IBM **Watson Studio** Projects Community Services Manage Support

spark-ct ibm ai week4 keras-ml

VERSION AUTHOR	LAST UPDATED	LANGUAGE
Unknown	24 Jul 2018, 6:29 PM	Python 2 with Spark 2.1

Assignment 4

Understaning scaling of linear algebra operations on Apache Spark using Apache SystemML

In this assignment we want you to understand how to scale linear algebra operations from a single machine to multiple machines, memory and CPU cores using Apache SystemML. Therefore we want you to understand how to migrate from a numpy program to a SystemML DML program. Don't worry. We will give you a lot of hints. Finally, you won't need this knowledge anyways if you are sticking to Keras only, but once you go beyond that point you'll be happy to see what's going on behind the scenes. As usual, we run some import statements:

In [1]:

```
!pip install --upgrade systemml
```

```
Requirement already up-to-date: systemml in /gpfs/global_fs01/sym_share
d/YPProdSpark/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.
7/site-packages
Requirement already up-to-date: pandas in /gpfs/global_fs01/sym_shared/
YPProdSpark/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/
site-packages (from systemml)
Requirement already up-to-date: scipy>=0.15.1 in /gpfs/global_fs01/sym_
shared/YPProdSpark/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/pyt
hon2.7/site-packages (from systemml)
Requirement already up-to-date: scikit-learn in /gpfs/global_fs01/sym_s
hared/YPProdSpark/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/pyth
on2.7/site-packages (from systemml)
Requirement already up-to-date: Pillow>=2.0.0 in /gpfs/global_fs01/sym_
shared/YPProdSpark/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/pyt
hon2.7/site-packages (from systemml)
Requirement already up-to-date: numpy>=1.8.2 in /gpfs/global_fs01/sym_s
hared/YPProdSpark/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/pyth
on2.7/site-packages (from systemml)
Requirement already up-to-date: python-dateutil>=2.5.0 in /gpfs/global_
fs01/sym_shared/YPProdSpark/user/s0f2-ba03446bdf62cc-bd5847e99873/.loca
l/lib/python2.7/site-packages (from pandas->systemml)
Requirement already up-to-date: pytz>=2011k in /gpfs/global_fs01/sym_sh
ared/YPProdSpark/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/pytho
n2.7/site-packages (from pandas->systemml)
Requirement already up-to-date: six>=1.5 in /gpfs/global_fs01/sym_share
d/YPProdSpark/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.
7/site-packages (from python-dateutil>=2.5.0->pandas->systemml)
```

In [ ]:

```
'''
import pip

try:
    __import__('pandas')
except ImportError:
    pip.main(['install', 'pandas'])

try:
    __import__('dateutil')
except ImportError:
    pip.main(['install', 'dateutil'])
```

```
try:
    __import__('systemml')
except ImportError:
    pip.main(['install', 'systemml'])
'''
```

In [ ]:

```
#!pip uninstall python-dateutil
#!pip install python-dateutil --upgrade
'''
pip.main(['uninstall', 'python-dateutil'])
pip.main(['install', 'python-dateutil'])
'''
```

In [ ]:

```
'''
from pandas.compat.numpy import dateutil
'''
```

In [2]:

```
from systemml import MLContext, dml
import numpy as np
import time
```

```
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/_libs/__init__.py:4: RuntimeWarning: numpy.dtype si
ze changed, may indicate binary incompatibility. Expected 96, got 88
  from .tslib import iNaT, NaT, Timestamp, Timedelta, OutOfBoundsDateti
me
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/__init__.py:26: RuntimeWarning: numpy.dtype size ch
anged, may indicate binary incompatibility. Expected 96, got 88
  from pandas._libs import (hashtable as _hashtable,
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/core/dtypes/common.py:6: RuntimeWarning: numpy.dtyp
e size changed, may indicate binary incompatibility. Expected 96, got 8
8
  from pandas._libs import algos, lib
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/core/util/hashing.py:7: RuntimeWarning: numpy.dtype
size changed, may indicate binary incompatibility. Expected 96, got 88
  from pandas._libs import hashing, tslib
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/core/indexes/base.py:7: RuntimeWarning: numpy.dtype
size changed, may indicate binary incompatibility. Expected 96, got 88
  from pandas._libs import (lib, index as libindex, tslib as libts,
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/tseries/offsets.py:21: RuntimeWarning: numpy.dtype
size changed, may indicate binary incompatibility. Expected 96, got 88
  import pandas._libs.tslibs.offsets as liboffsets
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/core/ops.py:16: RuntimeWarning: numpy.dtype size ch
anged, may indicate binary incompatibility. Expected 96, got 88
  from pandas._libs import algos as libalgos, ops as libops
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/core/indexes/interval.py:32: RuntimeWarning: numpy.
dtype size changed, may indicate binary incompatibility. Expected 96, g
ot 88
  from pandas._libs.interval import (
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/core/internals.py:14: RuntimeWarning: numpy.dtype s
ize changed, may indicate binary incompatibility. Expected 96, got 88
  from pandas._libs import internals as libinternals
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
```

```
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/core/sparse/array.py:33: RuntimeWarning: numpy.dtyp
e size changed, may indicate binary incompatibility. Expected 96, got 8
8
  import pandas._libs.sparse as splib
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/core/window.py:36: RuntimeWarning: numpy.dtype size
changed, may indicate binary incompatibility. Expected 96, got 88
  import pandas._libs.window as _window
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/core/groupby/groupby.py:68: RuntimeWarning: numpy.d
type size changed, may indicate binary incompatibility. Expected 96, go
t 88
  from pandas._libs import (lib, reduction,
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/core/reshape/reshape.py:30: RuntimeWarning: numpy.d
type size changed, may indicate binary incompatibility. Expected 96, go
t 88
  from pandas._libs import algos as _algos, reshape as _reshape
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/io/parsers.py:45: RuntimeWarning: numpy.dtype size
changed, may indicate binary incompatibility. Expected 96, got 88
  import pandas._libs.parsers as parsers
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/pandas/io/pytables.py:50: RuntimeWarning: numpy.dtype size
changed, may indicate binary incompatibility. Expected 96, got 88
  from pandas._libs import algos, lib, writers as libwriters
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/scipy/sparse/lil.py:19: RuntimeWarning: numpy.dtype size c
hanged, may indicate binary incompatibility. Expected 96, got 88
  from . import _csparsetools
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/scipy/sparse/csgraph/__init__.py:165: RuntimeWarning: nump
y.dtype size changed, may indicate binary incompatibility. Expected 96,
got 88
  from ._shortest_path import shortest_path, floyd_warshall, dijkstra,\
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/scipy/sparse/csgraph/_validation.py:5: RuntimeWarning: num
py.dtype size changed, may indicate binary incompatibility. Expected 9
6, got 88
  from ._tools import csgraph_to_dense, csgraph_from_dense,\
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/scipy/sparse/csgraph/__init__.py:167: RuntimeWarning: nump
y.dtype size changed, may indicate binary incompatibility. Expected 96,
got 88
  from ._traversal import breadth_first_order, depth_first_order, \
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/scipy/sparse/csgraph/__init__.py:169: RuntimeWarning: nump
y.dtype size changed, may indicate binary incompatibility. Expected 96,
got 88
  from ._min_spanning_tree import minimum_spanning_tree
/gpfs/fs01/user/s0f2-ba03446bdf62cc-bd5847e99873/.local/lib/python2.7/s
ite-packages/scipy/sparse/csgraph/__init__.py:170: RuntimeWarning: nump
y.dtype size changed, may indicate binary incompatibility. Expected 96,
got 88
  from ._reordering import reverse_cuthill_mckee, maximum_bipartite_mat
ching, \
```

Then we create an MLContext to interface with Apache SystemML. Note that we pass a SparkSession object as parameter so SystemML now knows how to talk to the Apache Spark cluster

In [3]:

```
ml = MLContext(spark)
```

Now we create some large random matrices to have numpy and SystemML crunch on it

In [4]:

```
u = np.random.rand(1000,10000)
s = np.random.rand(10000,1000)
w = np.random.rand(1000,1000)
```

Now we implement a short one-liner to define a very simple linear algebra operation

In case you are not familiar with matrix-matrix multiplication: https://en.wikipedia.org/wiki/Matrix_multiplication (https://en.wikipedia.org/wiki/Matrix_multiplication)

sum(U' *(W . (U* S)))

Legend	
'	transpose of a matrix
*	matrix-matrix multiplication
.	scalar multiplication

In [5]:

```
start = time.time()
res = np.sum(u.T.dot(w * u.dot(s)))
print time.time()-start
```

```
0.197705030441
```

As you can see this executes perfectly fine. Note that this is even a very efficient execution because numpy uses a C/C++ backend which is known for it's performance. But what happens if U, S or W get such big that the available main memory cannot cope with it? Let's give it a try:

In [6]:

```
'''
u = np.random.rand(10000,100000)
s = np.random.rand(100000,10000)
w = np.random.rand(10000,10000)
'''
```

In [12]:

```
u = np.random.rand(10000,1100000)
s = np.random.rand(1100000,10000)
w = np.random.rand(10000,10000)
```

```
MemoryErrorTraceback (most recent call last)
<ipython-input-12-b69efdec6149> in <module>()
----> 1 u = np.random.rand(10000,1100000)
      2 s = np.random.rand(1100000,10000)
      3 w = np.random.rand(10000,10000)

mtrand.pyx in mtrand.RandomState.rand()

mtrand.pyx in mtrand.RandomState.random_sample()

mtrand.pyx in mtrand.cont0_array()
mtrand.pyx in mtrand.cont0_array()

MemoryError:
```

After a short while you should see a memory error. This is because the operating system process was not able to allocate enough memory for storing the numpy array on the heap. Now it's time to re-implement the very same operations as DML in SystemML, and this is your task. Just replace all ###your_code_goes_here sections with proper code, please consider the following table which contains all DML syntax you need:

Syntax	
t(M)	transpose of a matrix, where M is the matrix
%*%	matrix-matrix multiplication
*	scalar multiplication

Task

In [13]:

```
#res = np.sum(u.T.dot(w * u.dot(s)))
#res = sum(###your_code_goes_here(U) %*% (W * (U ###your_code_goes_here
 S)))
script = """
res = sum( t(U) %*% (W * (U %*% S)))
"""
```

To get consistent results we switch from a random matrix initialization to something deterministic

In [14]:

```
u = np.arange(100000).reshape((100, 1000))
s = np.arange(100000).reshape((1000, 100))
w = np.arange(10000).reshape((100, 100))
```

In [15]:

```
prog = dml(script).input('U', u).input('S', s).input('W', w).output('re
s')
res = ml.execute(prog).get('res')
print
```