**Exercise commands**

**<DIY1> Connect to HiPerGator, launch standalone Spark cluster**

ssh <YOUR\_ID>@hpg.rc.ufl.edu

cp /ufrc/spark\_workshop/share/\* ~/

ls

sbatch spark-local-cluster.sh

squeue –u <your ID>

**<DIY1> Monitor the Spark cluster**

grep MasterWebUI spark\_cluster.err

module load ubuntu

firefox &

**<DIY3> Spark interactive shells**

**Spark Interactive shells in Scala**

SPARK\_MASTER=$(grep "Starting Spark master" \*.err | cut -d " " -f 9)

module load spark

spark-shell --master $SPARK\_MASTER

**Spark Interactive shells in Python**

SPARK\_MASTER=$(grep "Starting Spark master" \*.err | cut -d " " -f 9)

module load spark

pyspark --master $SPARK\_MASTER

------ type the following python code in -------

from operator import add

from random import random

partitions = 10

n = 100000 \* partitions

def f(\_):

 x = random() \* 2 - 1

 y = random() \* 2 - 1

 return 1 if x \*\* 2 + y \*\* 2 <= 1 else 0

count = sc.parallelize(range(1, n + 1), partitions).map(f).reduce(add)

print("Pi is roughly %f" % (4.0 \* count / n))

**<DIY4> Running python script via pyspark command line**

cat diy4.py

SPARK\_MASTER=$(grep "Starting Spark master" \*.err | cut -d " " -f 9)

PYTHONSTARTUP=mypi.py pyspark --master $SPARK\_MASTER

**<DIY5> Submit Spark job using spark-submit**

SPARK\_MASTER=$(grep "Starting Spark master" \*.err | cut -d " " -f 9)

spark-submit --master $SPARK\_MASTER

$SPARK\_HOME/examples/src/main/python/pi.py 10

spark-submit --master $SPARK\_MASTER

$SPARK\_HOME/examples/src/main/python/pi.py 10 2> /dev/null

**<DIY6> Wordcount example using spark-submit**

SPARK\_MASTER=$(grep "Starting Spark master" \*.err | cut -d " " -f 9)

spark-submit --master $SPARK\_MASTER

$SPARK\_HOME/examples/src/main/python/wordcount.py

spark\_cluster.err > wc.result

cat wc.result