

**Bootcamp IGTI: Desenvolvedor Python****Trabalho Prático**

<b>Módulo 4</b>	<b>Python Avançado</b>
-----------------	------------------------

**Objetivos**

Exercitar os seguintes conceitos trabalhados no Módulo:

- ✓ Python para aprendizado de máquina
- ✓ Programação concorrente com Python

**Enunciado**

A linguagem Python é bastante utilizada no meio da ciência de dados e do aprendizado de máquina. Outras áreas do conhecimento, como a matemática, a física e a probabilidade, também a empregam bastante para ajudar a simular e a resolver problemas através da programação. Boa parte da popularidade dessa linguagem está relacionada à forma como o código é construído e à sua facilidade de leitura e de interpretação.

Neste trabalho, será utilizado o scikit-learn e os conceitos apresentados na disciplina para a realização de uma análise de dados baseada em modelos de aprendizado de máquina. Para isso, será utilizado um conjunto de dados público.

Esse conjunto de dados consiste em uma pesquisa realizada com um grupo de mulheres. Nessa pesquisa, foram coletados alguns indicadores sobre a saúde e os hábitos dessas pacientes, como número de vezes em que engravidou, seu nível de glicose, sua pressão sanguínea, etc., a fim de prever se ela vai ou não desenvolver a diabetes. Na última coluna do conjunto de dados (outcome), o valor “1” indica que a paciente possui diabetes, e “0” indica que não. Desse modo, será possível utilizar os algoritmos apresentados e o scikit-learn para tentar prever a ocorrência ou não da doença.

O código abaixo deve ser utilizado para as questões de concorrência:

## Código 1

```
import threading
import time
from random import randint

def funcao_1(num):
    n=num
    while n>0:
        n-=1
        print("n_1: {}".format(n))
        time.sleep(randint(0,2))

def funcao_2(num):
    n=num

    while n<100:
        n+=1
        print("n_2: {}".format(n))
        time.sleep(randint(0,2))

if __name__ == "__main__":

    t1 = threading.Thread(target=funcao_1, args=(100,))
    t2 = threading.Thread(target=funcao_2, args=(0,))

    t1.start()
    t2.start()

    t1.join()
    t2.join()

    print("Fim!")
```

## Atividades

Os alunos deverão desempenhar as seguintes atividades:

1. Acessar o link abaixo e realizar o *download* do arquivo “**datasets\_diabetes.csv**”.

[https://drive.google.com/drive/folders/1-KhRI5Q1QK8d\\_N4AMRRoZti8gdQFGLpX?usp=sharing](https://drive.google.com/drive/folders/1-KhRI5Q1QK8d_N4AMRRoZti8gdQFGLpX?usp=sharing)

2. Para a implementação dos algoritmos, utilize as definições abaixo:

### Algoritmo KNN:

```
clf_KNN = KNeighborsClassifier(n_neighbors=5)
```

### Algoritmo Árvore de Decisão

```
clf_arvore = DecisionTreeClassifier(random_state=1)
```

### Algoritmo Floresta Randômica

```
clf_floresta = RandomForestClassifier(max_depth=10, random_state=1)
```

### Algoritmo SVM

```
clf_svm=SVC(gamma='auto', random_state=1)
```

### Algoritmo Rede MLP

```
clf_mlp = MLPClassifier(solver='lbfgs', alpha=1e-5, hidden_layer_sizes=(5, 5), random_state=1)
```

### Observações

- Utilize como entrada as colunas: ***Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction*** e ***Age***. A saída para os algoritmos deve ser a coluna ***Outcome***.
- Realize a carga dos valores utilizando o método `pandas.read_csv()`
- Utilize, para normalização dos dados, as definições:
  - `normaliza = MinMaxScaler()` *#objeto para a normalização*
  - `entradas_normalizadas=normaliza.fit_transform(entradas)`
- Utilize, para divisão entre treinamento e teste do algoritmo, as definições:  
`train_test_split(entradas_normalizadas, saida, test_size=0.30, random_state=42)`
- Utilize esta sequência de operações para chegar no resultado final:
  - Divida os dados entre entrada e saída;
  - Normalize apenas as entradas utilizando o ***MinMaxScaler***;
  - Aplique a divisão entre o treinamento e o teste com o ***train\_test\_split***.
- Utilize os dados de “teste” para avaliar as previsões de classificação dos modelos.



## **Respostas Finais**

Os alunos deverão desenvolver a prática e, depois, responder às seguintes questões objetivas: