CASE ANTT – FERROVIAS BRASILEIRAS

Olá, tudo bem?

Neste primeiro semestre de 2023 eu iniciei pela **USP-ESALQ o MBA em Data Science & Analytics** e gostaria de compartilhar com você o meu primeiro projeto utilizando um algoritmo de árvore de decisão.

Se você ainda não viu a minha publicação sobre as ferrovias Brasileiras, poderá se surpreender e descobrir dados importantes sobre o modal. Basta acessar o link: https://www.linkedin.com/posts/guilherme-lima-747355169_contrata%C3%A7%C3%A3o-de-r-446-bilh%C3%B5es-em-investimentos-activity-7017670654606946304-oyO-?utm_source=share&utm_medium=member_desktop

No estudo de caso atual eu irei utilizar um algoritmo supervisionado de machine learning *(aprendizado de máquina)*, conhecido como árvore de decisão. Uma árvore de decisão pode ser utilizada para prever categorias discretas (sim ou não, por exemplo) e para prever valores numéricos (o valor do lucro em reais).

No case que iremos visualizar eu construí uma categorização discreta, com o objetivo de identificar padrões em meu conjunto de dados. Vamos começar ?!

INÍCIO – DEFININDO O OBJETIVO

Nosso objetivo será identificar alguns dos produtos que são transportados nas ferrovias Brasileiras e que tiveram aumento em sua quantidade de toneladas transportadas, comparando ano vs ano (YOY).

Como o meu notebook pessoal não é lá essas coisas, vamos escolher apenas 1 produto do nosso conjunto de dados e trabalhar com este produto.

Nosso produto alvo então será **GRÃOS DE MILHO**, que tem o código da **NCM = 11042300**.

A minha base de dados foi exportada através do site oficial da Agência Nacional de Transportes Terrestres ANTT – link: https://dados.antt.gov.br/

O material foi construído utilizando a linguagem R e o script está disponibilizado no meu perfil do GITHUB -> https://github.com/iugamil

---

ARVORE DE DECISÃO – UTILIZANDO O MODELO

Após realizar toda a limpeza dos dados e as manipulações necessárias (ETL), temos como resultado o conjunto de dados abaixo:

UF_Origem	COD_FERROVIA	ANO	TONELADAS	QUARTIL_TU	flg_foco
BA	[10,20]	[2006,2009]	695.760	Q1	N
BA	[10,20]	[2006,2009]	4888.930	Q2	N
BA	[10,20]	[2006,2009]	13723.578	Q3	N
DF	[10,20]	[2006,2009]	300.833	Q1	N
ES	[10,20]	[2006,2009]	334.161	Q1	N

A coluna do nosso objetivo é a coluna **FLG_FOCO** pois nela eu já apliquei as condições que vão me retornar se aquele produto é realmente o grão de milho e se ele obteve um aumento na quantidade de toneladas transportadas no ano contra ano (YOY).

Após finalizar todo o trabalho com o conjunto de dados deveremos criar a árvore de decisão. Ela pode ser observada abaixo:

```
18 ############################################################
19 ###################### ARVORE
20 print(paste0("CRIANDO ARVORE","-",now()))
21 # ARVORE
22 set.seed(14263)
23 arvore <- rpart::rpart(flg_foco ~ .,
24                        data= base_arvore,
25                        xval=5,
26                        parms = list(split = 'information'), # podemc
27                        method='class', # Essa opção indica que a res
28                        control=rpart.control(maxdepth = 30, cp=0)
29 )
30
```

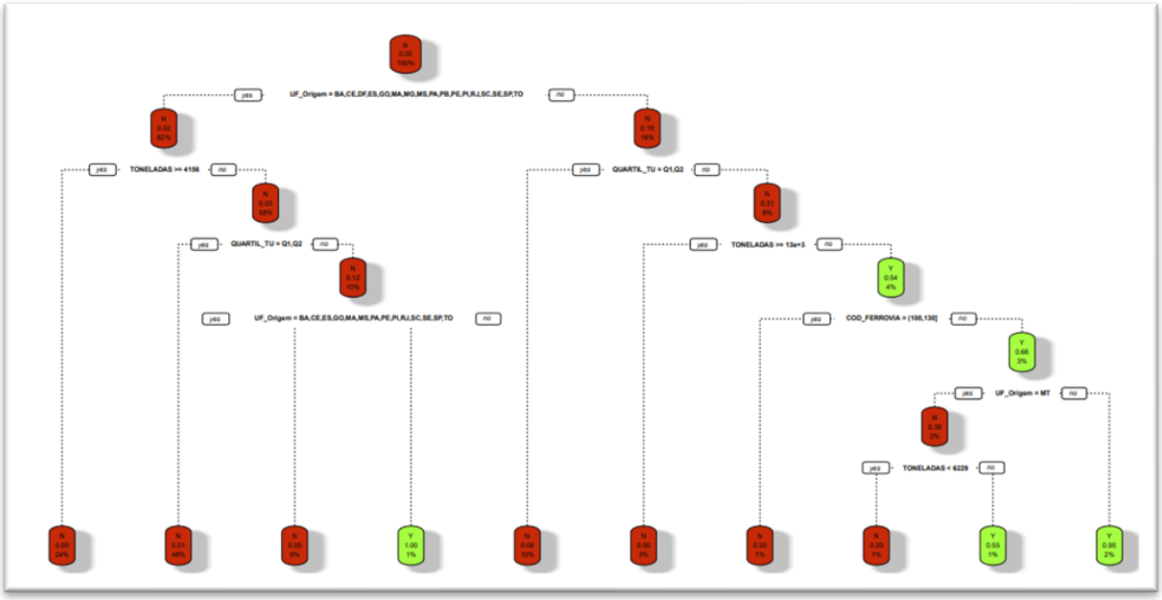
Nesta árvore apontamos a coluna foco para previsão e deixamos que o algoritmo realize os cálculos e as classificações, tomando como base as demais colunas do nosso conjunto de dados. Isso significa que ao utilizar a coluna **FLG_FOCO** em nossa árvore o algoritmo irá ignorar ela e tentará prever o seu valor, de acordo com as demais colunas restantes.

Algumas características importantes desta árvore é que estou definindo como 30 a sua profundidade máxima e estou pedindo que a árvore não seja podada (CP=0), ou seja, ela será completamente desenvolvida até a profundidade máxima especificada pelo parâmetro "maxdepth".

Outra característica observada é o método escolhido para divisão **INFORMATION *(ganho de informação)***. O ganho de informação é uma métrica que mede a redução na incerteza (entropia) após a divisão dos dados em um determinado atributo. O algoritmo de árvore de decisão busca escolher a divisão que maximize o ganho de informação, ou seja, aquela que melhor separa os dados de acordo com a variável de resposta que se está tentando prever.

Por fim temos o parâmetro **XVAL = 5**, que realiza a validação cruzada k-fold. Isso divide os dados em cinco partes e treina/avalia o modelo cinco vezes. Isso nos permite uma maior generalização para dados futuros.

Após rodar a árvore e plotar as suas informações, obtivemos as seguintes classificações:



Obs: devido ao tamanho da árvore a sua visualização será melhor através do PDF que estará disposto junto do material, no meu perfil do github.

```
1) root 1276 63 N (0.95062696 0.04937304)
  2) UF_Origem=BA,CE,DF,ES,GO,MA,MG,MS,PA,PB,PE,PI,RJ,SC,SE,SP,TO 1049 23 N (0.97807436 0.02192564)
    4) TONELADAS>=4156.423 312  0 N (1.00000000 0.00000000) *
    5) TONELADAS< 4156.423 737 23 N (0.96879240 0.03120760)
     10) QUARTIL_TU=Q1,Q2 612  8 N (0.98692810 0.01307190) *
     11) QUARTIL_TU=Q3 125 15 N (0.88000000 0.12000000)
       22) UF_Origem=BA,CE,ES,GO,MA,MS,PA,PE,PI,RJ,SC,SE,SP,TO 116  6 N (0.94827586 0.05172414) *
       23) UF_Origem=MG 9  0 Y (0.00000000 1.00000000) *
  3) UF_Origem=MT,PR,RS 227 40 N (0.82378855 0.17621145)
    6) QUARTIL_TU=Q1,Q2 131 10 N (0.92366412 0.07633588) *
    7) QUARTIL_TU=Q3 96 30 N (0.68750000 0.31250000)
     14) TONELADAS>=12636.91 40  0 N (1.00000000 0.00000000) *
     15) TONELADAS< 12636.91 56 26 Y (0.46428571 0.53571429)
       30) COD_FERROVIA=(100,130] 15  3 N (0.80000000 0.20000000) *
       31) COD_FERROVIA=(70,100] 41 14 Y (0.34146341 0.65853659)
         62) UF_Origem=MT 21  8 N (0.61904762 0.38095238)
          124) TONELADAS< 6228.84 10  2 N (0.80000000 0.20000000) *
          125) TONELADAS>=6228.84 11  5 Y (0.45454545 0.54545455) *
         63) UF_Origem=PR,RS 20  1 Y (0.05000000 0.95000000) *
```

Agora, verificando a matriz de confusão abaixo, vemos que a nossa árvore classificou de forma errada 29 produtos como não estando no nosso objetivo.

Ela classificou errado também outros 6 produtos dizendo que eles eram parte do objetivo, mas na verdade não eram.

classificacao	N	Y
FALSE	1207	29
TRUE	6	34

A acuracidade deste modelo foi de 97,25%.

CONCLUSÃO - FIM DA ANÁLISE

Com uma acuracidade de 97% podemos afirmar que nosso modelo foi bastante positivo e ao observarmos as classificações feitas podemos obter à seguinte conclusão:

- Os transportes com menos de 13 mil toneladas, que não foram movimentados nas ferrovias com código entre 100 e 130 (RMS-EFPO-RMP-FTC) e que vieram dos estados MG,PR,RS. Tem as maiores probabilidades de estarem dentro do nosso objetivo, que era identificar quais produtos tiveram aumento no transporte ano vs ano.

*Lembrando que no estudo de caso desenvolvido aqui nós consideramos apenas 1 produto.

O que podemos tirar disso tudo é que existe um comportamento de aumento nos transportes de milho para os estados de: MG,PR,RS.

Com isso poderíamos realizar visitas técnicas para entender o que mudou na rotina da equipe ou encontrar outra variável operacional que tenha culminado esse aumento. Posteriormente, tomando as melhores decisões para não gerar gargalos nesta produção que vem aumentando a cada ano dentro destas regiões.

Muitos outros objetivos poderiam ser elucidados utilizando as mesmas técnicas que vimos neste material, se você gostou por gentileza deixe seu comentário/feedback.

OBRIGADO!!