

Bootcamp IGTI: Engenheiro(a) de Machine Learning

Desafio

Módulo 3	Seleção de Modelos de Aprendizado de Máquina
-----------------	---

Objetivos

Exercitar os seguintes conceitos trabalhados no Módulo:

- ✓ Neste desafio, vamos trabalhar os conceitos vistos sobre métricas de desempenho, técnicas de validação e sintonia de hiperparâmetros.
- ✓ Também vamos olhar para o fluxo completo de seleção de um modelo de aprendizado.

Enunciado

Neste desafio, vamos fazer um apanhado geral de tudo que foi visto no módulo. Vamos usar a tarefa de classificação para validar um modelo, otimizar os hiperparâmetros desse modelo e avaliar o resultado encontrado de acordo com algumas métricas de desempenho vistas durante o módulo.

Atividades

Os alunos deverão desempenhar as seguintes atividades:

1. Baixar o arquivo com os dados no link <https://www.openml.org/d/1480>. O formato do arquivo deve ser CSV.
2. Obter informações relativas a número de features e amostras.
3. Verificar a necessidade de tratamento de dados categóricos e valores faltantes.

4. Modelar o SVC e o Random Forest Classifier, com Random Search para sintonia de hiperparâmetros e validação cruzada estratificada, usando as parametrizações abaixo.

5. Parametrização SVC

- a. Bibliotecas para importação

- i. `from sklearn.ensemble import RandomForestClassifier`
 - ii. `from sklearn.svm import SVC`
 - iii. `from sklearn.model_selection import StratifiedKFold`
 - iv. `from sklearn.model_selection import RandomizedSearchCV`
 - v. `from scipy.stats import uniform`
 - vi. `from scipy.stats import randint`
 - vii. `from sklearn.metrics import f1_score, make_scorer`

- b. Kfold estratificado com 10 conjuntos.

- c. Métrica de avaliação f1

- i. `f1 = make_scorer(f1_score)`

- d. Parâmetro de kernel:

- i. Sigmoidal, polinomial e RBF.

- e. Parâmetro de regularização C:

- i. Distribuição uniforme variando entre 1 e 10

- f. `Random_state = 54`

- g. Número de iterações = 5.

6. Avaliar o resultado da modelagem usando as métricas:

- a. `best_score_`
 - b. `best_params_`
 - c. `best_estimator_`

7. Repetir o processo usando o Random Forest:

- a. Faça a instanciação do Random Forest fixando o `random_state = 54`

- i. `RandomForestClassifier(random_state = 54)`

- b. Kfold estratificado com 10 conjuntos.

- c. Métrica de avaliação f1

- i. `f1 = make_scorer(f1_score)`

- d. Parâmetro do número de árvores:

- i. Distribuição aleatória inteira de valores entre 10 e 1000

- e. Parâmetro Bootstrap:

- i. Verdadeiro e Falso

- f. Parâmetro Criterion:

- i. Gini e Entropy

- g. `Random_state = 54`

- h. Número de iterações = 5.

8. Avaliar o resultado da modelagem usando as métricas:

- a. `best_score_`

- b. `best_params_`

- c. `best_estimator_`



