keystroke

November 12, 2018

```
In [1]: from source.util import utils as u
        from source import metrics, plots
        from source import handshake2, scarge, hs
        import sys
        import time
        import os
        import psutil
        import resource
In [2]: poolsize = 150
       clusters = 4
        n\_components = 4
        epsilon = 0.1
        percent = 30
In [3]: base = '/home/localuser/Documentos/procopio/tcc/datasets/keystroke.txt'
        dataset, data_labeled, dataset_train, l_train, stream, l_stream, n_features = u.criar_da
  Handshake
In [4]: start = time.time()
        predicted, updt = handshake2.handshake2(dataset, data_labeled, dataset_train, l_train, s
        end = time.time()
        mem = resource.getrusage(resource.RUSAGE_SELF).ru_maxrss
        tempo = end - start
  SCARGC
In [5]: startScargc = time.time()
        predictedS, updtS = scargc.scargc_1NN(dataset, data_labeled, dataset_train, l_train, str
        endScargc = time.time()
        memS = resource.getrusage(resource.RUSAGE_SELF).ru_maxrss
        tempoS = endScargc - startScargc
```

```
/home/localuser/anaconda3/lib/python3.6/site-packages/sklearn/cluster/k_means_.py:896: RuntimeWa
 return_n_iter=True)
In [6]: acc_percent, f1_per, mcc_per = metrics.makeBatches(l_stream, predicted, len(stream))
        score, f1, mcc, std = metrics.metrics(acc_percent, l_stream, predicted, f1_type = 'macro
        acc_percentScargc, f1_S, mcc_S = metrics.makeBatches(l_stream, predictedS, len(stream))
        scoreS, f1S, mccS, stdS = metrics.metrics(acc_percentScarge, l_stream, predictedS, f1_ty
/home/localuser/anaconda3/lib/python3.6/site-packages/sklearn/metrics/classification.py:1135: Ur
  'precision', 'predicted', average, warn_for)
In [7]: print('Tempo de Execução: ', tempo)
        print('memory peak: ', mem)
        print('Acc: ', score)
        print('Macro-F1: ', f1)
        print('MCC: ', mcc)
        print('Desvio Padrão: ', std)
        print('Numero de atualizações: ', updt)
        plots.plotAcc(acc_percent, 100, 'Keystroke_Handshake')
        plots.plotF1(f1_per, 100, 'Keystroke_Handshake')
        print('Tempo de Execução: ', tempoS)
        print('memory peak: ', memS)
        print('Acc: ', scoreS)
        print('Macro-F1: ', f1S)
        print('MCC: ', mccS)
        print('Desvio Padrão: ', stdS)
        print('Numero de atualizações: ', updtS)
        plots.plotAcc(acc_percentScargc, 100, 'Keystroke_SCARGC')
        plots.plotF1(f1_S, 100, 'Keystroke_SCARGC')
        listTime = [tempo, tempoS]
        listAcc = [score, scoreS]
        listMethod = ['Handshake', 'SCARGC']
        matrixAcc = [acc_percent[0], acc_percentScargc[0]]
        matrixF1 = [f1_per[0], f1_S[0]]
        plots.plotTime(listTime, listMethod)
        plots.plotAverageAcc(listAcc, listMethod)
        plots.plotAccuracyCurves(matrixAcc, listMethod)
        plots.plotBoxplot('acc', matrixAcc, listMethod)
        plots.plotBoxplot('f1', matrixF1, listMethod)
Tempo de Execução: 1.480457067489624
memory peak: 130148
```

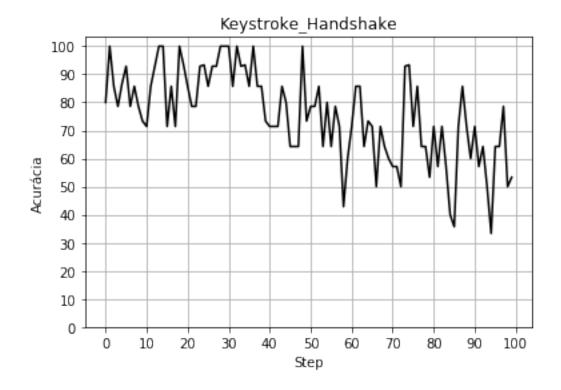
Acc: 0.756047619047619

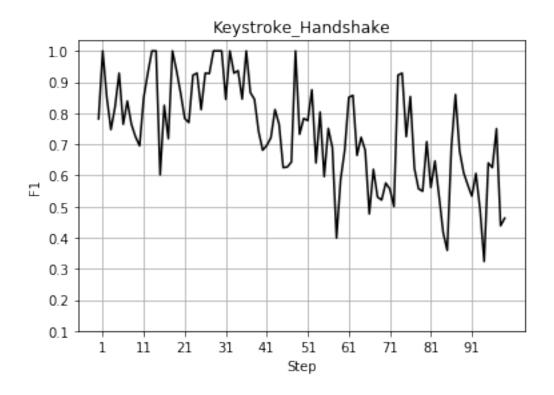
Macro-F1: 0.7536170370778992

MCC: 0.6755450776002652

Desvio Padrão: 0.15745258797592834

Numero de atualizações: 3





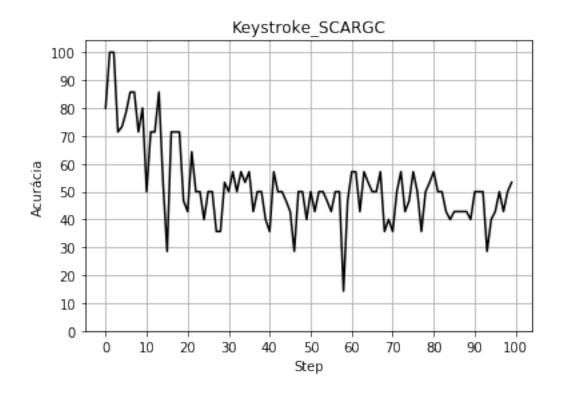
Tempo de Execução: 0.9399621486663818

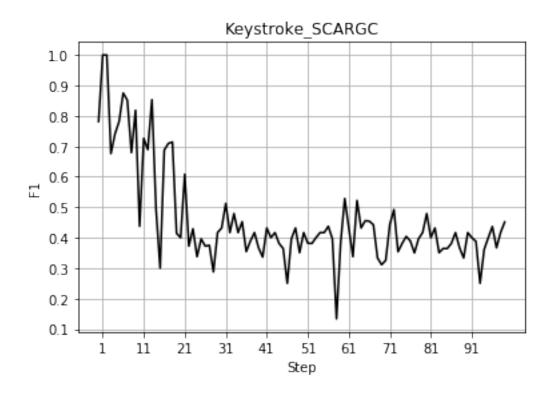
memory peak: 130776 Acc: 0.520952380952381

Macro-F1: 0.4838557725313432 MCC: 0.38370631569458036

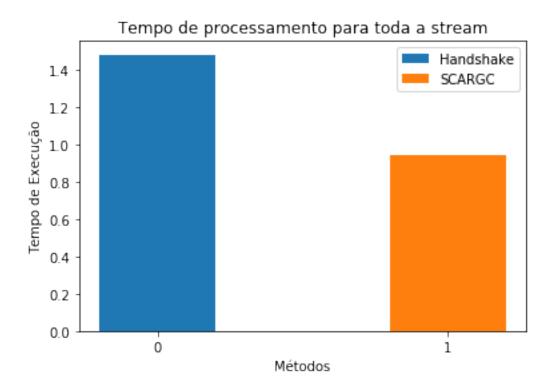
Desvio Padrão: 0.14471333771763642

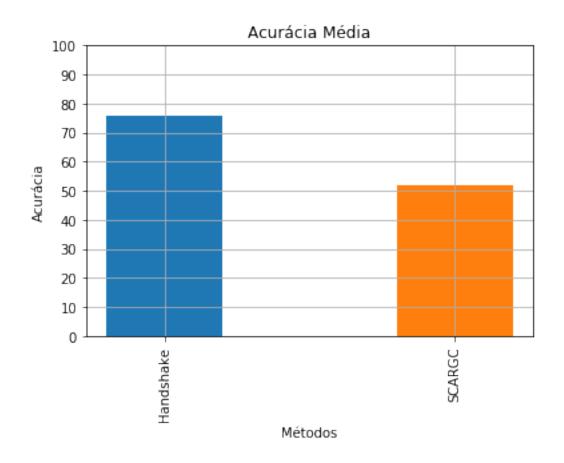
Numero de atualizações: 10

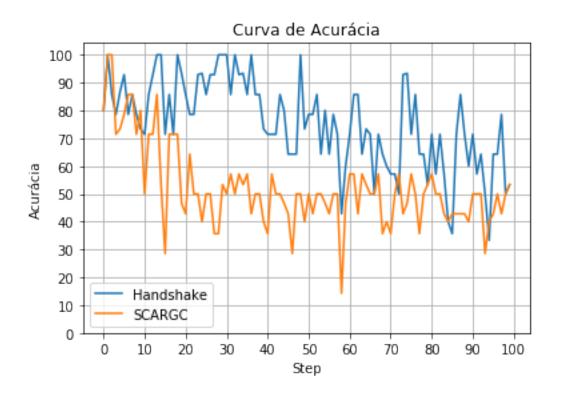


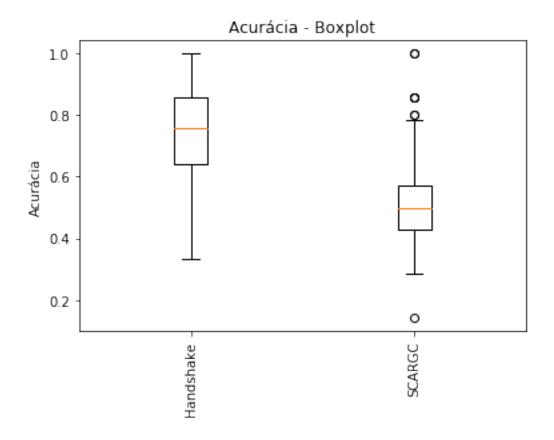


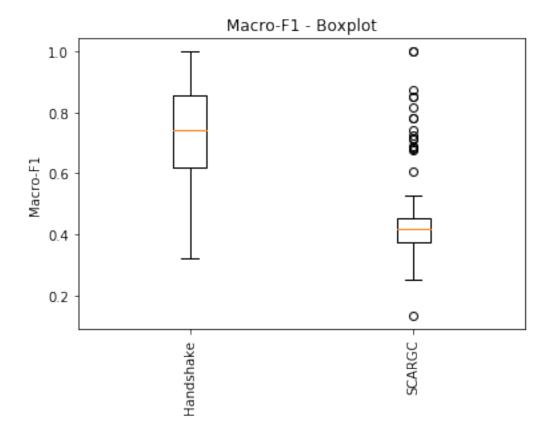
/home/localuser/anaconda3/lib/python3.6/site-packages/matplotlib/cbook/deprecation.py:106: Matpl warnings.warn(message, mplDeprecation, stacklevel=1)











In [8]: plots.plotPerBatches(stream, predicted, l_stream, len(stream))



