Pontos importantes para considerarmos na execução de um projeto de ciência de dados - Regressão Linear

1. Você recebe um dataset

* Considere um Data Set como sendo uma ou mais tabelas com dados.
* As tabelas podem estar em um banco de dados (SQL ou NoSQL).
* As tabelas podem estar em diferentes formatos .CSV,.JASON, .HDF, .PARQUET, entre outros.
* Em um mesmo dataset diferentes tabelas podem estar salvas em diferentes formatos.

1. Exploração dos Dados

* Verificar se os tipos de dados são condizentes com as informações que estão representadas.
  + Variáveis categóricas devem ser do tipo object ou string
  + Variáveis binárias devem ser do tipo int ou boolean
* Verificar se há dados faltantes:
  + Até 5% das linhas faltantes:
    - Os dados podem ser excluídos.
    - Os dados preenchidos podem ser preenchidos
  + De 5% até 50% das linhas faltantes:
    - Investigar se os dados podem ser preenchidos. Há casos que o preenchimento de dados não é desejável, como por exemplo, algumas aplicações médicas.
    - Se o preenchimento fizer sentido do ponto de vista de negócio, fazer o preenchimento (com valor único, criar regras para casos específicos, regressão considerando os vizinhos).
  + Acima de 50% dos dados faltantes:
    - Essa quantidade de dados faltantes começa a se tornar perigosa, para fazermos o preenchimento. Por esse motivo, a investigação sobre a necessidade de preenchimento deve ser feita de maneira muita cuidadosa, para não inserir algum tipo de viés no dado.
* Verificar se todas as colunas disponíveis trazem alguma informação relevante ao modelo:
  + Podemos excluir por exemplo colunas que tem valor único por linha.
* Transformar variáveis categóricas:
  + Dois principais são:
    - OneHotEncoder (get\_dummy do pandas): nesse caso podemos aplicar o encoder antes de fazer a divisão entre treino e teste.
    - FrequencyEncoder: nesse caso temos que fazer a separação entre teste e treino antes de aplicar esse encoder.

1. Divisão entre treino e teste

* Verificar se os dados possuem ordenamento temporal ou não:
  + Caso não tenha: A divisão entre treino e teste dos dados pode ser feita de maneira aleatória. Quantidade recomendada 80% dos dados para treinar o modelo e 20% para testar o modelo. A razão da escolha desses números ficará clara mais à frente no curso.
  + Caso tenha ordenamento temporal: Devemos treinar o modelo com dados do passado e testar os dados com dados do futuro. Não há uma quantificação correta do tamanho do conjunto de treino, na verdade esse parâmetro deve ser ajustado para termos o melhor caso possível.
  + Retirar a coluna que contém os dados que temos como alvo.
* Caso tenha variáveis categóricas para aplicar o FrequencyEncoder, esse procedimento pode ser feito agora (lembrar das aulas dos dias 25 e 28/09/2020).
* Normalizar os dados, lembrar do procedimento:
  + Scaler\_applied = scaler.fit(dados\_de\_treino)
  + Scaler\_applied.transform(dados\_de\_treino)
  + Scaler\_applied.transform(dados\_de\_teste)

1. Aplicação da regressão linear

* Fazer o treinamento do modelo (lembrar dos códigos das aulas dos dias 25 e 28/09/2020, olhar a linha que contém LinearRegression().fit(dados\_de\_treino).
* Fazer predições com os dados de treino (.predict(dados\_de\_treino)) e teste (.predict(dados\_de\_teste)).
* Escolher uma métrica para avaliar as duas predições feitas acima:
  + O modelo de regressão linear nos dá automaticamente o valor de R2 (score\_).
  + É considerado boa prática avaliar seu modelo de regressão inicialmente com RMSE
* A métrica no treino não pode ser muito melhor que a métrica no teste.
  + Os limites de aceitação desta diferença são muito relativos, temos que analisar caso a caso.
* Podemos avaliar a relevância de cada feature (colunas do data set), avaliando o valor dos coeficientes que multiplicam cada colunas (.coef\_).

1. Como melhorar o modelo

* Criar novas features:
  + Fazer algum tipo de operação entre colunas disponíveis:
    - Soma, diferença, média, multiplicação.
* Considerar novas fetures.
* Aumentar a quantidade de dados.
* Mudar o scaler.
* Caso tenta features categóricas mudar a maneira de transformar dos dados categóricos