].dtypes)
```

```
Entradas:       643
Variáveis:      4

Pred_class        int64
probabilidade   float64
status           object
True_class      float64
dtype: object
```

Portanto, podemos observar que possuímos um total de 643 registros na aba em que estamos trabalhando e um total de 4 colunas, como já havia sido apresentado anteriormente.

0.0.5 Porcentagem de valores ausentes no dataset

Como os valores ausentes ou nulos podem interferir na análise dos dados, é necessário verificar a presença deles em nosso dataset.

[ ]:
```
(data['Análise_ML'].isnull().sum() / data['Análise_ML'].shape[0]).
↪sort_values(ascending=False)
```

[ ]:
```
True_class       0.718507
status           0.000000
probabilidade    0.000000
Pred_class       0.000000
dtype: float64
```

A única coluna com dados nulos é a coluna `True_class`, nos momentos em que os valores dessa coluna são nulos, serão considerados os valores de `Pred_class`.

0.0.6 Informações mais gerais dos dados

Agora serão apresentadas informações mais gerais dos dados numéricos, como média dos valores, desvio padrão e quartis de valores.

[ ]:
```
data["Análise_ML"].describe()
```

[ ]:
```
       Pred_class  probabilidade  True_class
count  643.000000     643.000000  181.000000
mean    52.712286       0.622436   38.574586
std     37.602068       0.266811   39.581017
min      2.000000       0.043858    0.000000
25%     12.000000       0.408017    0.000000
50%     59.000000       0.616809   24.000000
75%     81.000000       0.870083   74.000000
max    118.000000       1.000000  117.000000
```

Primeiramente vamos análisar a coluna de probabilidade.

- A média das probabilidades corresponde a 62.24% de acerto, esse valor irá nos ajudar a categorizar o modelo posteriormente.
- O desvio padrão possuí um valor de 0.266811, isso corresponde ao quanto os valores, de modo geral, se distanciam da média.
- O valor minímo da probabilidade é de 4.3858% e o valor máximo de 100%, o que mostra que há casos em que o modelo conseguiu alcançar 100% de exito na identificação da classe.

Análise da coluna True_class

- O total de itens encontrados foi de 181, portanto, de 643 registros, um total de 462 valores não possuem o valor desta coluna e terão que ser considerados os valores da coluna Pred_class.