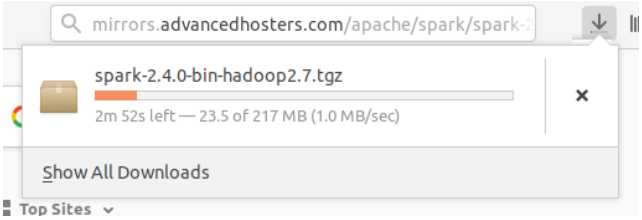
Name: Xinrun Zhang

Date: 11/27/2018

1. Spark Installation

   a. Download Apache Spark



   b. Extract setup file

```
zhangxinrun@ubuntu:~/Downloads$ tar xvfz spark-2.4.0-bin-hadoop2.7.tgz
spark-2.4.0-bin-hadoop2.7/
spark-2.4.0-bin-hadoop2.7/python/
spark-2.4.0-bin-hadoop2.7/python/setup.cfg
```

   c. Move extracted folder

```
zhangxinrun@ubuntu:~/Downloads$ sudo mv spark-2.4.0-bin-hadoop2.7 /usr/local/sp
ark
[sudo] password for zhangxinrun:
```

   d. Change ownership

```
zhangxinrun@ubuntu:~$ sudo chown -R zhangxinrun:zhangxinrun /usr/local/spark
```

   e. Update . bashrc file

```
export SPARK_HOME=/usr/local/spark
export PATH=$PATH:$SPARK_HOME/bin/
```

```
zhangxinrun@ubuntu:~$ gedit ~/.bashrc
zhangxinrun@ubuntu:~$ source .bashrc
```

   f. Copy file and add line

```
zhangxinrun@ubuntu:~$ cd /usr/local/spark/conf
zhangxinrun@ubuntu:/usr/local/spark/conf$ ls
docker.properties.template   slaves.template
fairscheduler.xml.template   spark-defaults.conf.template
log4j.properties.template    spark-env.sh.template
metrics.properties.template
zhangxinrun@ubuntu:/usr/local/spark/conf$ cp spark-env.sh.template spark-env.sh
zhangxinrun@ubuntu:/usr/local/spark/conf$ ls
docker.properties.template   slaves.template
fairscheduler.xml.template   spark-defaults.conf.template
log4j.properties.template    spark-env.sh
metrics.properties.template  spark-env.sh.template
```

```
zhangxinrun@ubuntu:/usr/local/spark/conf$ gedit spark-env.sh
```

```
export SPARK_DIST_CLASSPATH=$(hadoop classpath)
```

   g. Start Hadoop

   At first, when I executed the commands:

```
zhangxinrun@ubuntu:~$ start-all.sh
This script is Deprecated. Instead use start-dfs.sh and start-yarn.sh
Starting namenodes on [localhost]
localhost: starting namenode, logging to /usr/local/hadoop/logs/hadoop-zhangxin
run-namenode-ubuntu.out
localhost: starting datanode, logging to /usr/local/hadoop/logs/hadoop-zhangxin
run-datanode-ubuntu.out
Starting secondary namenodes [0.0.0.0]
0.0.0.0: starting secondarynamenode, logging to /usr/local/hadoop/logs/hadoop-z
hangxinrun-secondarynamenode-ubuntu.out
starting yarn daemons
starting resourcemanager, logging to /usr/local/hadoop/logs/yarn-zhangxinrun-re
sourcemanager-ubuntu.out
localhost: starting nodemanager, logging to /usr/local/hadoop/logs/yarn-zhangxi
nrun-nodemanager-ubuntu.out
zhangxinrun@ubuntu:~$ pyspark
/usr/local/spark/bin/pyspark: line 45: python: command not found
env: ‘python’: No such file or directory
zhangxinrun@ubuntu:~$
```

so, I went to /usr/lib to check out my python version:

```
zhangxinrun@ubuntu:~$ cd /usr/lib
zhangxinrun@ubuntu:/usr/lib$ ls
accountsservice                libqmi
```

in /usr/lib I found:

```
python2.7
python3
python3.6
python3.7
```

so I went to ~/.bashrc and added:

```
export PYTHONPATH=$PYTHONPAH:/usr/lib/python3
export PYSPARK_PYTHON=python3
```

and then, I ran the pyspark command again:

```
zhangxinrun@ubuntu:~$ gedit ~/.bashrc
zhangxinrun@ubuntu:~$ source .bashrc
zhangxinrun@ubuntu:~$ pyspark
/usr/local/spark/bin/pyspark: line 45: python: command not found
Python 3.6.6 (default, Sep 12 2018, 18:26:19)
[GCC 8.0.1 20180414 (experimental) [trunk revision 259383]] on linux
Type "help", "copyright", "credits" or "license" for more information.
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/local/spark/jars/slf4j-log4j12-1.7.16.ja
r!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hadoop/share/hadoop/common/lib/slf
```

```
2018-11-27 20:10:29 WARN  Utils:66 - Your hostname, ubuntu resolves to a loopba
ck address: 127.0.1.1; using 172.16.251.134 instead (on interface ens33)
2018-11-27 20:10:29 WARN  Utils:66 - Set SPARK_LOCAL_IP if you need to bind to
another address
2018-11-27 20:10:30 WARN  NativeCodeLoader:62 - Unable to load native-hadoop li
brary for your platform... using builtin-java classes where applicable
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLev
el(newLevel).
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 2.4.0
      /_/

Using Python version 3.6.6 (default, Sep 12 2018 18:26:19)
SparkSession available as 'spark'.
>>>
```

h. Install numpy and scipy

```
zhangxinrun@ubuntu:~$ sudo apt-get install python-pip
[sudo] password for zhangxinrun:
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

```
zhangxinrun@ubuntu:~$ sudo pip install numpy scipy
The directory '/home/zhangxinrun/.cache/pip/http' or its parent directory is no
t owned by the current user and the cache has been disabled. Please check the p
ermissions and owner of that directory. If executing pip with sudo, you may wan
t sudo's -H flag.
The directory '/home/zhangxinrun/.cache/pip' or its parent directory is not own
ed by the current user and caching wheels has been disabled. check the permissi
ons and owner of that directory. If executing pip with sudo, you may want sudo'
s -H flag.
Collecting numpy
  Downloading https://files.pythonhosted.org/packages/de/37/fe7db552f4507f379d8
1dcb78e58e05030a8941757b1f664517d581b5553/numpy-1.15.4-cp27-cp27mu-manylinux1_x
86_64.whl (13.8MB)
    100% |[redacted]| 13.8MB 76kB/s
Collecting scipy
  Downloading https://files.pythonhosted.org/packages/2a/f3/de9c1bd163119827112
09edaa8c6caa962db30ebb6a8cc6f1dcd2d3ef616/scipy-1.1.0-cp27-cp27mu-manylinux1_x8
6_64.whl (30.8MB)
    100% |[redacted]| 30.8MB 30kB/s
Installing collected packages: numpy, scipy
Successfully installed numpy-1.15.4 scipy-1.1.0
zhangxinrun@ubuntu:~$
```

after this, when I tried to check if I installed these libraries correctly, I put commands in Pyspark. However, I got some errors:

```
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 2.4.0
      /_/

Using Python version 3.6.6 (default, Sep 12 2018 18:26:19)
SparkSession available as 'spark'.
>>> import numpy scipy
  File "<stdin>", line 1
    import numpy scipy
                     ^
SyntaxError: invalid syntax
>>> import numpy
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'numpy'
>>>
```

I went to python3 and typed the command:

```
zhangxinrun@ubuntu:~$ python3
Python 3.6.6 (default, Sep 12 2018, 18:26:19)
[GCC 8.0.1 20180414 (experimental) [trunk revision 259383]] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import numpy
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'numpy'
>>>
```

finally, I went to pip and found that it automatically adds the numpy and scipy into Python2.7:

```
zhangxinrun@ubuntu:~$ pip show numpy
Name: numpy
Version: 1.15.4
Summary: NumPy: array processing for numbers, strings, records, and objects.
Home-page: http://www.numpy.org
Author: Travis E. Oliphant et al.
Author-email: None
License: BSD
Location: /usr/local/lib/python2.7/dist-packages
Requires:
```

considered some answers from:

https://stackoverflow.com/questions/2812520/pip-dealing-with-multiple-python-versions

I decided to installed a new python3-pip:

```
zhangxinrun@ubuntu:~$ sudo apt-get install python3-pip
[sudo] password for zhangxinrun:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
```

and installed the numpy and scipy by typing the command:

```
zhangxinrun@ubuntu:~$ python3 -m pip install numpy
Collecting numpy
  Downloading https://files.pythonhosted.org/packages/ff/7f/9d804d2348471c67a7d
8b5f84f9bc59fd1cefa148986f2b74552f8573555/numpy-1.15.4-cp36-cp36m-manylinux1_x8
6_64.whl (13.9MB)
    100% |[redacted]| 13.9MB 108kB/s
Installing collected packages: numpy
Successfully installed numpy-1.15.4
zhangxinrun@ubuntu:~$ python3 -m pip install scipy
Collecting scipy
  Downloading https://files.pythonhosted.org/packages/a8/0b/f163da98d3a01b3e0ef
1cab8dd2123c34aee2bafbb1c5bffa354cc8a1730/scipy-1.1.0-cp36-cp36m-manylinux1_x86
_64.whl (31.2MB)
    100% |[redacted]| 31.2MB 49kB/s
Collecting numpy>=1.8.2 (from scipy)
  Using cached https://files.pythonhosted.org/packages/ff/7f/9d804d2348471c67a7
d8b5f84f9bc59fd1cefa148986f2b74552f8573555/numpy-1.15.4-cp36-cp36m-manylinux1_x
86_64.whl
Installing collected packages: numpy, scipy
Successfully installed numpy-1.15.4 scipy-1.1.0
```

I tried again to import numpy in pyspark:

```
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 2.4.0
      /_/

Using Python version 3.6.7 (default, Oct 22 2018 11:32:17)
SparkSession available as 'spark'.
>>> import numpy
>>> import scipy
```

now it works correctly, and I deleted pip and libraries in python2.7.

2. Task 2

   a. Create a new directory in Hadoop cluster

```
zhangxinrun@ubuntu:~$ hdfs dfs -mkdir /user/spark/
zhangxinrun@ubuntu:~$ hdfs dfs -mkdir /user/spark/task1/
```

   b. Upload test2.txt to the Hadoop cluster

```
zhangxinrun@ubuntu:~$ cd handson-4/task1
zhangxinrun@ubuntu:~/handson-4/task1$ ls
spark_wc.py  task2.txt
zhangxinrun@ubuntu:~/handson-4/task1$ hdfs dfs -copyFromLocal task2.txt /user/s
park/task1
zhangxinrun@ubuntu:~/handson-4/task1$ hdfs dfs -ls /user/spark/task1
Found 1 items
-rw-r--r--   1 zhangxinrun supergroup       1457 2018-12-07 11:17 /user/spark/t
ask1/task2.txt
```

   c. Edit the xinrun_zhang_task2.py

```
Open ▾   *xinrun_zhang_task2.py   Save
~/handson-4/task1
# Command: spark-submit spark_wc.py
from pyspark import SparkConf, SparkContext
import re

# define a new function
def foo(x):
        x = re.sub(r'[^\w\s]',"",x)
        x = x.lower().split(' ')
        return x


# Spark set-up
conf = SparkConf()
conf.setAppName("Word count App")
sc   = SparkContext(conf=conf)

# uncomment the sc.setLoglevel line, when your program works fine.
# Run the program again to take the screenshot.
#sc.setLogLevel("WARN")

# Upload data file in Hadoop and provide its path in textFile function
rdd = sc.textFile("/user/spark/task2/task2.txt")
rdd = rdd.flatMap(lambda x: x.split(' '))
rdd = rdd.flatMap(foo)
rdd = rdd.map(lambda x: (x, 1))
# Add few lines of code below
rdd  = rdd.reduceByKey(lambda x,y: x+y)
# Add few lines of code below
out = rdd.sortBy(lambda x: x[-1],False)
# out = rdd.sortBy(lambda x: x[1])
out = out.take(10)
# you may store top 10 results in out variable
# and use it to display as mentioned below.
for item in out:
        print(item[0],':\t',str(item[1]))
```

   d. Execute in spark

```
2018-12-07 21:15:11 INFO  DAGScheduler:54 - Job 0 finished: runJob at PythonRDD
.scala:153, took 1.950168 s
the :     23
of :      17
and :     11
is :      8
science :         6
a :       5
in :      5
be :      5
to :      5
it :      4
```

3. Task 3

a. Create a new directory in Hadoop cluster

```
zhangxinrun@ubuntu:~$ hdfs dfs -mkdir /user/spark/task3/
```

b. Upload task3.txt to the cluster

```
zhangxinrun@ubuntu:~$ cd handson-4/task3
zhangxinrun@ubuntu:~/handson-4/task3$ ls
task3.txt  xinrun_zhang_task3.py
zhangxinrun@ubuntu:~/handson-4/task3$ hdfs dfs -copyFromLocal task3.txt /user/s
park/task3/
```

c. Edit xinrun_zhang_task3.py

```
# cast attributes type if needed.
rdd = rdd.map(lambda x: x.split('\t'))
rdd = rdd.map(lambda x: Row(city_name = x[2], profit = float(x[4])))

sqlContext = SQLContext(sc)
# Add code to convert RDD to dataframe
df = sqlContext.createDataFrame(rdd)

# create SQL table from data frame.
df.registerTempTable('ds_table')

# Write query using sqlContext.sql() function
result = sqlContext.sql("SELECT city_name, ROUND(AVG(profit), 2) AS avg_profit,
ROUND(STDDEV_POP(profit), 3) AS stddev_profit FROM ds_table GROUP BY city_name")

# You may convert SQL dataframe in RDD
out = result.rdd.map(lambda x:x.city_name + '\t' + str(x.avg_profit) + '\t' +
str(x.stddev_profit))
# and use it for pretty formatting as mentioned below
# city\t(average sale with 2 digits after decimal)\t(standard deviation in sale
with 3 digits after decimal)
# For example:
# Las Vegas     1200.56 23.321
out = out.collect()
```

d. Execute in spark

```
angxinrun/handson-4/task3/xinrun_zhang_task3.py:42, took 18.814556 s
North Las Vegas 263.3   153.357
Phoenix 254.71  142.503
Omaha   274.8   144.09
Anchorage       242.33  137.122
Anaheim 267.54  141.513
Greensboro      282.5   144.626
Dallas  270.74  150.367
Oakland 276.15  134.079
Laredo  249.54  138.548
Scottsdale      274.61  147.412
San Antonio     272.73  132.919
Bakersfield     253.53  161.236
Raleigh 298.71  141.993
Chula Vista     216.2   154.272
Philadelphia    262.54  129.635
Louisville      223.46  133.304
Los Angeles     247.55  152.484
Chandler        239.07  138.451
Sacramento      260.18  161.181
Indianapolis    256.46  141.145
```

4. Task 4
   a. Create a Hadoop cluster

```
zhangxinrun@ubuntu:~$ hdfs dfs -mkdir /user/spark/task4/
```

   b. Upload task4.txt to the cluster

```
zhangxinrun@ubuntu:~/handson-4/task4$ ls
spark_ml_reg.py  task4.txt
zhangxinrun@ubuntu:~/handson-4/task4$ hdfs dfs -copyFromLocal task4.txt /user/s
park/task4/
```

   c. Execute xinrun_zhang_task4.py

```
2018-12-08 01:14:43 WARN  BLAS:61 - Failed to load implementation from: com.git
hub.fommil.netlib.NativeRefBLAS
[(125.68454547221556, 117.195), (126.17388995752985, 118.129), (125.61080091910
561, 118.595), (125.87891174509008, 125.472), (126.52457000540423, 127.696)]
RMSE: 2.210146792566532
R2: 0.8713910826291172
```

   d.

5. Task 5
   a. Make a new Hadoop cluster

```
zhangxinrun@ubuntu:~$ hdfs dfs -mkdir /user/spark/task5
```

   b. Upload task5.txt to the cluster

```
zhangxinrun@ubuntu:~/handson-4/task5$ hdfs dfs -copyFromLocal task5.txt /user/s
park/task5
```

   c. Execute xinrun_zhang_task5.py

```
hub.fommil.netlib.NativeRefBLAS
+----------+-----+
|prediction|label|
+----------+-----+
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       2.0|    2|
|       2.0|    2|
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       3.0|    2|
|       3.0|    2|
|       1.0|    1|
|       2.0|    2|
|       2.0|    2|
|       1.0|    1|
|       3.0|    3|
+----------+-----+
only showing top 20 rows
```

```
None
Accuracy: 0.94
Precision: 0.951
Recall: 0.942
F-measure: 0.942
```

d.

6. Task 6
   a. Modify the xinrun_zhang_task6.py
   b. Execute xinrun_zhang_task6.py

```
2018-12-08 12:42:19 INFO  ContextHandler:781 - Started o.s.j.s.ServletContextHa
ndler@7e5f3345{/metrics/json,null,AVAILABLE,@Spark}
+----------+-----+
|prediction|label|
+----------+-----+
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       2.0|    2|
|       2.0|    2|
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       1.0|    1|
|       2.0|    2|
|       2.0|    2|
|       1.0|    1|
|       2.0|    2|
|       2.0|    2|
|       1.0|    1|
|       3.0|    3|
+----------+-----+
only showing top 20 rows

None
Error:0.04347826086956519
```

   c. Comparison with task5

      From the result, we can see the accuracy of random forest is much better than the logistic regression. However, we still can’ t say the random forest is better than logistic regression, because maybe just that random forest is more fitting to this dataset than logistic regression.