Dummy Variables e One Hot Encoding

Esses dois processos ajudam a transformar variáveis nominais e ordinarias em numéricas.

Dummy Variables

Com Dummy Variables criamos uma coluna para cada variável e dá o valor de 1 para os dados que possuíam essa variável. Por exemplo:

Cidade	Area	Preço
SP	2000	5000
RJ	2400	7000

As cidades são variáveis nominais, e não da pra trabalhar com machine learning com esse tipo de variável. Logo é preciso transformá-las. Depois de usar o OneHotEncoding, a tabela ficará assim:

Cidade	Area	Preço	SP	RJ
SP	2000	5000	1	0
RJ	2400	7000	0	1

Depois de importar o pandas, as formulas usadas para criar essas tabelas são:

```
import pandas as pd
dummies = pd.get_dummies(df.Cidade)

merged = pd.concat([df, dummies], axis ='columns')
```

Assim, o novo dataframe merged irá conter os dados de antes e as novas colunas com as variáveis nominais.

A próxima coisa a se fazer é tirar uma coluna dessas variáveis que criamos. Por que? Por causa de uma coisa chamada em inglês de Dummy Variable Trap. O significado dela em português é esse: A armadilha Dummy Variable é um cenário no qual as variáveis independentes são multicolineares - um cenário no qual duas ou mais variáveis são altamente correlacionadas; em termos simples, uma variável pode ser prevista a partir das outras. Ou seja, logicamente não precisamos de uma se já sabemos das outras, essa uma são todas aquelas que não foram selecionadas. Se não tirarmos essa única coluna de nossa escolha, pode dar problema, por isso chamada de trap. O código para tirar uma delas pode ser esse:

final = merged.drop([‘Cidade’, ‘SP’], axis =’columns’)

Area	Preço	RJ
2000	5000	0
2400	7000	1

Agora o dataframe está bem melhor para se trabalhar com algum modelo de machine learning.

One Hot Encoding

Agora vamos transformar as variáveis nominais diretamente em numéricas na própria coluna:

```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
dfle = df
dfle.Cidade = le.fit_transform(dfle.Cidade)
```

Com isso, ficará assim:

Cidade	Area	Preço
1	2000	5000
0	2400	7000

Depois precisamos separar as variáveis X do y. X vai ser Cidade e área e y, o preço:

```
X = dfle[‘Cidade’, ‘Area’]
y = dfle[‘Preço’]
```

Importar o OneHotEnconder, aplicar ele na coluna 0 e :

```
from sklearn.preprocessing import OneHotEncoder
ohe = OneHotEncoder(categorical_features=[0])
X = ohe.fit_transform(X).toarray()
```

É preciso tirar uma coluna como antes, por exemplo a 0:

X = X[:,1:]

Agora pode aplicar um modelo de machine learning como o LinearRegression