print_measures

June 2, 2021

[1]:
```
import numpy as np
import pandas as pd
import pathlib
import sys

# Absolute path to this file
MY_PAPI_DIR = pathlib.Path().absolute()
# Now, we have to move to the root of this workspace ([prev. path]/TFG)
MY_PAPI_DIR = MY_PAPI_DIR.parent.parent.parent.parent.absolute()
# From the root (TFG/) access to my_papi dir. and its content
MY_PAPI_DIR = MY_PAPI_DIR / "my_papi"
# Folder where the configuration files are located
CFG_DIR = MY_PAPI_DIR / "conf"
# Folder where the library is located
LIB_DIR = MY_PAPI_DIR / "lib"
# Folder where the source codes are located
SRC_DIR = MY_PAPI_DIR / "src"

# Add the source path and import the library
sys.path.insert(0, str(SRC_DIR))
from MyPapi import *
```

[19]:
```
def df_from_csv(csv_file):
    # Read csv with the following name of columns
    df = pd.read_csv(csv_file, header=None, sep=":", names=["CPU", "Value",␣
 ↪"Unit", "Event Name"])

    # Get the rows which have no 0 values
    df = df[df.Value != 0]

    # Define the number of epochs and number of measures
    num_epochs = 30
    num_measures = 5

    # Get the events measured
    events = df["Event Name"].unique()
```

```
    # Split the Dataframe in num_measures
    arrs = np.array_split(df, num_measures)

    # Store that mean columns in a new Dataframe
    data = []
    headers = []
    i = 0
    # Calculate the mean of each iteration
    for arr in arrs:
        arr = arr.reset_index(drop=True)
        arr['Avg'] = arr.groupby('Event Name')['Value'].transform('sum')
        i = i + 1
        headers.append("Measure_" + str(i))
        data.append(arr['Avg'].head(len(events)))

    # Creates a new df from the avg of the iters
    df = pd.concat(data, axis=1, keys=headers)

    # Add the events names
    df.insert(0, 'Event Name', events)

    # Creates a new avg column from all the measures
    df['Avg'] = df.mean(axis=1)
    return df
```

[20]:
```
# Time to print the values of the measure M1
#pd.set_option('display.width', 200)

# Name of csv file
csv_file = "/home/jlpadillas01/TFG/tests/tensorflow/mnist/out/mnist_train_papi.
 ↪csv"

df1 = df_from_csv(csv_file)
df1
```

[20]:

	Event Name	Measure_1	Measure_2	\
0	PERF_COUNT_HW_INSTRUCTIONS	193532612763	191722933694	
1	PERF_COUNT_HW_BRANCH_INSTRUCTIONS	16637978284	16420111891	
2	L1-DCACHE-LOADS	67197413612	66654715550	
3	L1-DCACHE-STORES	15638234031	15191754159	
4	fp_arith_inst_retired.scalar_double	247316414	247112323	
5	fp_arith_inst_retired.scalar_single	154688753	154688753	
6	fp_arith_inst_retired.128b_packed_single	15056280	15056280	
7	fp_arith_inst_retired.256b_packed_single	3403140	3403140	
8	fp_arith_inst_retired.512b_packed_single	122693346870	122693346870	

Measure_3	Measure_4	Measure_5	Avg

```
0  193274627483  193088825159  191796478519  1.926831e+11
1   16585350509   16537362003   16438151694  1.652379e+10
2   67136605844   67102454857   66676152816  6.695347e+10
3   15620269336   15612795448   15199728296  1.545256e+10
4     247317042     247318820     247108318  2.472346e+08
5     154688753     154688753     154688753  1.546888e+08
6      15056280      15056280      15056280  1.505628e+07
7       3403140       3403140       3403140  3.403140e+06
8  122693346870  122693346870  122693346870  1.226933e+11
```

[21]:
```
# Time to print the values of the measure M1

# Name of csv file
csv_file = "/home/jlpadillas01/TFG/tests/tensorflow/mnist/out/
 ↪mnist_train_callback.csv"

df2 = df_from_csv(csv_file)
df2
```

[21]:
```
                                Event Name     Measure_1     Measure_2  \
0               PERF_COUNT_HW_INSTRUCTIONS  193348191211  191548125053
1         PERF_COUNT_HW_BRANCH_INSTRUCTIONS   16622251581   16399300908
2                           L1-DCACHE-LOADS   67139940133   66622156153
3                          L1-DCACHE-STORES   15615410143   15195885986
4       fp_arith_inst_retired.scalar_double     247318651     247116172
5       fp_arith_inst_retired.scalar_single     154688753     154688753
6  fp_arith_inst_retired.128b_packed_single      15056280      15056280
7  fp_arith_inst_retired.256b_packed_single       3403140       3403140
8  fp_arith_inst_retired.512b_packed_single  122693346870  122693346870

      Measure_3     Measure_4     Measure_5           Avg
0  193282675683  193303688563  193141257833  1.929248e+11
1   16609379322   16613014698   16573047894  1.656340e+10
2   67126302193   67126573565   67106292252  6.702425e+10
3   15611010058   15639401699   15616753098  1.553569e+10
4     247323240     247323978     247323807  2.472812e+08
5     154688753     154688753     154688753  1.546888e+08
6      15056280      15056280      15056280  1.505628e+07
7       3403140       3403140       3403140  3.403140e+06
8  122693346870  122693346870  122693346870  1.226933e+11
```

[22]:
```
# Time to print the values of the measure M2

# Name of csv file
csv_file = "/home/jlpadillas01/TFG/tests/tensorflow/mnist/out/
 ↪mnist_train_each_epoch.csv"
```

```
df3 = df_from_csv(csv_file)
df3
```

[22]:

	Event Name	Measure_1	Measure_2	Measure_3	Measure_4	Measure_5	Avg
0	PERF_COUNT_HW_INSTRUCTIONS	193018654911	192931572282	192749210060	192882615270	191559184090	1.926282e+11
1	PERF_COUNT_HW_BRANCH_INSTRUCTIONS	16570142702	16536470673	16504035415	16530853681	16418061513	1.651191e+10
2	L1-DCACHE-LOADS	67066125578	67079106212	67003101111	67055582971	66614889265	6.696376e+10
3	L1-DCACHE-STORES	15580596331	15593821161	15560556177	15593172820	15169626081	1.549955e+10
4	fp_arith_inst_retired.scalar_double	247323788	247311964	247114824	247316462	247109695	2.472353e+08
5	fp_arith_inst_retired.scalar_single	154688693	154688693	154688693	154688693	154688693	1.546887e+08
6	fp_arith_inst_retired.128b_packed_single	15056280	15056280	15056280	15056280	15056280	1.505628e+07
7	fp_arith_inst_retired.256b_packed_single	3403140	3403140	3403140	3403140	3403140	3.403140e+06
8	fp_arith_inst_retired.512b_packed_single	122693346870	122693346870	122693346870	122693346870	122693346870	1.226933e+11

[24]:
```
# Time to print the values of the measure M3

# Name of csv file
csv_file = "/home/jlpadillas01/TFG/tests/tensorflow/mnist/out/
 ↪mnist_train_each_batch.csv"

df4 = df_from_csv(csv_file)
df4
```

[24]:

	Event Name	Measure_1	Measure_2	Measure_3	Measure_4	Measure_5	Avg
0	PERF_COUNT_HW_INSTRUCTIONS	132830362070	132828259980	132814086410	132801859154	132888970331	1.328327e+11
1	PERF_COUNT_HW_BRANCH_INSTRUCTIONS	4891139917	4890817864				
2	L1-DCACHE-LOADS	48814711277	48815201214				
3	L1-DCACHE-STORES	4368069325	4368426079				
4	fp_arith_inst_retired.scalar_double	239661702	239661275				
5	fp_arith_inst_retired.scalar_single	152115060	152115060				
6	fp_arith_inst_retired.128b_packed_single	15150000	15150000				
7	fp_arith_inst_retired.256b_packed_single	3345001	3345001				
8	fp_arith_inst_retired.512b_packed_single	120849165006	120849165006				

```
1    4889634622    4886124317    4907140109  4.892971e+09
2   48810558927   48809871115   48824586381  4.881499e+10
3    4366934859    4365278614    4365119175  4.366766e+09
4     239661279     239661351     239661162  2.396614e+08
5     152115060     152115060     152115060  1.521151e+08
6      15150000      15150000      15150000  1.515000e+07
7       3345001       3345001       3345001  3.345001e+06
8  120849165006  120849165006  120849165006  1.208492e+11
```

[31]:
```
# Join the two dataframes in another one
events = pd.Series(df4["Event Name"].unique())
data = [events, df1['Avg'], df2['Avg'], df3['Avg'], df4['Avg']]
headers = ['Event Name', 'df1', 'df2', 'df3', 'df4']
df5 = pd.concat(data, axis=1, keys=headers)
#df5['Value_diff'] = (df3['df1'] - df3['df2']).abs()
df5
```

[31]:
```
                              Event Name           df1           df2  \
0                PERF_COUNT_HW_INSTRUCTIONS  1.926831e+11  1.929248e+11
1          PERF_COUNT_HW_BRANCH_INSTRUCTIONS  1.652379e+10  1.656340e+10
2                           L1-DCACHE-LOADS  6.695347e+10  6.702425e+10
3                          L1-DCACHE-STORES  1.545256e+10  1.553569e+10
4        fp_arith_inst_retired.scalar_double  2.472346e+08  2.472812e+08
5        fp_arith_inst_retired.scalar_single  1.546888e+08  1.546888e+08
6  fp_arith_inst_retired.128b_packed_single  1.505628e+07  1.505628e+07
7  fp_arith_inst_retired.256b_packed_single  3.403140e+06  3.403140e+06
8  fp_arith_inst_retired.512b_packed_single  1.226933e+11  1.226933e+11

            df3           df4
0  1.926282e+11  1.328327e+11
1  1.651191e+10  4.892971e+09
2  6.696376e+10  4.881499e+10
3  1.549955e+10  4.366766e+09
4  2.472353e+08  2.396614e+08
5  1.546887e+08  1.521151e+08
6  1.505628e+07  1.515000e+07
7  3.403140e+06  3.345001e+06
8  1.226933e+11  1.208492e+11
```

[33]:
```
with pd.ExcelWriter('output.xlsx') as writer:
    df1.to_excel(writer, sheet_name='mnist_train_papi')
    df2.to_excel(writer, sheet_name='mnist_train_callback')
    df3.to_excel(writer, sheet_name='mnit_train_each_epoch')
    df4.to_excel(writer, sheet_name='mnit_train_each_batch')
    df5.to_excel(writer, sheet_name='mnit_train_comparation')
```

[ ]:
```

```