DATSCIW261 ASSIGNMENT 13

MIDS UC Berkeley, Machine Learning at Scale

AUTHORS : Rajesh Thallam
EMAIL : rajesh.thallam@ischool.berkeley.edu
WEEK : 13
DATE : 09-Dec-15

HW13.1

Spark implementation of basic PageRank

Write a basic Spark implementation of the iterative PageRank algorithm that takes sparse adjacency lists as input. Make sure that your implementation utilizes teleportation (1-damping/the number of nodes in the network), and further, distributes the mass of dangling nodes with each iteration so that the output of each iteration is correctly normalized (sums to 1).

[NOTE: The PageRank algorithm assumes that a random surfer (walker), starting from a random web page, chooses the next page to which it will move by clicking at random, with probability d, one of the hyperlinks in the current page. This probability is represented by a so-called 'damping factor' d, where d $\in$ (0, 1). Otherwise, with probability (1 – d), the surfer jumps to any web page in the network. If a page is a dangling end, meaning it has no outgoing hyperlinks, the random surfer selects an arbitrary web page from a uniform distribution and "teleports" to that page]

In your Spark solution, please use broadcast variables and caching to make sure your code is as efficient as possible.

As you build your code, use the following [test data to](s3://ucb-mids-mls-networks/PageRank-test.txt) check you implementation:

Set the teleportation parameter to 0.15 (1-d, where d, the damping factor is set to 0.85), and crosscheck your work with the true result, displayed in the first image in the Wikipedia articlehttps [PageRank] (//en.wikipedia.org/wiki/PageRank) and here for reference are the corresponding resulting PageRank probabilities:

A,0.033
B,0.384
C,0.343
D,0.039
E,0.081
F,0.039
G,0.016
H,0.016
I,0.016
J,0.016
K,0.016

Run this experiment locally first. Report the local configuration that you used and how long in minutes and seconds it takes to complete your job.

Repeat this experiment on AWS. Report the AWS cluster configuration that you used and how long in minutes and seconds it takes to complete your job. (in your notebook, cat the cluster config file)

Algorithm

```
Stage 1
Function Map(Pi, Value)
    #Value contains the url of a page and one of its outlinks:[Pi Pik]
1: output(Pi; Pik)
2: output(Pik; "")

Function Reduce(Text Key, Text Values[]
   #For Key = Pi, Values contains list of outlinks of P[Pi0 Pi1 Pi2 ...]
3: Outlinks <- Ranki(Initial Rank)
4: for each element Value in Values
5:   Outlinks += Value // add Value to Outlinks String
6: end for
7: output(Pi, Outlinks)
```

```
Function Stage-1-Map(Text Pi, Text Value)
    #Value contains the rank of page Pi and its outlinks: [Ri Pi0 Pi1 Pi2
...]
1:  if page Pi has outlinks then
2:      for each outlink Pk in Value
3:          Ni = Number of outlinks
4:          output(Pk, (Ri + r + (1-a)/N)/Ni)
5:      end for
6:      output(Pi, "m" Pi0 Pi1 Pi2 ...)(m indicates that the value is the
list of outlinks)
7:   else if page Pi doesn't have outlinks then
8:      output(-1, Ri + r + (1-a)/N)
9:      output(Pi, "m")
10:  end if

Function
   Stage-1-Reduce(Text Key, Text Values[])
     #For Key = -1, Values contains Rank contributions of pages without ou
tlinks -> [Rn0 Rn1 Rn2 ...]
     #For Key = P, k, Values contains list of outlinks of Pk and rank cont
ributions to Pk from other pages -> [[m Pi0 Pi1 Pi2 ...] R0/N0 R1/N1 R2/N2
...]
11:
12:   if Key = -1 then
13:       r <- 0
14:       for each element Rni in Values
15:           r += Rni
16:       end for
17:       r = a * r/N  //N is the number of total pages
18:       Write r into a HDFS file
19:   else
20:       rk <- 0
21:       for each element Value in Values
22:           if Value is the list of outlinks then
23:               Outlinks <- Value delete m
24:           else
25:               rk += Ri/Ni
26:           end if
27:       end for
28:       rk = a * rk
29:       output(Pk, rk Outlinks)
30:   end if
```

In [87]:

```
import os
import sys #current as of 9/26/2015
spark_home = os.environ['SPARK_HOME'] = '/usr/local/spark'

if not spark_home:
    raise ValueError('SPARK_HOME enviroment variable is not set')
sys.path.insert(0,os.path.join(spark_home,'python'))
sys.path.insert(0,os.path.join(spark_home,'python/lib/py4j-0.8.2.1-src.zip'))
execfile(os.path.join(spark_home,'python/pyspark/shell.py'))
```

```
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 1.5.2
      /_/

Using Python version 2.7.6 (default, Jun 22 2015 17:58:13)
SparkContext available as sc, HiveContext available as sqlContext.
```

Pagerank implemetation for toyset

In [194]:

```
%%writefile pagerank_13_1.py
#!/usr/bin/python
import re
import sys
import os
import sys
import ast
from operator import add

from pyspark import SparkContext

def pagerank_init(line):
    # initialize page rank as 1/N for all nodes with
    # outgoing links and emit with graph structure
    node, ol = line.split('\t')
    neighbors = '|'.join(ast.literal_eval(ol).keys())
    yield node.encode('utf-8'), [1/N, neighbors]

def distribute(node, rank_links):
    """Calculates URL contributions to the rank of other URLs."""
    r = rank_links[0]
    links = rank_links[1]

    ol = str(links).split('|')
    Ni = len(ol)

    # if the node is for dangling (i.e. no outgoing link),
```

```
    # emit the loss to redistribute to all the incoming

    # links to the dangling node
    if (Ni == 1 and ol[0] == '') or Ni == 0:
        yield 'DANGLING', r
    else:
        r_new = float(r)/float(Ni)
        for l in ol:
            yield l, r_new

    # recover graph structure
    if links <> '':
        yield node, links

# update pagerank by combining the mass
def combine_mass(rank_links):
    r = 0.0
    out = ''

    for i in rank_links.split('~'):
        try:
            i = ast.literal_eval(i)
            if type(i) == float:
                r += i
            else:
                out = i if i else out
        except:
            out = i if i else out
            pass

    return str(r) + '~' + str(out)

def update_pagerank(node, rank_links, loss, N, a = 0.15):
    r = 0.0
    out_links = ""

    for i in str(rank_links).split('~'):
        try:
            i = ast.literal_eval(i)
            if type(i) == float:
                r = float(i)
            else:
                out_links = i if i else out_links
        except:
            out_links = i if i else out_links
            pass

    r_new = a * (1/N) + (1-a) * (loss/N + r)
    return node, [round(r_new, 5), out_links]

if __name__ == "__main__":
    if len(sys.argv) != 4:
        print("Usage: pagerank <source_file> <iterations> <target_file>")
        exit(-1)
```

```
    # Initialize the spark context.
    sc = SparkContext(appName="PythonPageRank")

    lines = sc.textFile(sys.argv[1], 1)
    N = 11.0
    D = 0.85
    a = 0.15

    # parse and initialize pagerank
    ranks = lines.flatMap(lambda pages: pagerank_init(pages))

    for iteration in range(int(sys.argv[2])):
        # contribution from each page
        contribs = ranks \
                    .flatMap(lambda (node, rank_links): distribute(node, rank_li
nks)) \
                    .reduceByKey(lambda prev, curr: combine_mass(str(prev) + '~'
+ str(curr))).cache()

        # find dangling mass
        dangling_nodes = contribs.lookup('DANGLING')
        dangling_mass = 0.0 if len(dangling_nodes) == 0 else float(str(dangling_
nodes[0]).strip('~'))

        # update page rank
        ranks_new = contribs \
                    .filter(lambda (k, v): k != 'DANGLING') \
                    .map(lambda (node, rank_links): update_pagerank(node, rank_l
inks, dangling_mass, N, a))
        ranks = ranks_new

    ranks \
        .map(lambda (node, rank_links): (node, round(rank_links[0], 3), rank_lin
ks[1])) \
        .saveAsTextFile(sys.argv[3])

    sc.stop()
```

```
Overwriting pagerank_13_1.py
```

Running on Local

In [185]:

```
#!/usr/bin/python
import time

start_time = time.time()

!rm -fR out_hw13_1
!time $SPARK_HOME/bin/spark-submit --name "PythonPageRank" --master local[4]
./pagerank_13_1.py ./PageRank-test.txt 100 out_hw13_1

end_time = time.time()

print "="*80
print "Time taken to find page rank of the network = {:.2f} seconds".format(end_
time - start_time)
print "="*80

print "Pagerank of the graph is"
!cat out_hw13_1/part-000* | sort
```

```
15/12/07 15:17:14 WARN NativeCodeLoader: Unable to load native-hadoo
p library for your platform... using builtin-java classes where appl
icable
15/12/07 15:17:14 WARN Utils: Your hostname, rtubuntu resolves to a
loopback address: 127.0.1.1; using 10.0.2.15 instead (on interface e
th0)
15/12/07 15:17:14 WARN Utils: Set SPARK_LOCAL_IP if you need to bind
to another address
15/12/07 15:17:16 WARN MetricsSystem: Using default name DAGSchedule
r for source because spark.app.id is not set.
22.63user 1.86system 0:35.03elapsed 69%CPU (0avgtext+0avgdata 498340
maxresident)k
0inputs+3104outputs (0major+269468minor)pagefaults 0swaps
================================================================================
Time taken to find page rank of the network = 35.26 seconds
================================================================================
Pagerank of the graph is
('A', 0.033, '')
('B', 0.384, 'C')
('C', 0.343, 'B')
('D', 0.039, 'A|B')
('E', 0.081, 'B|D|F')
('F', 0.039, 'B|E')
('G', 0.016, 'B|E')
('H', 0.016, 'B|E')
('I', 0.016, 'B|E')
('J', 0.016, 'E')
('K', 0.016, 'E')
```

Running on AWS

In [ ]:

```
aws emr create-cluster --name "rt-hw13" --release-label emr-4.2.0 --applications
Name=Spark --ec2-attributes KeyName=rthallam_sa_east --log-uri s3://ucb-mids-mls
-rajeshthallam/hw13/logs --instance-type m3.xlarge  --instance-count 10 --use-de
fault-roles --configurations file://./emr_config_spark_rt.json
```

In [186]:

```
!scp -i ~/rthallam_sa_east.pem ./PageRank-test_indexed.txt hadoop@ec2-54-233-144
-86.sa-east-1.compute.amazonaws.com:/home/hadoop/src
```

```
PageRank-test_indexed.txt                     100%  168     0.2KB/s
00:00
```

In [173]:

```
#!/usr/bin/python
import time

start_time = time.time()

# copying latest script
!scp -i ~/rthallam_sa_east.pem ./pagerank_13_1.py hadoop@ec2-54-233-144-86.sa-ea
st-1.compute.amazonaws.com:/home/hadoop/src
# removing target directory
!aws s3 rm s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/ --recursive
# launching script
!ssh -i ~/rthallam_sa_east.pem hadoop@ec2-54-233-144-86.sa-east-1.compute.amazon
aws.com /usr/lib/spark/bin/spark-submit --master yarn-cluster /home/hadoop/src/p
agerank_13_1.py s3n://ucb-mids-mls-networks/PageRank-test.txt 100 s3n://ucb-mids
-mls-rajeshthallam/hw13/results/hw13_1

end_time = time.time()

print "="*80
print "Time taken to find page rank of the network = {:.2f} seconds".format(end_
time - start_time)
print "="*80

print "Pagerank of the graph is"
!rm -f ./out_hw13_1/part*
!aws s3 cp s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/ ./out_hw13_1 --r
ecursive
!cat out_hw13_1/part-000* | sort
```

```
pagerank_13_1.py                              100% 3062     3.0KB/s
00:00
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/_SUCCESS
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
```

```
09
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
10
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
00
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
01
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
05
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
06
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
04
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
07
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
11
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
08
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
12
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
13
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
15
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
16
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
14
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
17
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
18
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
19
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
20
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
23
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
21
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
22
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
25
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
24
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
27
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
26
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
28
```

```
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
29
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
30
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
32
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
33
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
34
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
35
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
31
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
03
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
02
15/12/07 22:49:53 WARN util.NativeCodeLoader: Unable to load native-
hadoop library for your platform... using builtin-java classes where
applicable
15/12/07 22:49:53 INFO client.RMProxy: Connecting to ResourceManager
at ip-172-31-32-212.sa-east-1.compute.internal/172.31.32.212:8032
15/12/07 22:49:53 INFO yarn.Client: Requesting a new application fro
m cluster with 9 NodeManagers
15/12/07 22:49:53 INFO yarn.Client: Verifying our application has no
t requested more than the maximum memory capability of the cluster (
11520 MB per container)
15/12/07 22:49:53 INFO yarn.Client: Will allocate AM container, with
1408 MB memory including 384 MB overhead
15/12/07 22:49:53 INFO yarn.Client: Setting up container launch cont
ext for our AM
15/12/07 22:49:53 INFO yarn.Client: Setting up the launch environmen
t for our AM container
15/12/07 22:49:53 INFO yarn.Client: Preparing resources for our AM c
ontainer
15/12/07 22:49:54 INFO yarn.Client: Uploading resource file:/usr/lib
/spark/lib/spark-assembly-1.5.2-hadoop2.6.0-amzn-2.jar -> hdfs://ip-
172-31-32-212.sa-east-1.compute.internal:8020/user/hadoop/.sparkStag
ing/application_1449482525945_0023/spark-assembly-1.5.2-hadoop2.6.0-
amzn-2.jar
15/12/07 22:49:54 INFO metrics.MetricsSaver: MetricsConfigRecord dis
abledInCluster: false instanceEngineCycleSec: 60 clusterEngineCycleS
ec: 60 disableClusterEngine: false maxMemoryMb: 3072 maxInstanceCoun
t: 500 lastModified: 1449482533009
15/12/07 22:49:54 INFO metrics.MetricsSaver: Created MetricsSaver j-
KBN00RIHUZBE:i-d5952e37:SparkSubmit:31545 period:60 /mnt/var/em/raw/
i-d5952e37_20151207_SparkSubmit_31545_raw.bin
15/12/07 22:49:56 INFO metrics.MetricsSaver: 1 aggregated HDFSWriteD
elay 2651 raw values into 1 aggregated values, total 1
15/12/07 22:49:56 INFO yarn.Client: Uploading resource file:/home/ha
doop/src/pagerank_13_1.py -> hdfs://ip-172-31-32-212.sa-east-1.compu
te.internal:8020/user/hadoop/.sparkStaging/application_1449482525945
```

```
_0023/pagerank_13_1.py
15/12/07 22:49:56 INFO yarn.Client: Uploading resource file:/usr/lib
/spark/python/lib/pyspark.zip -> hdfs://ip-172-31-32-212.sa-east-1.c
ompute.internal:8020/user/hadoop/.sparkStaging/application_144948252
5945_0023/pyspark.zip
15/12/07 22:49:56 INFO yarn.Client: Uploading resource file:/usr/lib
/spark/python/lib/py4j-0.8.2.1-src.zip -> hdfs://ip-172-31-32-212.sa
-east-1.compute.internal:8020/user/hadoop/.sparkStaging/application_
1449482525945_0023/py4j-0.8.2.1-src.zip
15/12/07 22:49:56 INFO yarn.Client: Uploading resource file:/tmp/spa
rk-da6da678-31a0-4307-bd5d-f4e28422910a/__spark_conf__72062871229811
83140.zip -> hdfs://ip-172-31-32-212.sa-east-1.compute.internal:8020
/user/hadoop/.sparkStaging/application_1449482525945_0023/__spark_co
nf__7206287122981183140.zip
15/12/07 22:49:56 INFO spark.SecurityManager: Changing view acls to:
hadoop
15/12/07 22:49:56 INFO spark.SecurityManager: Changing modify acls t
o: hadoop
15/12/07 22:49:56 INFO spark.SecurityManager: SecurityManager: authe
ntication disabled; ui acls disabled; users with view permissions: S
et(hadoop); users with modify permissions: Set(hadoop)
15/12/07 22:49:56 INFO yarn.Client: Submitting application 23 to Res
ourceManager
15/12/07 22:49:56 INFO impl.YarnClientImpl: Submitted application ap
plication_1449482525945_0023
15/12/07 22:49:57 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: ACCEPTED)
15/12/07 22:49:57 INFO yarn.Client:
         client token: N/A
         diagnostics: N/A
         ApplicationMaster host: N/A
         ApplicationMaster RPC port: -1
         queue: default
         start time: 1449528596441
         final status: UNDEFINED
         tracking URL: http://ip-172-31-32-212.sa-east-1.compute.int
ernal:20888/proxy/application_1449482525945_0023/
         user: hadoop
15/12/07 22:49:58 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: ACCEPTED)
15/12/07 22:49:59 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: ACCEPTED)
15/12/07 22:50:00 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: ACCEPTED)
15/12/07 22:50:01 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: ACCEPTED)
15/12/07 22:50:02 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: ACCEPTED)
15/12/07 22:50:03 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:03 INFO yarn.Client:
         client token: N/A
         diagnostics: N/A
```

```
         ApplicationMaster host: 172.31.42.129
         ApplicationMaster RPC port: 0
         queue: default
         start time: 1449528596441
         final status: UNDEFINED
         tracking URL: http://ip-172-31-32-212.sa-east-1.compute.int
ernal:20888/proxy/application_1449482525945_0023/
         user: hadoop
15/12/07 22:50:04 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:05 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:06 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:07 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:08 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:09 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:10 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:11 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:12 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:13 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:14 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:15 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:16 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:17 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:18 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:19 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:20 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:21 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:22 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:23 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:24 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:25 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:26 INFO yarn.Client: Application report for applicati
```

```
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:27 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:28 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:29 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:30 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:31 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:32 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:33 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:34 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:35 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:36 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:37 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:38 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:39 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:40 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:41 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:42 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:43 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:44 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:45 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:46 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:47 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:48 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:49 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:50 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:51 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:52 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
```

```
15/12/07 22:50:53 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:54 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:55 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:56 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:57 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:58 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:50:59 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:00 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:01 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:02 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:03 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:04 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:05 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:06 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:07 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:08 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:09 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:10 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:11 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:12 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:13 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: RUNNING)
15/12/07 22:51:14 INFO yarn.Client: Application report for applicati
on_1449482525945_0023 (state: FINISHED)
15/12/07 22:51:14 INFO yarn.Client:
         client token: N/A
         diagnostics: N/A
         ApplicationMaster host: 172.31.42.129
         ApplicationMaster RPC port: 0
         queue: default
         start time: 1449528596441
         final status: SUCCEEDED
         tracking URL: http://ip-172-31-32-212.sa-east-1.compute.int
```

```
ernal:20888/proxy/application_1449482525945_0023/history/application
_1449482525945_0023/1
         user: hadoop
15/12/07 22:51:14 INFO util.ShutdownHookManager: Shutdown hook calle
d
15/12/07 22:51:14 INFO util.ShutdownHookManager: Deleting directory
/tmp/spark-da6da678-31a0-4307-bd5d-f4e28422910a
================================================================
============
Time taken to find page rank of the network = 98.49 seconds
================================================================
============
Pagerank of the graph is
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0001 to out_hw13_1/part-00001
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0007 to out_hw13_1/part-00007
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0000 to out_hw13_1/part-00000
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/_SUCCE
SS to out_hw13_1/_SUCCESS
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0004 to out_hw13_1/part-00004
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0006 to out_hw13_1/part-00006
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0008 to out_hw13_1/part-00008
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0009 to out_hw13_1/part-00009
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0010 to out_hw13_1/part-00010
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0014 to out_hw13_1/part-00014
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0015 to out_hw13_1/part-00015
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0011 to out_hw13_1/part-00011
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0012 to out_hw13_1/part-00012
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0013 to out_hw13_1/part-00013
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0016 to out_hw13_1/part-00016
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0017 to out_hw13_1/part-00017
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0022 to out_hw13_1/part-00022
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0021 to out_hw13_1/part-00021
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0018 to out_hw13_1/part-00018
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0020 to out_hw13_1/part-00020
```

```
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0019 to out_hw13_1/part-00019
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0023 to out_hw13_1/part-00023
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0024 to out_hw13_1/part-00024
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0026 to out_hw13_1/part-00026
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0029 to out_hw13_1/part-00029
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0027 to out_hw13_1/part-00027
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0025 to out_hw13_1/part-00025
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0028 to out_hw13_1/part-00028
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0030 to out_hw13_1/part-00030
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0031 to out_hw13_1/part-00031
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0032 to out_hw13_1/part-00032
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0033 to out_hw13_1/part-00033
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0035 to out_hw13_1/part-00035
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0034 to out_hw13_1/part-00034
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0005 to out_hw13_1/part-00005
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0002 to out_hw13_1/part-00002
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0003 to out_hw13_1/part-00003
('A', 0.033, '')
('B', 0.384, 'C')
('C', 0.343, 'B')
('D', 0.039, 'A|B')
('E', 0.081, 'B|D|F')
('F', 0.039, 'B|E')
('G', 0.016, 'B|E')
('H', 0.016, 'B|E')
('I', 0.016, 'B|E')
('J', 0.016, 'E')
('K', 0.016, 'E')
```

HW13.2

Applying PageRank to the Wikipedia hyperlinks network

Run your Spark PageRank implementation on the Wikipedia dataset for 10 iterations, and display the top 100 ranked nodes (with alpha = 0.85).

Run your PageRank implementation on the Wikipedia dataset for 50 iterations, and display the top 100 ranked nodes (with teleportation factor of 0.15).

Plot the pagerank values for the top 100 pages resulting from the 50 iterations run. Then plot the pagerank values for the same 100 pages that resulted from the 10 iterations run. Comment on your findings. Have the top 100 ranked pages changed? Have the pagerank values changed? Explain.

Report the AWS cluster configuration that you used and how long in minutes and seconds it takes to complete your job.

NOTE: Wikipedia data is located on S3 at
-- s3://ucb-mids-mls-networks/wikipedia/
-- s3://ucb-mids-mls-networks/wikipedia/all-pages-indexed-out.txt # Graph
-- s3://ucb-mids-mls-networks/wikipedia/indices.txt # Page titles and page Ids

Cluster Creation for Wiki Page Rank

In [ ]:

```
aws emr create-cluster --name "rt-hw13" \
    --release-label emr-4.2.0 \
    --applications Name=Spark \
    --ec2-attributes KeyName=rthallam_sa_east \
    --log-uri s3://ucb-mids-mls-rajeshthallam/hw13/logs \
    --instance-type m3.xlarge  \
    --instance-count 10 \
    --use-default-roles \
    --configurations file://./emr_config_spark_rt.json \
    --bootstrap-actions Path=s3://ucb-mids-mls-rajeshthallam/bootstrap_actions.s
h
```

Running with indexed toy data set

In [189]:

```
#!/usr/bin/python
import time

start_time = time.time()

# copying latest script
!scp -i ~/rthallam_sa_east.pem ./pagerank_13_1.py hadoop@ec2-54-233-144-86.sa-ea
st-1.compute.amazonaws.com:/home/hadoop/src
# removing target directory
!aws s3 rm s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/ --recursive
# launching script
!ssh -i ~/rthallam_sa_east.pem hadoop@ec2-54-233-144-86.sa-east-1.compute.amazon
aws.com /usr/lib/spark/bin/spark-submit --master yarn-cluster /home/hadoop/src/p
agerank_13_1.py s3n://ucb-mids-mls-rajeshthallam/hw13/PageRank-test_indexed.txt
100 s3n://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1

end_time = time.time()

print "="*80
print "Time taken to find page rank of the network = {:.2f} seconds".format(end_
time - start_time)
print "="*80

print "Pagerank of the graph is"
!rm -f ./out_hw13_1/part*
!aws s3 cp s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/ ./out_hw13_1 --r
ecursive
!cat out_hw13_1/part-000* | sort
```

```
pagerank_13_1.py                                  100% 3220     3.1KB/s
00:00
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/_SUCCESS
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
05
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
10
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
09
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
11
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
12
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
13
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
15
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
14
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
17
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
```

```
16
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
19
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
20
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
02
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
07
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
01
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
21
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
22
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
18
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
23
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
24
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
25
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
26
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
27
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
30
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
28
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
32
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
31
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
08
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
33
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
29
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
35
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
34
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
00
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
04
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
03
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-000
06
```

```
15/12/07 23:34:12 WARN util.NativeCodeLoader: Unable to load native-
hadoop library for your platform... using builtin-java classes where
applicable
15/12/07 23:34:12 INFO client.RMProxy: Connecting to ResourceManager
at ip-172-31-32-212.sa-east-1.compute.internal/172.31.32.212:8032
15/12/07 23:34:12 INFO yarn.Client: Requesting a new application fro
m cluster with 9 NodeManagers
15/12/07 23:34:12 INFO yarn.Client: Verifying our application has no
t requested more than the maximum memory capability of the cluster (
11520 MB per container)
15/12/07 23:34:12 INFO yarn.Client: Will allocate AM container, with
1408 MB memory including 384 MB overhead
15/12/07 23:34:12 INFO yarn.Client: Setting up container launch cont
ext for our AM
15/12/07 23:34:12 INFO yarn.Client: Setting up the launch environmen
t for our AM container
15/12/07 23:34:12 INFO yarn.Client: Preparing resources for our AM c
ontainer
15/12/07 23:34:13 INFO yarn.Client: Uploading resource file:/usr/lib
/spark/lib/spark-assembly-1.5.2-hadoop2.6.0-amzn-2.jar -> hdfs://ip-
172-31-32-212.sa-east-1.compute.internal:8020/user/hadoop/.sparkStag
ing/application_1449482525945_0027/spark-assembly-1.5.2-hadoop2.6.0-
amzn-2.jar
15/12/07 23:34:13 INFO metrics.MetricsSaver: MetricsConfigRecord dis
abledInCluster: false instanceEngineCycleSec: 60 clusterEngineCycleS
ec: 60 disableClusterEngine: false maxMemoryMb: 3072 maxInstanceCoun
t: 500 lastModified: 1449482533009
15/12/07 23:34:13 INFO metrics.MetricsSaver: Created MetricsSaver j-
KBN00RIHUZBE:i-d5952e37:SparkSubmit:26959 period:60 /mnt/var/em/raw/
i-d5952e37_20151207_SparkSubmit_26959_raw.bin
15/12/07 23:34:15 INFO metrics.MetricsSaver: 1 aggregated HDFSWriteD
elay 2650 raw values into 1 aggregated values, total 1
15/12/07 23:34:15 INFO yarn.Client: Uploading resource file:/home/ha
doop/src/pagerank_13_1.py -> hdfs://ip-172-31-32-212.sa-east-1.compu
te.internal:8020/user/hadoop/.sparkStaging/application_1449482525945
_0027/pagerank_13_1.py
15/12/07 23:34:15 INFO yarn.Client: Uploading resource file:/usr/lib
/spark/python/lib/pyspark.zip -> hdfs://ip-172-31-32-212.sa-east-1.c
ompute.internal:8020/user/hadoop/.sparkStaging/application_144948252
5945_0027/pyspark.zip
15/12/07 23:34:15 INFO yarn.Client: Uploading resource file:/usr/lib
/spark/python/lib/py4j-0.8.2.1-src.zip -> hdfs://ip-172-31-32-212.sa
-east-1.compute.internal:8020/user/hadoop/.sparkStaging/application_
1449482525945_0027/py4j-0.8.2.1-src.zip
15/12/07 23:34:15 INFO yarn.Client: Uploading resource file:/tmp/spa
rk-07519b58-7cf3-4b2c-89ab-78820cae8125/__spark_conf__88557622664353
51902.zip -> hdfs://ip-172-31-32-212.sa-east-1.compute.internal:8020
/user/hadoop/.sparkStaging/application_1449482525945_0027/__spark_co
nf__8855762266435351902.zip
15/12/07 23:34:15 INFO spark.SecurityManager: Changing view acls to:
hadoop
15/12/07 23:34:15 INFO spark.SecurityManager: Changing modify acls t
o: hadoop
```

```
15/12/07 23:34:15 INFO spark.SecurityManager: SecurityManager: authe
ntication disabled; ui acls disabled; users with view permissions: S
et(hadoop); users with modify permissions: Set(hadoop)
15/12/07 23:34:15 INFO yarn.Client: Submitting application 27 to Res
ourceManager
15/12/07 23:34:15 INFO impl.YarnClientImpl: Submitted application ap
plication_1449482525945_0027
15/12/07 23:34:16 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: ACCEPTED)
15/12/07 23:34:16 INFO yarn.Client:
         client token: N/A
         diagnostics: N/A
         ApplicationMaster host: N/A
         ApplicationMaster RPC port: -1
         queue: default
         start time: 1449531255431
         final status: UNDEFINED
         tracking URL: http://ip-172-31-32-212.sa-east-1.compute.int
ernal:20888/proxy/application_1449482525945_0027/
         user: hadoop
15/12/07 23:34:17 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: ACCEPTED)
15/12/07 23:34:18 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: ACCEPTED)
15/12/07 23:34:19 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: ACCEPTED)
15/12/07 23:34:20 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: ACCEPTED)
15/12/07 23:34:21 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: ACCEPTED)
15/12/07 23:34:22 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:22 INFO yarn.Client:
         client token: N/A
         diagnostics: N/A
         ApplicationMaster host: 172.31.42.131
         ApplicationMaster RPC port: 0
         queue: default
         start time: 1449531255431
         final status: UNDEFINED
         tracking URL: http://ip-172-31-32-212.sa-east-1.compute.int
ernal:20888/proxy/application_1449482525945_0027/
         user: hadoop
15/12/07 23:34:23 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:24 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:25 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:26 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:27 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
```

```
15/12/07 23:34:28 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:29 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:30 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:31 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:32 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:33 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:34 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:35 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:36 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:37 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:38 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:39 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:40 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:41 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:42 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:43 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:44 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:45 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:46 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:47 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:48 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:49 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:50 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:51 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:52 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:53 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:54 INFO yarn.Client: Application report for applicati
```

```
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:55 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:56 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:57 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:58 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:34:59 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:00 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:01 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:02 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:03 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:04 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:05 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:06 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:07 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:08 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:09 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:10 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:11 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:12 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:13 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:14 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:15 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:16 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:17 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:18 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:19 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:20 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
```

```
15/12/07 23:35:21 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:22 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:23 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:24 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:25 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:26 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:27 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:28 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: RUNNING)
15/12/07 23:35:29 INFO yarn.Client: Application report for applicati
on_1449482525945_0027 (state: FINISHED)
15/12/07 23:35:29 INFO yarn.Client:
         client token: N/A
         diagnostics: N/A
         ApplicationMaster host: 172.31.42.131
         ApplicationMaster RPC port: 0
         queue: default
         start time: 1449531255431
         final status: SUCCEEDED
         tracking URL: http://ip-172-31-32-212.sa-east-1.compute.int
ernal:20888/proxy/application_1449482525945_0027/history/application
_1449482525945_0027/1
         user: hadoop
15/12/07 23:35:29 INFO yarn.Client: Deleting staging directory .spar
kStaging/application_1449482525945_0027
15/12/07 23:35:29 INFO util.ShutdownHookManager: Shutdown hook calle
d
15/12/07 23:35:29 INFO util.ShutdownHookManager: Deleting directory
/tmp/spark-07519b58-7cf3-4b2c-89ab-78820cae8125
================================================================
============
Time taken to find page rank of the network = 98.21 seconds
================================================================
============
Pagerank of the graph is
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0000 to out_hw13_1/part-00000
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0001 to out_hw13_1/part-00001
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0003 to out_hw13_1/part-00003
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0005 to out_hw13_1/part-00005
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0007 to out_hw13_1/part-00007
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
```

```
0008 to out_hw13_1/part-00008
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0002 to out_hw13_1/part-00002
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/_SUCCE
SS to out_hw13_1/_SUCCESS
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0009 to out_hw13_1/part-00009
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0010 to out_hw13_1/part-00010
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0011 to out_hw13_1/part-00011
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0004 to out_hw13_1/part-00004
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0006 to out_hw13_1/part-00006
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0015 to out_hw13_1/part-00015
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0014 to out_hw13_1/part-00014
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0017 to out_hw13_1/part-00017
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0020 to out_hw13_1/part-00020
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0023 to out_hw13_1/part-00023
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0021 to out_hw13_1/part-00021
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0024 to out_hw13_1/part-00024
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0022 to out_hw13_1/part-00022
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0012 to out_hw13_1/part-00012
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0025 to out_hw13_1/part-00025
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0026 to out_hw13_1/part-00026
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0028 to out_hw13_1/part-00028
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0030 to out_hw13_1/part-00030
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0027 to out_hw13_1/part-00027
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0031 to out_hw13_1/part-00031
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0032 to out_hw13_1/part-00032
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0033 to out_hw13_1/part-00033
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0029 to out_hw13_1/part-00029
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0034 to out_hw13_1/part-00034
```

```
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0035 to out_hw13_1/part-00035
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0013 to out_hw13_1/part-00013
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0016 to out_hw13_1/part-00016
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0018 to out_hw13_1/part-00018
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_1/part-0
0019 to out_hw13_1/part-00019
('10', 0.016, 5)
('1', 0.033, '')
('11', 0.016, 5)
('2', 0.384, 3)
('3', 0.343, 2)
('4', 0.039, '1|2')
('5', 0.081, '2|4|6')
('6', 0.039, '2|5')
('7', 0.016, '2|5')
('8', 0.016, '2|5')
('9', 0.016, '2|5')
```

Pagerank on Wikipedia data set

In [7]:

```
%%writefile pagerank_13_2.py
#!/usr/bin/python
import re
import sys
import os
import sys
import ast
from operator import add

from pyspark import SparkContext

def pagerank_init(line):
    # initialize page rank as 1/N for all nodes with
    # outgoing links and emit with graph structure
    node, ol = line.split('\t')
    neighbors = '|'.join(ast.literal_eval(ol).keys())
    yield node.encode('utf-8'), [1/N, neighbors]

def distribute(node, rank_links):
    """Calculates URL contributions to the rank of other URLs."""
    r = rank_links[0]
    links = rank_links[1]

    ol = str(links).split('|')
    Ni = len(ol)

    # if the node is for dangling (i.e. no outgoing link),
```

```
    # emit the loss to redistribute to all the incoming

    # links to the dangling node
    if (Ni == 1 and ol[0] == '') or Ni == 0:
        yield 'DANGLING', r
    else:
        r_new = float(r)/float(Ni)
        for l in ol:
            yield l, r_new

    # recover graph structure
    if links <> '':
        yield node, links

# update pagerank by combining the mass
def combine_mass(rank_links):
    r = 0.0
    out = ''

    for i in rank_links.split('~'):
        try:
            i = ast.literal_eval(i)
            if type(i) == float:
                r += i
            else:
                out = i if i else out
        except:
            out = i if i else out
            pass

    return str(r) + '~' + str(out)

def update_pagerank(node, rank_links, loss, N, a = 0.15):
    r = 0.0
    out_links = ""

    for i in str(rank_links).split('~'):
        try:
            i = ast.literal_eval(i)
            if type(i) == float:
                r = float(i)
            else:
                out_links = i if i else out_links
        except:
            out_links = i if i else out_links
            pass

    r_new = a * (1/N) + (1-a) * (loss/N + r)
    return node, [r_new, out_links]

if __name__ == "__main__":
    if len(sys.argv) != 4:
        print("Usage: pagerank <source_file> <iterations> <target_file>")
        exit(-1)
```

```
    # Initialize the spark context.
    sc = SparkContext(appName="WikiPageRank")

    lines = sc.textFile(sys.argv[1], 1)
    N = 15192277.0
    #N = 11.0
    D = 0.85
    a = 0.15

    # parse and initialize pagerank
    ranks = lines.flatMap(lambda pages: pagerank_init(pages))

    for iteration in range(int(sys.argv[2])):
        # contribution from each page
        contribs = ranks \
                    .flatMap(lambda (node, rank_links): distribute(node, rank_li
nks)) \
                    .reduceByKey(lambda prev, curr: combine_mass(str(prev) + '~'
+ str(curr))).cache()

        # find dangling mass
        dangling_nodes = contribs.lookup('DANGLING')
        dangling_mass = 0.0 if len(dangling_nodes) == 0 else float(str(dangling_
nodes[0]).strip('~'))

        # update page rank
        ranks_new = contribs \
                    .filter(lambda (k, v): k != 'DANGLING') \
                    .map(lambda (node, rank_links): update_pagerank(node, rank_l
inks, dangling_mass, N, a))
        ranks = ranks_new.cache()

        if iteration in [9, 49]:
            top_100 = ranks.top(100, key = lambda (node, rank_links): rank_links
[0])
            sc.parallelize(top_100) \
                .map(lambda (node, rank_links): str(node) + '|' + str(rank_links
[0])) \
                .saveAsTextFile(sys.argv[3] + "/" + str(iteration))

    sc.stop()
```

```
Overwriting pagerank_13_2.py
```

Running Pagerank on Wikipedia data set for 10 iterations

In [ ]:

```
#!/usr/bin/python
import time

start_time = time.time()

# copying latest script
!scp -i ~/rthallam_sa_east.pem ./pagerank_13_2.py hadoop@ec2-54-233-144-86.sa-ea
st-1.compute.amazonaws.com:/home/hadoop/src
# removing target directory
!aws s3 rm s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_10/ --recurs
ive
# launching script
!ssh -i ~/rthallam_sa_east.pem hadoop@ec2-54-233-144-86.sa-east-1.compute.amazon
aws.com /usr/lib/spark/bin/spark-submit --master yarn-cluster /home/hadoop/src/p
agerank_13_2.py s3n://ucb-mids-mls-networks/wikipedia/all-pages-indexed-out.txt
10 s3n://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_10/ > ./hw_13_2_ite
r10.log 2>&1
#!ssh -i ~/rthallam_sa_east.pem hadoop@ec2-54-233-144-86.sa-east-1.compute.amazo
naws.com /usr/lib/spark/bin/spark-submit --master yarn-cluster /home/hadoop/src/
pagerank_13_2.py s3n://ucb-mids-mls-rajeshthallam/hw13/PageRank-test_indexed.txt
10 s3n://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_10/

end_time = time.time()

print "="*80
print "Time taken to find page rank of the network = {:.2f} seconds".format(end_
time - start_time)
print "="*80
```

```
pagerank_13_2.py                              100% 3463     3.4KB/s   00:0
0
15/12/08 02:31:23 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
15/12/08 02:31:24 INFO client.RMProxy: Connecting to ResourceManager at ip
-172-31-32-212.sa-east-1.compute.internal/172.31.32.212:8032
15/12/08 02:31:24 INFO yarn.Client: Requesting a new application from clus
ter with 9 NodeManagers
15/12/08 02:31:24 INFO yarn.Client: Verifying our application has not requ
ested more than the maximum memory capability of the cluster (11520 MB per
container)
15/12/08 02:31:24 INFO yarn.Client: Will allocate AM container, with 1408
MB memory including 384 MB overhead
15/12/08 02:31:24 INFO yarn.Client: Setting up container launch context fo
r our AM
15/12/08 02:31:24 INFO yarn.Client: Setting up the launch environment for
our AM container
```

```
15/12/08 02:31:24 INFO yarn.Client: Preparing resources for our AM contain
er
15/12/08 02:31:24 INFO yarn.Client: Uploading resource file:/usr/lib/spark
/lib/spark-assembly-1.5.2-hadoop2.6.0-amzn-2.jar -> hdfs://ip-172-31-32-21
2.sa-east-1.compute.internal:8020/user/hadoop/.sparkStaging/application_14
49482525945_0035/spark-assembly-1.5.2-hadoop2.6.0-amzn-2.jar
15/12/08 02:31:25 INFO metrics.MetricsSaver: MetricsConfigRecord disabledI
nCluster: false instanceEngineCycleSec: 60 clusterEngineCycleSec: 60 disab
leClusterEngine: false maxMemoryMb: 3072 maxInstanceCount: 500 lastModifie
d: 1449482533009
15/12/08 02:31:25 INFO metrics.MetricsSaver: Created MetricsSaver j-KBN00R
IHUZBE:i-d5952e37:SparkSubmit:03344 period:60 /mnt/var/em/raw/i-d5952e37_2
0151208_SparkSubmit_03344_raw.bin
15/12/08 02:31:26 INFO metrics.MetricsSaver: 1 aggregated HDFSWriteDelay 1
152 raw values into 1 aggregated values, total 1
15/12/08 02:31:26 INFO yarn.Client: Uploading resource file:/home/hadoop/s
rc/pagerank_13_2.py -> hdfs://ip-172-31-32-212.sa-east-1.compute.internal:
8020/user/hadoop/.sparkStaging/application_1449482525945_0035/pagerank_13_
2.py
15/12/08 02:31:26 INFO yarn.Client: Uploading resource file:/usr/lib/spark
/python/lib/pyspark.zip -> hdfs://ip-172-31-32-212.sa-east-1.compute.inter
nal:8020/user/hadoop/.sparkStaging/application_1449482525945_0035/pyspark.
zip
15/12/08 02:31:26 INFO yarn.Client: Uploading resource file:/usr/lib/spark
/python/lib/py4j-0.8.2.1-src.zip -> hdfs://ip-172-31-32-212.sa-east-1.comp
ute.internal:8020/user/hadoop/.sparkStaging/application_1449482525945_0035
/py4j-0.8.2.1-src.zip
15/12/08 02:31:26 INFO yarn.Client: Uploading resource file:/tmp/spark-3c2
f91b6-6d4b-480d-a130-bd8c5bc68322/__spark_conf__8771859733817262757.zip ->
hdfs://ip-172-31-32-212.sa-east-1.compute.internal:8020/user/hadoop/.spark
Staging/application_1449482525945_0035/__spark_conf__8771859733817262757.z
ip
15/12/08 02:31:26 INFO spark.SecurityManager: Changing view acls to: hadoo
p
15/12/08 02:31:26 INFO spark.SecurityManager: Changing modify acls to: had
oop
15/12/08 02:31:26 INFO spark.SecurityManager: SecurityManager: authenticat
ion disabled; ui acls disabled; users with view permissions: Set(hadoop);
users with modify permissions: Set(hadoop)
15/12/08 02:31:26 INFO yarn.Client: Submitting application 35 to ResourceM
anager
15/12/08 02:31:26 INFO impl.YarnClientImpl: Submitted application applicat
ion_1449482525945_0035
15/12/08 02:31:27 INFO yarn.Client: Application report for application_144
9482525945_0035 (state: ACCEPTED)
```

```
...
15/12/08 04:25:32 INFO yarn.Client: Application report for application_144
9482525945_0035 (state: RUNNING)
15/12/08 04:25:33 INFO yarn.Client: Application report for application_144
9482525945_0035 (state: FINISHED)
15/12/08 04:25:33 INFO yarn.Client:
     client token: N/A
     diagnostics: N/A
     ApplicationMaster host: 172.31.42.131
     ApplicationMaster RPC port: 0
     queue: default
     start time: 1449541886956
     final status: SUCCEEDED
     tracking URL: http://ip-172-31-32-212.sa-east-1.compute.internal:2088
8/proxy/application_1449482525945_0035/history/application_1449482525945_0
035/1
     user: hadoop
15/12/08 04:25:33 INFO util.ShutdownHookManager: Shutdown hook called
15/12/08 04:25:33 INFO util.ShutdownHookManager: Deleting directory /tmp/s
park-3c2f91b6-6d4b-480d-a130-bd8c5bc68322
================================================================================
======
Time taken to find page rank of the network = 6866.21 seconds
    ================================================================================
    ======
```

In [7]:

```
!rm -f ./out_hw13_2/iter_10/part*
!aws s3 cp s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_10/9/ ./out_
hw13_2/iter_10/ --recursive
!cat ./out_hw13_2/iter_10/part* > ./out_hw13_2/top100_pr_10iter.txt
!head ./out_hw13_2/top100_pr_10iter.txt
```

```
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/_SUCCESS to out_hw13_2/iter_10/_SUCCESS
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00002 to out_hw13_2/iter_10/part-00002
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00000 to out_hw13_2/iter_10/part-00000
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00004 to out_hw13_2/iter_10/part-00004
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00009 to out_hw13_2/iter_10/part-00009
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00003 to out_hw13_2/iter_10/part-00003
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00005 to out_hw13_2/iter_10/part-00005
```

```
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00006 to out_hw13_2/iter_10/part-00006
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00007 to out_hw13_2/iter_10/part-00007
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00001 to out_hw13_2/iter_10/part-00001
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00010 to out_hw13_2/iter_10/part-00010
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00011 to out_hw13_2/iter_10/part-00011
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00008 to out_hw13_2/iter_10/part-00008
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00013 to out_hw13_2/iter_10/part-00013
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00012 to out_hw13_2/iter_10/part-00012
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00014 to out_hw13_2/iter_10/part-00014
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00015 to out_hw13_2/iter_10/part-00015
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00021 to out_hw13_2/iter_10/part-00021
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00017 to out_hw13_2/iter_10/part-00017
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00019 to out_hw13_2/iter_10/part-00019
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00020 to out_hw13_2/iter_10/part-00020
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00016 to out_hw13_2/iter_10/part-00016
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00022 to out_hw13_2/iter_10/part-00022
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00023 to out_hw13_2/iter_10/part-00023
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00024 to out_hw13_2/iter_10/part-00024
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00018 to out_hw13_2/iter_10/part-00018
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00025 to out_hw13_2/iter_10/part-00025
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00030 to out_hw13_2/iter_10/part-00030
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00031 to out_hw13_2/iter_10/part-00031
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00032 to out_hw13_2/iter_10/part-00032
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00028 to out_hw13_2/iter_10/part-00028
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00029 to out_hw13_2/iter_10/part-00029
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00027 to out_hw13_2/iter_10/part-00027
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
```

```
0/9/part-00026 to out_hw13_2/iter_10/part-00026
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00033 to out_hw13_2/iter_10/part-00033
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00034 to out_hw13_2/iter_10/part-00034
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/9/part-00035 to out_hw13_2/iter_10/part-00035
13455888|0.00124386335387
1184351|0.000586194420448
4695850|0.000547761527379
5051368|0.000491622780828
1384888|0.000398020431356
6113490|0.000392924911541
2437837|0.000380263755056
7902219|0.000379339642339
6076759|0.000368423441293
13425865|0.000363696668566
```

Running Pagerank on Wikipedia data set for 50 iterations

In [ ]:

```
#!/usr/bin/python
import time

start_time = time.time()

# copying latest script
!scp -i ~/rthallam_sa_east.pem ./pagerank_13_2.py hadoop@ec2-54-233-144-86.sa-ea
st-1.compute.amazonaws.com:/home/hadoop/src
# removing target directory
!aws s3 rm s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_10/ --recurs
ive
# launching script
!ssh -i ~/rthallam_sa_east.pem hadoop@ec2-54-233-144-86.sa-east-1.compute.amazon
aws.com /usr/lib/spark/bin/spark-submit --master yarn-cluster /home/hadoop/src/p
agerank_13_2.py s3n://ucb-mids-mls-networks/wikipedia/all-pages-indexed-out.txt
50 s3n://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_10/ > ./hw_13_2_ite
r10.log 2>&1

end_time = time.time()

print "="*80
print "Time taken to find page rank of the network = {:.2f} seconds".format(end_
time - start_time)
print "="*80
```

Cluster Configuration and Run Time

Cluster Size	9 mx.large (WORKERS) and 1 mx.large (MASTER)
Run time	10hours 10 minutes

In [8]:

```
!rm -f ./out_hw13_2/iter_50/part*
!aws s3 cp s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_10/49/ ./out
_hw13_2/iter_50/ --recursive
!cat ./out_hw13_2/iter_50/part* > ./out_hw13_2/top100_pr_50iter.txt
!head ./out_hw13_2/top100_pr_50iter.txt
```

```
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00001 to out_hw13_2/iter_50/part-00001
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00000 to out_hw13_2/iter_50/part-00000
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/_SUCCESS to out_hw13_2/iter_50/_SUCCESS
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00002 to out_hw13_2/iter_50/part-00002
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00004 to out_hw13_2/iter_50/part-00004
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00006 to out_hw13_2/iter_50/part-00006
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00003 to out_hw13_2/iter_50/part-00003
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00005 to out_hw13_2/iter_50/part-00005
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00007 to out_hw13_2/iter_50/part-00007
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00010 to out_hw13_2/iter_50/part-00010
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00009 to out_hw13_2/iter_50/part-00009
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00011 to out_hw13_2/iter_50/part-00011
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00008 to out_hw13_2/iter_50/part-00008
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00013 to out_hw13_2/iter_50/part-00013
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00014 to out_hw13_2/iter_50/part-00014
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00016 to out_hw13_2/iter_50/part-00016
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00015 to out_hw13_2/iter_50/part-00015
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00021 to out_hw13_2/iter_50/part-00021
```

```
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00019 to out_hw13_2/iter_50/part-00019
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00018 to out_hw13_2/iter_50/part-00018
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00012 to out_hw13_2/iter_50/part-00012
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00022 to out_hw13_2/iter_50/part-00022
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00020 to out_hw13_2/iter_50/part-00020
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00017 to out_hw13_2/iter_50/part-00017
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00023 to out_hw13_2/iter_50/part-00023
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00024 to out_hw13_2/iter_50/part-00024
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00026 to out_hw13_2/iter_50/part-00026
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00029 to out_hw13_2/iter_50/part-00029
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00027 to out_hw13_2/iter_50/part-00027
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00030 to out_hw13_2/iter_50/part-00030
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00032 to out_hw13_2/iter_50/part-00032
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00028 to out_hw13_2/iter_50/part-00028
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00031 to out_hw13_2/iter_50/part-00031
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00025 to out_hw13_2/iter_50/part-00025
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00033 to out_hw13_2/iter_50/part-00033
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00034 to out_hw13_2/iter_50/part-00034
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_1
0/49/part-00035 to out_hw13_2/iter_50/part-00035
13455888|0.00146123288559
1184351|0.000665898159074
4695850|0.000639539383074
5051368|0.000574642371439
1384888|0.000450045113107
2437837|0.000446570248047
6113490|0.000444554733289
7902219|0.000443782019836
13425865|0.000433037750573
6076759|0.000427618817211
```

Results

In [2]:

```
#!/usr/bin/python
from tabulate import tabulate
import sys
import os

LOOKUP = os.path.join('out_hw13_2', 'indices.txt')
TOP10_ITER = os.path.join('out_hw13_2', 'top100_pr_10iter.txt')
TOP50_ITER = os.path.join('out_hw13_2', 'top100_pr_50iter.txt')

lookup = { key.strip():value.strip() for value, key, v1, v2 in (line.split("\t")
for line in open(LOOKUP).read().strip().split('\n')) }
pr_10 = [ (page, float(rank)) for page, rank in (line.split("|") for line in ope
n(TOP10_ITER).read().strip().split('\n')) ]
pr_50 = [ (page, float(rank)) for page, rank in (line.split("|") for line in ope
n(TOP50_ITER).read().strip().split('\n')) ]

pr_10 = sorted(pr_10, key=lambda x: -x[1])
pr_50 = sorted(pr_50, key=lambda x: -x[1])
```

In [5]:

```
print "-"*100
print "Comparing Top 100 pages with {} and {} iterations".format(10, 50)
print "-"*100

results = []
for i in xrange(100):
    results.append([
                    i+1,
                    lookup.get(pr_10[i][0].replace("\"",""), 'NA'),
                    pr_10[i][1],
                    lookup.get(pr_50[i][0].replace("\"",""), 'NA'),
                    pr_50[i][1]
                    ])

print tabulate(results, headers=["#","Page (10)", "Rank (10)", "Page (50)", "Ran
k (50)"])
```

```
--------------------------------------------------------------------------------
--------------------------------
Comparing Top 100 pages with 10 and 50 iterations
--------------------------------------------------------------------------------
--------------------------------
  #  Page (10)                                 Rank (10)  Page (50
)                                 Rank (50)
---  ----------------------------------------  -----------  --------
--------------------------------  -----------
  1  United States                             0.00124386   United S
tates                             0.00146123
  2  Animal                                    0.000586194  Animal
0.000665898
```

3	France		0.000547762	France
0.000639539				
4	Germany		0.000491623	Germany
0.000574642				
5	Arthropod		0.00039802	Arthropo
d		0.000450045		
6	Insect		0.000392925	Canada
0.00044657				
7	Canada		0.000380264	Insect
0.000444555				
8	List of sovereign states		0.00037934	List of
sovereign states		0.000443782		
9	India		0.000368423	United K
ingdom		0.000433038		
10	United Kingdom		0.000363697	India
0.000427619				
11	England		0.000361869	England
0.000423324				
12	Iran		0.000343762	Iran
0.000397745				
13	World War II		0.000324042	World Wa
r II		0.000385394		
14	Poland		0.000309507	Poland
0.000362587				
15	village		0.000300406	village
0.000343523				
16	Countries of the world		0.000294427	Countrie
s of the world		0.000337984		
17	List of countries		0.000285808	Japan
0.000329149				
18	Japan		0.000281883	Italy
0.000328921				
19	Italy		0.00028006	List of
countries		0.000326141		
20	Australia		0.000277023	Australi
a		0.000325039		
21	Lepidoptera		0.000272473	Voivodes
hips of Poland		0.000312619		
22	National Register of Historic Places		0.000271264	National
Register of Historic Places		0.000309512		
23	Voivodeships of Poland		0.000270559	Lepidopt
era		0.000307927		
24	Powiat		0.000262697	Powiat
0.00030306				
25	Gmina		0.000258066	Gmina
0.000297489				
26	London		0.000238864	The New
York Times		0.000285961		
27	The New York Times		0.000235059	London
0.000283553				
28	English language		0.000226878	English
language		0.00026899		
29	China		0.000222865	China

0.000263952					
30	Russia		0.000222504	Russia	0.000260927
31	Departments of France		0.000221376	New York City	0.000257634
32	moth		0.000217131	Departments of France	0.00025492
33	Communes of France		0.000216128	Spain	0.000250967
34	New York City		0.000215487	Communes of France	0.000248627
35	Spain		0.000214133	moth	0.000245322
36	Brazil		0.000210596	Brazil	0.000244669
37	Association football		0.000205145	Association football	0.000238598
38	association football		0.000200794	association football	0.000233255
39	Counties of Iran		0.000188894	California	0.000220583
40	Provinces of Iran		0.000188522	Counties of Iran	0.000214916
41	California		0.000187948	Provinces of Iran	0.000214506
42	Romania		0.000182176	Central European Time	0.000211159
43	Bakhsh		0.000182139	Romania	0.000211144
44	Central European Time		0.000181422	Bakhsh	0.000206994
45	Rural Districts of Iran		0.000178703	Sweden	0.000203257
46	Sweden		0.00017355	Rural Districts of Iran	0.000202494
47	Private Use Areas		0.000170599	Netherlands	0.000196969
48	Netherlands		0.000166894	Private Use Areas	0.000191359
49	Iran Standard Time		0.000164328	World War I	0.000190737
50	Central European Summer Time		0.000161538	New York	0.000188127
51	Mexico		0.000160058	Central European Summer Time	0.000187982
52	World War I		0.000159926	Mexico	0.000187003
53	New York		0.000158606	Iran Standard Time	0.000186699
54	Hangul		0.000158491	AllMusic	0.000185186
55	Iran Daylight Time		0.000157637	Iran Daylight Time	0.000178718

56	AllMusic	0.000156555	Hangul	0.000178283
57	gene	0.000148917	Scotland	0.000173309
58	Scotland	0.00014692	gene	0.000169453
59	Allmusic	0.000143579	Soviet Union	0.00016761
60	Norway	0.000142929	Norway	0.000167178
61	Soviet Union	0.00013989	Allmusic	0.000165367
62	New Zealand	0.00013706	Paris	0.000160658
63	Plant	0.000136014	New Zealand	0.000160488
64	Turkey	0.000135682	Turkey	0.000158972
65	Paris	0.000135243	Plant	0.000157586
66	Geographic Names Information System	0.000133383	Geographic Names Information System	0.000155239
67	Romanize	0.000131861	Switzerland	0.000154895
68	Switzerland	0.000131247	Los Angeles	0.000153252
69	Los Angeles	0.000128151	Romanize	0.000148809
70	United States Census Bureau	0.000124647	United States Census Bureau	0.000147822
71	Angiosperms	0.000124205	Europe	0.000147075
72	Europe	0.000123353	Angiosperms	0.000141818
73	South Africa	0.00012101	South Africa	0.000141269
74	census	0.000118841	census	0.000139036
75	protein	0.000118452	Flowering plant	0.000137617
76	Flowering plant	0.000117578	Austria	0.000136216
77	Austria	0.000115767	protein	0.000134871
78	U.S. state	0.000114222	U.S. state	0.000134712
79	Political divisions of the United States	0.000112744	Argentina	0.000130665
80	Argentina	0.000111865	Political divisions of the United States	0.00013018
81	Chordate	0.000110881	population density	0.000130008
82	population density	0.000110056	Catholic	

Church	0.000128378				
83	Belgium		0.00010736	Chordate	0.000128179
84	BBC		0.000105791	BBC	0.000127291
85	Catholic Church		0.00010564	Belgium	0.000127124
86	Chicago		0.000103702	Chicago	0.000124078
87	Pakistan		0.000103113	Washington, D.C.	0.000120905
88	Washington, D.C.		9.98312e-05	Pakistan	0.000120217
89	genus		9.88173e-05	Finland	0.000115754
90	Finland		9.86714e-05	The Guardian	0.000114478
91	species		9.79067e-05	Latin	0.000114443
92	Eastern European Time		9.73869e-05	Ontario	0.000114276
93	Ontario		9.70987e-05	Czech Republic	0.000113568
94	football (soccer)		9.66824e-05	Philippines	0.00011324
95	Eudicots		9.65718e-05	Denmark	0.00011321
96	Czech Republic		9.64946e-05	Greece	0.000113167
97	Philippines		9.636e-05	genus	0.00011289
98	Greece		9.60419e-05	football (soccer)	0.000112393
99	Denmark		9.59984e-05	Hungary	0.000112162
100	Hungary		9.56538e-05	Eastern European Time	0.000112098

In [7]:

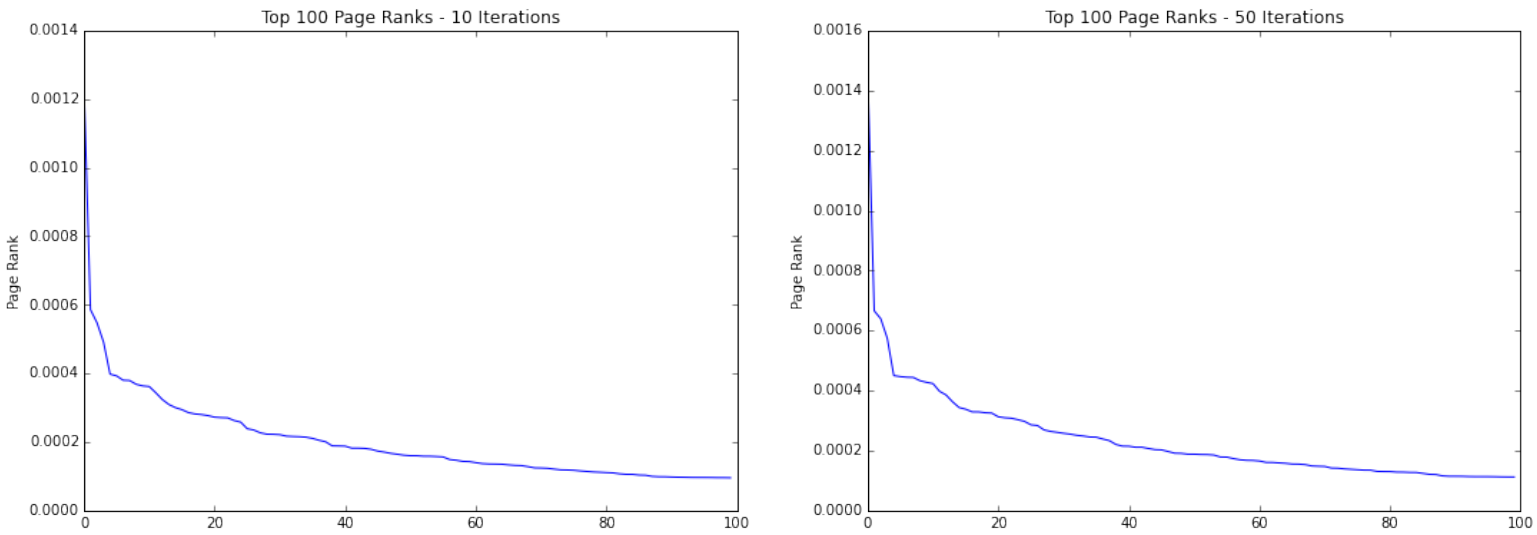
```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

plt.figure(figsize=(18,6))
plt.subplot(121)
plt.title("Top 100 Page Ranks - 10 Iterations")
plt.ylabel('Page Rank')
plt.plot([pr[1] for pr in pr_10])

plt.subplot(122)
plt.title("Top 100 Page Ranks - 50 Iterations")
plt.ylabel('Page Rank')
plt.plot([pr[1] for pr in pr_50])
```

Out[7]:

```
[<matplotlib.lines.Line2D at 0x7f8dc043f810>]
```



Report

- Pages associated with top 100 ranks for both 10 and 50 iterations are almost same though their order is different
- Page rank values itself differ between 10 and 50 iterations and with 50 iterations these tend to be relatively higher

---

HW13.3

Spark GraphX versus your implementation of PageRank

Run the Spark GraphX PageRank implementation on the Wikipedia dataset for 10 iterations, and display the top 100 ranked nodes (with alpha = 0.85).

Run your PageRank implementation on the Wikipedia dataset for 50 iterations, and display the top 100 ranked nodes (with teleportation factor of 0.15). Have the top 100 ranked pages changed? Comment on your findings. Plot both 100 curves.

Report the AWS cluster configuration that you used and how long in minutes and seconds it takes to complete this job.

Put the runtime results of HW13.2 and HW13.3 in a tabular format (with rows corresponding to implemention and columns corresponding to experiment setup (10 iterations, 50 iterations)). Discuss the run times and explaing the differences.

Plot the pagerank values for the top 100 pages resulting from the 50 iterations run (using GraphX). Then plot the pagerank values for the same 100 pages that resulted from the 50 iterations run of your homegrown pagerank implemnentation. Comment on your findings. Have the top 100 ranked pages changed? Have the pagerank values changed? Explain.

TO DO

In [ ]:

```
import org.apache.spark.SparkContext
import org.apache.spark.SparkContext._
import org.apache.spark.SparkConf
import org.apache.spark.graphx._
import org.apache.spark.rdd.RDD

object Pagerank {
        def main(args: Array[String]) {
        val conf = new SparkConf().setAppName("pagerank")
        val sc = new SparkContext(conf)
        val graph = GraphLoader.edgeListFile(sc, "file:/home/hadoop/src/pagerank
/followers")
        // Run PageRank
        //val ranks = graph.pageRank(10).vertices
        val ranks = graph.pageRank(0.0001).vertices

        // Print the result
        println(ranks.collect().mkString("\n"))
    }
}
```

In [ ]:

```
name := "pagerank"
version := "1.0"
scalaVersion := "2.10.5"
libraryDependencies ++= Seq("org.apache.spark" %% "spark-core" % "1.5.2", "org.a
pache.spark" %% "spark-graphx" % "1.5.2")
resolvers += "Akka Repository" at "http://repo.akka.io/releases/"
```

In [ ]:

```
sbt package
```

In [ ]:

```
/usr/lib/spark/bin/spark-submit --class "Pagerank" --master local [4] $(find tar
get -iname "*.jar")
```

HW13.4

Criteo Phase 2 baseline

The Criteo data is located in the following S3 bucket: [criteo-dataset] (https://console.aws.amazon.com/s3/home?region=us-west-1#&bucket=criteo-dataset&prefix=)

Using the training dataset, validation dataset and testing dataset in the Criteo bucket perform the following experiment:

-- write spark code (borrow from Phase 1 of this project) to train a logistic regression model with the following hyperparamters:
-- Number of buckets for hashing: 1,000
-- Logistic Regression: no regularization term
-- Logistic Regression: step size = 10

Report the AWS cluster configuration that you used and how long in minutes and seconds it takes to complete this job.

Report in tabular form the [AUC value](https://en.wikipedia.org/wiki/Receiver_operating_characteristic) for the Training, Validation, and Testing datasets. Report in tabular form the logLossTest for the Training, Validation, and Testing datasets.

Dont forget to put a caption on your tables (above each table).

Baseline Criteo Dataset using Raw Data

In [1]:

```
%%writefile criteo_13_4_1.py
from collections import defaultdict
import hashlib
import sys
from math import log, exp
from pyspark import SparkContext
from pyspark.mllib.linalg import SparseVector
from pyspark.mllib.regression import LabeledPoint
from pyspark.mllib.classification import LogisticRegressionWithSGD
from pyspark.mllib.evaluation import BinaryClassificationMetrics

def hashFunction(numBuckets, rawFeats, printMapping=False):
    """Calculate a feature dictionary for an observation's features based on has
hing.

    Note:
        Use printMapping=True for debug purposes and to better understand how th
e hashing works.

    Args:
        numBuckets (int): Number of buckets to use as features.
        rawFeats (list of (int, str)): A list of features for an observation.  R
epresented as
            (featureID, value) tuples.
        printMapping (bool, optional): If true, the mappings of featureString to
index will be
            printed.

    Returns:
        dict of int to float:  The keys will be integers which represent the buc
kets that the
            features have been hashed to.  The value for a given key will contai
n the count of the
            (featureID, value) tuples that have hashed to that key.
    """
    mapping = {}
    for ind, category in rawFeats:
        featureString = category + str(ind)
        mapping[featureString] = int(int(hashlib.md5(featureString).hexdigest(),
16) % numBuckets)
    if(printMapping): print mapping
    sparseFeatures = defaultdict(float)
    for bucket in mapping.values():
        sparseFeatures[bucket] += 1.0
    return dict(sparseFeatures)

def parseHashPoint(point, numBuckets):
    """Create a LabeledPoint for this observation using hashing.
```

```
    Args:
        point (str): A comma separated string where the first value is the label
and the rest are
            features.
        numBuckets: The number of buckets to hash to.

    Returns:
        LabeledPoint: A LabeledPoint with a label (0.0 or 1.0) and a SparseVecto
r of hashed
            features.
    """
    parsedPoints = parsePoint(point)
    items = point.split(',')
    label = items[0]
    features = hashFunction(numBuckets, parsedPoints, printMapping=False)
    return LabeledPoint(label, SparseVector(numBuckets, features))

def parsePoint(point):
    """Converts a comma separated string into a list of (featureID, value) tuple
s.

    Note:
        featureIDs should start at 0 and increase to the number of features - 1.

    Args:
        point (str): A comma separated string where the first value is the label
and the rest
            are features.

    Returns:
        list: A list of (featureID, value) tuples.
    """
    return [(i, item) for i, item in enumerate(point.split(',')[1:])]

def computeLogLoss(p, y):
    """Calculates the value of log loss for a given probabilty and label.

    Note:
        log(0) is undefined, so when p is 0 we need to add a small value (epsilo
n) to it
        and when p is 1 we need to subtract a small value (epsilon) from it.

    Args:
        p (float): A probabilty between 0 and 1.
        y (int): A label.  Takes on the values 0 and 1.

    Returns:
        float: The log loss value.
    """
    epsilon = 10e-12
    if p == 0:
        p = p + epsilon
    if p == 1:
```

```
        p = p - epsilon

    return -(y * log(p) + (1-y) * log(1-p))

def getP(x, w, intercept):
    """Calculate the probability for an observation given a set of weights and i
ntercept.

    Note:
        We'll bound our raw prediction between 20 and -20 for numerical purposes
.

    Args:
        x (SparseVector): A vector with values of 1.0 for features that exist in
this
            observation and 0.0 otherwise.
        w (DenseVector): A vector of weights (betas) for the model.
        intercept (float): The model's intercept.

    Returns:
        float: A probability between 0 and 1.
    """
    rawPrediction = x.dot(w) + intercept

    # Bound the raw prediction value
    rawPrediction = min(rawPrediction, 20)
    rawPrediction = max(rawPrediction, -20)
    return 1 / (1 + exp(-rawPrediction))

def evaluateResults(model, data):
    """Calculates the log loss for the data given the model.

    Args:
        model (LogisticRegressionModel): A trained logistic regression model.
        data (RDD of LabeledPoint): Labels and features for each observation.

    Returns:
        float: Log loss for the data.
    """
    return data.map(lambda x: computeLogLoss(getP(x.features, model.weights, mod
el.intercept), x.label)).sum() / data.count()

def evaluateMetrics(model, data, label):
    labelsAndScores = data.map(lambda lp:
                               (lp.label, getP(lp.features, model.weights, model.in
tercept)))

    auc = BinaryClassificationMetrics(labelsAndScores).areaUnderROC
    log_loss = evaluateResults(model, data)

    sys.stderr.write('\n LogLoss {0} = {1}'.format(label, log_loss))
    sys.stderr.write('\n AUC {0} = {1}\n'.format(label, auc))

    return (label, log_loss, auc)
```

```
if __name__ == '__main__':
    # Initialize the spark context.
    sc = SparkContext(appName="CriteoBaseline")

    # =========================
    # read raw criteo data set
    # =========================
    rawTrainData = (sc
                .textFile(sys.argv[1], 2)
                .map(lambda x: x.replace('\t', ','))
                .cache() )# work with either ',' or '\t' separated data
    print rawTrainData.take(1)

    rawTestData = (sc
                .textFile(sys.argv[2], 2)
                .map(lambda x: x.replace('\t', ','))
                .cache() )# work with either ',' or '\t' separated data
    print rawTestData.take(1)

    rawValidationData = (sc
                .textFile(sys.argv[3], 2)
                .map(lambda x: x.replace('\t', ','))
                .cache() )# work with either ',' or '\t' separated data
    print rawValidationData.take(1)

    # ====================================================
    # split into train, validation and test data set
    # ====================================================
    #weights = [.8, .1, .1]
    #seed = 42
    # Use randomSplit with weights and seed
    #rawTrainData, rawValidationData, rawTestData = rawData.randomSplit(weights,
seed)
    # Cache the data
    #rawTrainData.cache()
    #rawValidationData.cache()
    #rawTestData.cache()

    nTrain = rawTrainData.count()
    nVal = rawValidationData.count()
    nTest = rawTestData.count()
    print nTrain, nVal, nTest, nTrain + nVal + nTest

    # ====================================================
    # create hash features
    # ====================================================
    numBucketsCTR = 1000    # number of hash buckets

    hashTrainData = rawTrainData.map(lambda x: parseHashPoint(x, numBucketsCTR))
    hashTrainData.cache()
    hashValidationData = rawValidationData.map(lambda x: parseHashPoint(x, numBu
cketsCTR))
```

```
    hashValidationData.cache()

    hashTestData = rawTestData.map(lambda x: parseHashPoint(x, numBucketsCTR))
    hashTestData.cache()

    # ============================================================
    # train logistic regression model
    # ============================================================
    numIters = 100
    stepSize = 10.
    regParam = 0. # no regularization
    regType = 'l2'
    includeIntercept = True

    model = LogisticRegressionWithSGD.train(hashTrainData,
                                            iterations=numIters,
                                            step=stepSize,
                                            regParam=regParam,
                                            regType=regType,
                                            intercept=includeIntercept)
    sortedWeights = sorted(model.weights)

    sys.stderr.write('\n Model Intercept: {0}'.format(model.intercept))
    sys.stderr.write('\n Model Weights (Top 5): {0}\n'.format(sortedWeights[:5]))
)

    l_metrics = []

    l_metrics.append(evaluateMetrics(model, hashTrainData, 'TRAIN'))
    l_metrics.append(evaluateMetrics(model, hashValidationData, 'VALIDATE'))
    l_metrics.append(evaluateMetrics(model, hashTestData, 'TEST'))

    sc.parallelize(l_metrics).saveAsTextFile(sys.argv[4])

    sc.stop()
```

```
Overwriting criteo_13_4_1.py
```

In [ ]:

```
#!/usr/bin/python
import time

start_time = time.time()

# copying latest script
!scp -i ~/rthallam_sa_east.pem ./criteo_13_4_1.py hadoop@ec2-54-233-134-187.sa-e
ast-1.compute.amazonaws.com:/home/hadoop/src
# removing target directory
!aws s3 rm s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/ --recursive
# launching script
!ssh -i ~/rthallam_sa_east.pem hadoop@ec2-54-233-134-187.sa-east-1.compute.amazo
naws.com \
    /usr/lib/spark/bin/spark-submit --master yarn-cluster \
    /home/hadoop/src/criteo_13_4_1.py \
    s3://criteo-dataset/rawdata/train/ \
    s3://criteo-dataset/rawdata/test/ \
    s3://criteo-dataset/rawdata/validation/ \
    s3n://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/
#!ssh -i ~/rthallam_sa_east.pem hadoop@ec2-54-233-144-86.sa-east-1.compute.amazo
naws.com /usr/lib/spark/bin/spark-submit --master yarn-cluster /home/hadoop/src/
pagerank_13_2.py s3n://ucb-mids-mls-rajeshthallam/hw13/PageRank-test_indexed.txt
10 s3n://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_10/


end_time = time.time()

print "="*80
print "Time taken to find baseline metrics of the Criteo data set = {:.2f} secon
ds".format(end_time - start_time)
print "="*80
```

In [2]:

```
#Download results
!rm -fR .out_hw13_4/part*
!aws s3 cp s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/ . --recursive

print "Results (raw)"
!cat part*
```

```
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0000 to ./part-00000
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0005 to ./part-00005
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0004 to ./part-00004
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0001 to ./part-00001
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0008 to ./part-00008
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0003 to ./part-00003
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0002 to ./part-00002
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/_SUCCE
SS to ./_SUCCESS
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0006 to ./part-00006
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0007 to ./part-00007
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0011 to ./part-00011
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0013 to ./part-00013
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0012 to ./part-00012
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0014 to ./part-00014
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0010 to ./part-00010
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0015 to ./part-00015
download: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_4/part-0
0009 to ./part-00009
Results (raw)
('TRAIN', 0.5054639969509631, 0.6914759771327955)
('VALIDATE', 0.5056761120760903, 0.6918797233560421)
('TEST', 0.505602800351624, 0.6920070004287929)
```

Results

Results

Cluster Configuration and Run Time

Cluster Size	3 mx.large (WORKERS) and 1 mx.large (MASTER)
Run time	2hours 46 minutes

Model parameters

Parameter	**Value**
Iterations	100
Regularization	0.0
Regularization Type	L2
Include Intercept	True
Step Size	10

Results: Log loss and AUC

Data set	**Log Loss**	**AUC**
TRAIN	0.5054639969509631	0.6914759771327955
VALIDATION	0.5056761120760903	0.6918797233560421
TEST	0.505602800351624	0.6920070004287929

HW13.5

Criteo Phase 2 Hyperparameter Tuning

Using the training dataset, validation dataset and testing dataset in the Criteo bucket perform the following experiments:

-- write spark code (borrow from Phase 1 of this project) to train a logistic regression model with various hyperparamters. Do a gridsearch of the hyperparameter space and determine optimal settings using the validation set.
-- Number of buckets for hashing: 1,000, 10,000, ... explore different values here
-- Logistic Regression: regularization term: [1e-6, 1e-3] explore other values here also
-- Logistic Regression: step size: explore different step sizes. Focus on a stepsize of 1 initially.

Report the AWS cluster configuration that you used and how long in minutes and seconds it takes to complete this job.

Report in tabular form and using heatmaps the [AUC values](https://en.wikipedia.org/wiki/Receiver_operating_characteristic) for the Training, Validation, and Testing datasets. Report in tabular form and using heatmaps the logLossTest for the Training, Validation, and Testing datasets.

Dont forget to put a caption on your tables (above the table) and on your heatmap figures (put caption below figures) detailing the experiment associated with each table or figure (data, algorithm used, parameters and settings explored.

Discuss the optimal setting to solve this problem in terms of the following:
-- Features
-- Learning algortihm
-- Spark cluster

Justiy your recommendations based on your experimental results and cross reference with table numbers and figure numbers. Also highlight key results with annotations, both textual and line and box based, on your tables and graphs.

Criteo Phase 2 Hyperparameter Tuning

In [1]:

```
%%writefile criteo_13_5_1.py
from collections import defaultdict
import hashlib
import sys
from math import log, exp
from pyspark import SparkContext
from pyspark.mllib.linalg import SparseVector
from pyspark.mllib.regression import LabeledPoint
from pyspark.mllib.classification import LogisticRegressionWithSGD
from pyspark.mllib.evaluation import BinaryClassificationMetrics

def hashFunction(numBuckets, rawFeats, printMapping=False):
```

```
    """Calculate a feature dictionary for an observation's features based on has
hing.

    Note:
        Use printMapping=True for debug purposes and to better understand how th
e hashing works.

    Args:
        numBuckets (int): Number of buckets to use as features.
        rawFeats (list of (int, str)): A list of features for an observation.  R
epresented as
            (featureID, value) tuples.
        printMapping (bool, optional): If true, the mappings of featureString to
index will be
            printed.

    Returns:
        dict of int to float:  The keys will be integers which represent the buc
kets that the
            features have been hashed to.  The value for a given key will contai
n the count of the
            (featureID, value) tuples that have hashed to that key.
    """
    mapping = {}
    for ind, category in rawFeats:
        featureString = category + str(ind)
        mapping[featureString] = int(int(hashlib.md5(featureString).hexdigest(),
16) % numBuckets)
    if(printMapping): print mapping
    sparseFeatures = defaultdict(float)
    for bucket in mapping.values():
        sparseFeatures[bucket] += 1.0
    return dict(sparseFeatures)

def parseHashPoint(point, numBuckets):
    """Create a LabeledPoint for this observation using hashing.

    Args:
        point (str): A comma separated string where the first value is the label
and the rest are
            features.
        numBuckets: The number of buckets to hash to.

    Returns:
        LabeledPoint: A LabeledPoint with a label (0.0 or 1.0) and a SparseVecto
r of hashed
            features.
    """
    parsedPoints = parsePoint(point)
    items = point.split(',')
    label = items[0]
    features = hashFunction(numBuckets, parsedPoints, printMapping=False)
    return LabeledPoint(label, SparseVector(numBuckets, features))
```

```
def parsePoint(point):
    """Converts a comma separated string into a list of (featureID, value) tuple
s.

    Note:
        featureIDs should start at 0 and increase to the number of features - 1.

    Args:
        point (str): A comma separated string where the first value is the label
and the rest
            are features.

    Returns:
        list: A list of (featureID, value) tuples.
    """
    return [(i, item) for i, item in enumerate(point.split(',')[1:])]

def computeLogLoss(p, y):
    """Calculates the value of log loss for a given probabilty and label.

    Note:
        log(0) is undefined, so when p is 0 we need to add a small value (epsilo
n) to it
        and when p is 1 we need to subtract a small value (epsilon) from it.

    Args:
        p (float): A probabilty between 0 and 1.
        y (int): A label.  Takes on the values 0 and 1.

    Returns:
        float: The log loss value.
    """
    epsilon = 10e-12
    if p == 0:
        p = p + epsilon
    if p == 1:
        p = p - epsilon
    return -(y * log(p) + (1-y) * log(1-p))

def getP(x, w, intercept):
    """Calculate the probability for an observation given a set of weights and i
ntercept.

    Note:
        We'll bound our raw prediction between 20 and -20 for numerical purposes
.

    Args:
        x (SparseVector): A vector with values of 1.0 for features that exist in
this
            observation and 0.0 otherwise.
        w (DenseVector): A vector of weights (betas) for the model.
```

```
        intercept (float): The model's intercept.


    Returns:
        float: A probability between 0 and 1.
    """
    rawPrediction = x.dot(w) + intercept

    # Bound the raw prediction value
    rawPrediction = min(rawPrediction, 20)
    rawPrediction = max(rawPrediction, -20)
    return 1 / (1 + exp(-rawPrediction))

def evaluateResults(model, data):
    """Calculates the log loss for the data given the model.

    Args:
        model (LogisticRegressionModel): A trained logistic regression model.
        data (RDD of LabeledPoint): Labels and features for each observation.

    Returns:
        float: Log loss for the data.
    """
    return data.map(lambda x: computeLogLoss(getP(x.features, model.weights, mod
el.intercept), x.label)).sum() / data.count()

def evaluateMetrics(model, data, label):
    labelsAndScores = data.map(lambda lp:
                               (lp.label, getP(lp.features, model.weights, model.in
tercept)))

    auc = BinaryClassificationMetrics(labelsAndScores).areaUnderROC
    log_loss = evaluateResults(model, data)

    sys.stderr.write('\n LogLoss {0} = {1}'.format(label, log_loss))
    sys.stderr.write('\n AUC {0} = {1}\n'.format(label, auc))

    return (label, log_loss, auc)

if __name__ == '__main__':
    # Initialize the spark context.
    sc = SparkContext(appName="CriteoBaseline")

    # ========================
    # read raw criteo data set
    # ========================
    rawTrainData = (sc
                .textFile(sys.argv[1], 2)
                .map(lambda x: x.replace('\t', ','))
                .cache() )# work with either ',' or '\t' separated data
    print rawTrainData.take(1)

    rawTestData = (sc
                .textFile(sys.argv[2], 2)
```

```
                .map(lambda x: x.replace('\t', ','))

                .cache() )# work with either ',' or '\t' separated data
    print rawTestData.take(1)

    rawValidationData = (sc
                .textFile(sys.argv[3], 2)
                .map(lambda x: x.replace('\t', ','))
                .cache() )# work with either ',' or '\t' separated data
    print rawValidationData.take(1)

    # ================================================
    # split into train, validation and test data set
    # ================================================
    #weights = [.8, .1, .1]
    #seed = 42
    # Use randomSplit with weights and seed
    #rawTrainData, rawValidationData, rawTestData = rawData.randomSplit(weights,
seed)
    # Cache the data
    #rawTrainData.cache()
    #rawValidationData.cache()
    #rawTestData.cache()

    nTrain = rawTrainData.count()
    nVal = rawValidationData.count()
    nTest = rawTestData.count()
    print nTrain, nVal, nTest, nTrain + nVal + nTest

    # ================================================
    # create hash features
    # ================================================
    numBucketsCTR = [1000, 10000, 10000]    # number of hash buckets
    iteration = 0

    for numBuckets in numBucketsCTR:
        hashTrainData = rawTrainData.map(lambda x: parseHashPoint(x, numBuckets)
)
        hashTrainData.cache()
        hashValidationData = rawValidationData.map(lambda x: parseHashPoint(x, n
umBuckets))
        hashValidationData.cache()
        hashTestData = rawTestData.map(lambda x: parseHashPoint(x, numBuckets))
        hashTestData.cache()

        # ================================================
        # train logistic regression model
        # ================================================
        numIters = 10
        stepSizes = [1, 10, 100]
        regParams = [1e-6, 1e-3, 1e-1, 0]
        regType = 'l2'
        includeIntercept = True
```

```
        for stepSize in stepSizes:

            for regParam in regParams:
                iteration += 1
                l_metrics = []

                l_metrics.append('Buckets=' + str(numBuckets))
                l_metrics.append('Step Size=' + str(stepSize))
                l_metrics.append('RegParam=' + str(regParam))

                model = LogisticRegressionWithSGD.train(hashTrainData,
                                                        iterations=numIters,
                                                        step=stepSize,
                                                        regParam=regParam,
                                                        regType=regType,

intercept=includeIntercept)
                sortedWeights = sorted(model.weights)


                sys.stderr.write('\n Model Intercept: {0}'.format(model.intercep
t))
                sys.stderr.write('\n Model Weights (Top 5): {0}\n'.format(sorted
Weights[:5]))


                l_metrics.append('Intercept=' + str(model.intercept))
                l_metrics.append('Weights=' + str(sortedWeights[:5]))

                l_metrics.append(evaluateMetrics(model, hashTrainData, 'TRAIN'))
                l_metrics.append(evaluateMetrics(model, hashValidationData, 'VAL
IDATE'))
                l_metrics.append(evaluateMetrics(model, hashTestData, 'TEST'))

                sc.parallelize(l_metrics).saveAsTextFile(sys.argv[4] + '/' + str
(iteration))

    sc.stop()
```

```
Overwriting criteo_13_5_1.py
```

In [ ]:

```
#!/usr/bin/python
import time

start_time = time.time()

# copying latest script
!scp -i ~/rthallam_sa_east.pem ./criteo_13_5_1.py hadoop@ec2-54-233-134-187.sa-e
ast-1.compute.amazonaws.com:/home/hadoop/src
# removing target directory
!aws s3 rm s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/ --recursive
# launching script
!ssh -i ~/rthallam_sa_east.pem hadoop@ec2-54-233-134-187.sa-east-1.compute.amazo
naws.com \
    /usr/lib/spark/bin/spark-submit --master yarn-cluster \
    /home/hadoop/src/criteo_13_5_1.py \
    s3://criteo-dataset/rawdata/train/ \
    s3://criteo-dataset/rawdata/test/ \
    s3://criteo-dataset/rawdata/validation/ \
    s3n://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5
#!ssh -i ~/rthallam_sa_east.pem hadoop@ec2-54-233-144-86.sa-east-1.compute.amazo
naws.com /usr/lib/spark/bin/spark-submit --master yarn-cluster /home/hadoop/src/
pagerank_13_2.py s3n://ucb-mids-mls-rajeshthallam/hw13/PageRank-test_indexed.txt
10 s3n://ucb-mids-mls-rajeshthallam/hw13/results/hw13_2/iter_10/


end_time = time.time()

print "="*80
print "Time taken to find find hypertuning parameters for the Criteo data set =
{:.2f} seconds".format(end_time - start_time)
print "="*80
```

```
criteo_13_5_1.py                                  100% 8985     8.8KB/s
00:00
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/_SUCCESS
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
09
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
10
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
00
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
04
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
02
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
05
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
06
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
08
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
01
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
11
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
03
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
13
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
12
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
14
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
15
delete: s3://ucb-mids-mls-rajeshthallam/hw13/results/hw13_5/part-000
07
15/12/09 21:22:33 WARN util.NativeCodeLoader: Unable to load native-
hadoop library for your platform... using builtin-java classes where
applicable
```

Job is currently running and following scenarios are completed

Each scenario is running for ~50min on 6 mx.large CORE machines

TO DO - Pretty print and heatmap

```
::::::::::::::
1.txt
::::::::::::::
Buckets=1000
Step Size=1
RegParam=1e-06
Intercept=0.767382106444
Weights=[-0.19950713658592067, -0.19770805634256824, -0.19663784695655784,
-0.19489279080272207, -0.15943079452581677]
('TRAIN', 0.5434708131061389, 0.6871363591151867)
('VALIDATE', 0.5437134818109673, 0.6931797822485392)
('TEST', 0.5436171394106204, 0.708423401394534 9)
::::::::::::::
2.txt
::::::::::::::
Buckets=1000
Step Size=1
RegParam=0.001
Intercept=0.763020956329
Weights=[-0.19901607160959192, -0.19722434946331335, -0.19611383648712732,
-0.19436933708059378, -0.15896190333515761]
('TRAIN', 0.543498592171465, 0.6873449287545953)
('VALIDATE', 0.5437409918784116, 0.6923050452782179)
('TEST', 0.5436450978729863, 0.710165929684776)
::::::::::::::
3.txt
::::::::::::::
Buckets=1000
Step Size=1
RegParam=0.1
Intercept=0.416346799533
Weights=[-0.15832838965370033, -0.15704982530780254, -0.15378951908078359,
-0.15168715525251408, -0.12153274083272272]
('TRAIN', 0.5467704328912172, 0.7403338925210812)
('VALIDATE', 0.5469904626994729, 0.6496034652228997)
('TEST', 0.5469313836304891, 0.6875515864350759)
```

** -- END OF ASSIGNMENT 13 -- **