

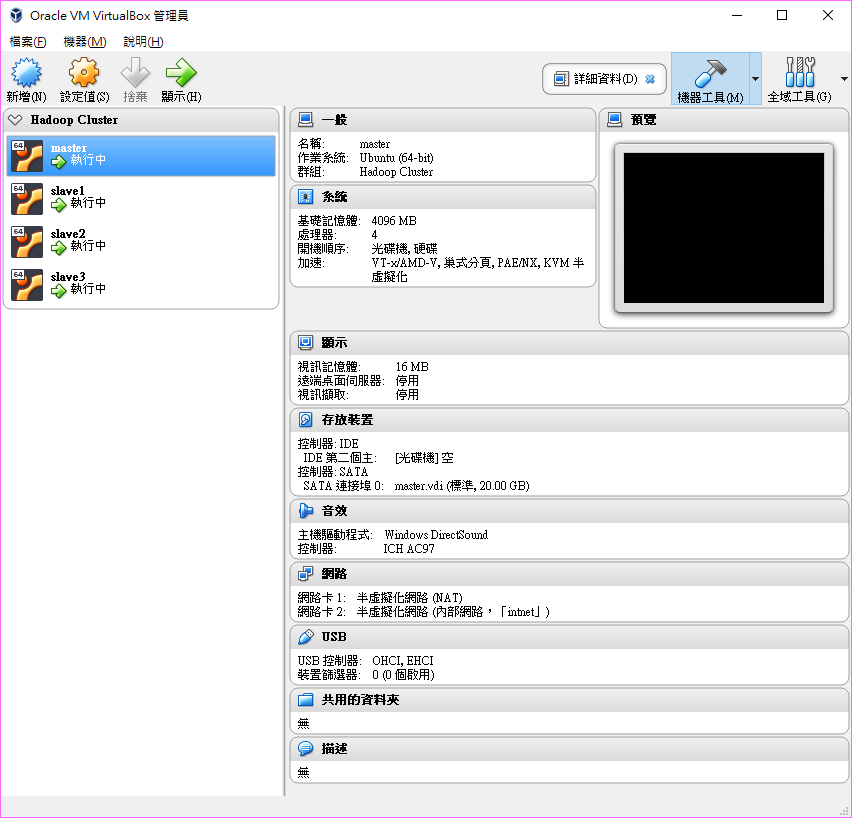
Big Data Mining and Applications, Spring 2018

HW1

萬俊瑋 | 106598018 | 2018/3/21

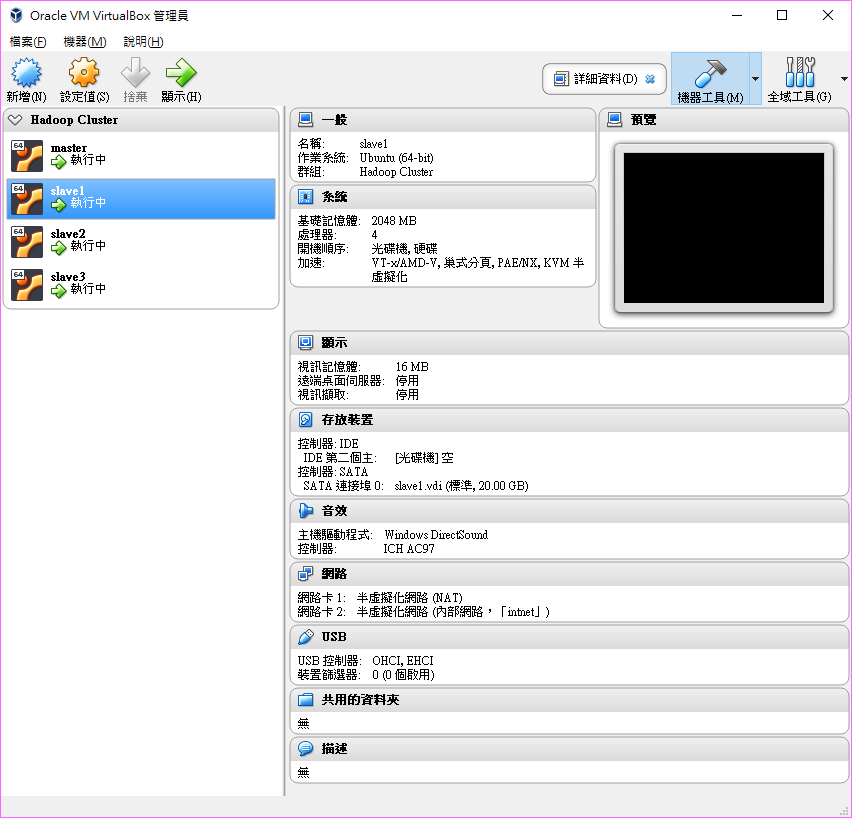
# 主機配置

都用VirtualBox裝在同一台主機，I7-7700 CPU、16G RAM。

Master 一台

Slave三台

除了記憶體改2G，其他都一樣。



# 環境設定

四台都安裝Ubuntu 16.04.4 LTS Server版，安裝完後執行以下指令。

更新系統  
sudo apt update  
sudo apt upgrade

安裝Java  
sudo add-apt-repository ppa:webupd8team/java  
sudo apt-get update  
sudo apt-get install oracle-java8-installer

安裝Hadoop  
wget http://ftp.twaren.net/Unix/Web/apache/hadoop/common/hadoop-2.7.5/hadoop-2.7.5.tar.gz  
sudo tar -zxvf hadoop-2.7.5.tar.gz  
sudo mv hadoop-2.7.5 /usr/local/hadoop  
ll /usr/local/Hadoop

安裝Scala  
wget http://www.scala-lang.org/files/archive/scala-2.11.12.tgz  
tar xvf scala-2.11.12.tgz  
sudo mv scala-2.11.12 /usr/local/scala

安裝Spark  
wget http://ftp.twaren.net/Unix/Web/apache/spark/spark-2.3.0/spark-2.3.0-bin-hadoop2.7.tgz  
tar zxf spark-2.3.0-bin-hadoop2.7.tgz  
sudo mv spark-2.3.0-bin-hadoop2.7.tgz /usr/local/spark/

安裝Anaconda  
wget https://repo.continuum.io/archive/Anaconda3-5.1.0-Linux-x86\_64.sh  
sh Anaconda3-5.1.0-Linux-x86\_64.sh –b

環境變數  
nano ~/.bashrc  
貼下以下內容  
# Anaconda3  
export PATH="/home/hadoop/anaconda3/bin:$PATH"  
export ANACONDA\_PATH=/home/hadoop/anaconda3  
  
# Java  
export JAVA\_HOME=/usr/lib/jvm/java-8-oracle  
export PATH=${JAVA\_HOME}/bin:${PATH}  
export HADOOP\_CLASSPATH=${JAVA\_HOME}/lib/tools.jar  
  
# Scala  
export SCALA\_HOME=/usr/local/scala  
export PATH=$PATH:$SCALA\_HOME/bin  
  
# Hadoop  
export HADOOP\_HOME=/usr/local/hadoop  
export PATH=$PATH:$HADOOP\_HOME/bin  
export HADOOP\_MAPRED\_HOME=$HADOOP\_HOME  
export HADOOP\_COMMON\_HOME=$HADOOP\_HOME  
export HADOOP\_HDFS\_HOME=$HADOOP\_HOME  
export YARN\_HOME=$HADOOP\_HOME  
export HADOOP\_COMMON\_LIB\_NATIVE\_DIR=$HADOOP\_HOME/lib/native  
export HADOOP\_OPTS="-Djava.library.path=$HADOOP\_HOME/lib"  
export JAVA\_LIBRARY\_PATH=$HADOOP\_HOME/lib/native:$JAVA\_LIBRARY\_PATH  
  
# Spark  
export SPARK\_HOME=/usr/local/spark  
export PATH=$PATH:$SPARK\_HOME/bin  
export PYSPARK\_DRIVER\_PYTHON=$ANACONDA\_PATH/bin/ipython  
export PYSPARK\_PYTHON=$ANACONDA\_PATH/bin/python  
export HADOOP\_CONF\_DIR=/usr/local/hadoop/etc/Hadoop

修改Hadoop組態設定檔  
sudo nano /usr/local/hadoop/etc/hadoop/hadoop-env.sh  
輸入以下內容  
export JAVA\_HOME=/usr/lib/jvm/java-8-oracle  
  
sudo nano /usr/local/hadoop/etc/hadoop/core-site.xml  
在<configuration></configuration>之間，輸入下列內容  
<property>  
 <name>fs.default.name</name>  
 <value>hdfs://master:9000</value>  
</property>  
  
sudo nano /usr/local/hadoop/etc/hadoop/yarn-site.xml  
在<configuration></configuration>之間，輸入下列內容  
<property>  
 <name>yarn.nodemanager.aux-services</name>  
 <value>mapreduce\_shuffle</value>  
</property>  
<property>  
 <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>  
 <value>org.apache.hadoop.mapred.ShuffleHandler</value>  
</property>  
<property>  
 <name>yarn.resourcemanager.resource-tracker.address</name>  
 <value>master:8025</value>  
</property>  
<property>  
 <name>yarn.resourcemanager.scheduler.address</name>  
 <value>master:8030</value>  
</property>  
<property>  
 <name>yarn.resourcemanager.address</name>  
 <value>master:8050</value>  
</property>  
  
sudo cp /usr/local/hadoop/etc/hadoop/mapred-site.xml.template /usr/local/hadoop/etc/hadoop/mapred-site.xml  
sudo nano /usr/local/hadoop/etc/hadoop/mapred-site.xml  
在<configuration></configuration>之間，輸入下列內容  
<property>  
 <name>mapreduce.framework.name</name>  
 <value>yarn</value>  
</property>  
<property>  
 <name>mapred.job.tracker</name>  
 <value>master:54311</value>  
</property>

sudo nano /usr/local/hadoop/etc/hadoop/master  
輸入下列內容  
master

sudo nano /usr/local/hadoop/etc/hadoop/slaves  
輸入下列內容  
slave1  
slave2  
slave3

複製master主機為三台分別為slave1、slave2、slave3  
設定四台的內部網路IP分別為100、101、102、103  
sudo nano /etc/network/interfaces  
輸入以下內容  
# The loopback network interface  
auto lo  
iface lo inet loopback  
# NAT interface  
auto enp0s3  
iface enp0s3 inet dhcp  
# Inter interface  
auto enp0s8  
iface enp0s8 inet static  
address 192.168.56.100  
netmask 255.255.255.0  
network 192.168.56.0  
broadcast 192.168.56.255

設定hostname  
sudo nano /etc/hostname  
分別為master、slave1、slave2、slave3

設定hosts檔案  
sudo nano /etc/hosts  
輸入以下內容  
127.0.0.1 localhost  
192.168.56.100 master  
192.168.56.101 slave1  
192.168.56.102 slave2  
192.168.56.103 slave3

master主機  
sudo nano /usr/local/hadoop/etc/hadoop/hdfs-site.xml  
在<configuration></configuration>之間，輸入下列內容  
<property>  
 <name>dfs.replication</name>  
 <value>3</value>  
</property>  
<property>  
 <name>dfs.namenode.name.dir</name>  
 <value> file:/usr/local/hadoop/hadoop\_data/hdfs/namenode</value>  
</property>  
  
slave主機  
sudo nano /usr/local/hadoop/etc/hadoop/hdfs-site.xml  
在<configuration></configuration>之間，輸入下列內容  
<property>  
 <name>dfs.replication</name>  
 <value>3</value>  
</property>  
<property>  
 <name>dfs.datanode.data.dir</name>  
 <value> file:/usr/local/hadoop/hadoop\_data/hdfs/datanode</value>  
 </property>

設定ssh無密碼登入在master做  
mkdir -p ~/.ssh  
chmod 700 ~/.ssh  
ssh-keygen  
在產生金鑰的過程中，會詢問一些問題，對於一般的使用者而言，全部都使用預設值（直接按下 Enter 鍵）即可。  
Generating public/private rsa key pair.  
Enter file in which to save the key (/home/seal/.ssh/id\_rsa):  
首先指定金鑰儲存的位置，使用預設值即可，直接按下 Enter 鍵。  
Enter passphrase (empty for no passphrase):  
指定金鑰保護密碼，如果有設定密碼的話，以後每次使用都要輸入密碼，除你需要非常高的安全性，否則就不用設定了，直接按下 Enter 鍵即可。  
Enter same passphrase again:  
再次輸入密碼，直接按下 Enter 鍵，接著就會產生金鑰了。  
Your identification has been saved in /home/seal/.ssh/id\_rsa.  
Your public key has been saved in /home/seal/.ssh/id\_rsa.pub.  
The key fingerprint is…..  
複製金鑰到三台slave主機上  
ssh-copy-id USER@HOST

建立與格式化HDFS 目錄  
slave1、2、3  
ssh slave1  
sudo rm -rf /usr/local/hadoop/hadoop\_data/hdfs  
sudo mkdir -p /usr/local/hadoop/hadoop\_data/hdfs/datanode  
sudo chown hduser:hduser -R /usr/local/Hadoop  
exit

master  
sudo rm -rf /usr/local/hadoop/hadoop\_data/hdfs  
mkdir -p /usr/local/hadoop/hadoop\_data/hdfs/namenode  
sudo chown -R hduser:hduser /usr/local/hadoop  
hadoop namenode –format

啟動Hadoop  
/usr/local/hadoop/sbin/start-all.sh

查看目前所執行的行程   
jps

Hadoop Resource­Manager Web介面網址  
http://master:8088/

開啟HDFS Web UI網址  
http://master:50070/

啟動Spark  
/usr/local/spark/sbin/start-all.sh

開啟Spark UI網址  
http://master:8080/

設定外部訪問jupyter notebook  
jupyter notebook --generate-config

修改默認配置文件  
nano ~/.jupyter/jupyter\_notebook\_config.py  
c.NotebookApp.ip='\*' # 就是設置所有ip皆可訪問

啟動jupyter notebook  
PYSPARK\_DRIVER\_PYTHON=jupyter PYSPARK\_DRIVER\_PYTHON\_OPTS="notebook" pyspark --master spark://master:7077

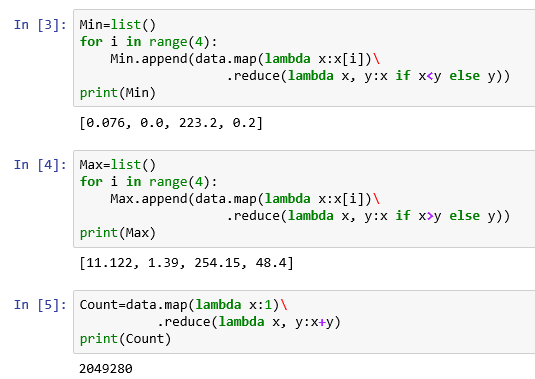
開啟jupyter notebook網址  
http://master:8888/

# 原始碼



# 答案

## (30pt) (1) Output the minimum, maximum, and count of the columns: ‘global active power’, ‘global reactive power’, ‘voltage’, and ‘global intensity’



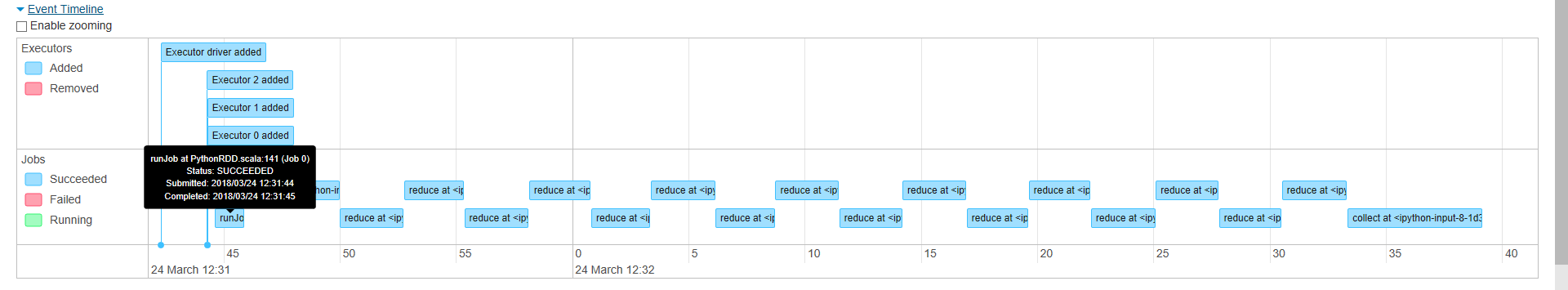
## (30pt) (2) Output the mean and standard deviation of these columns

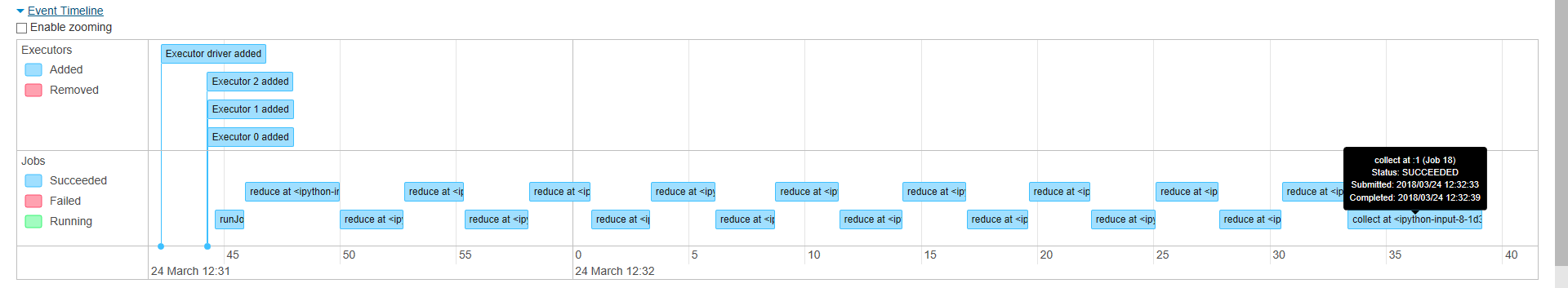


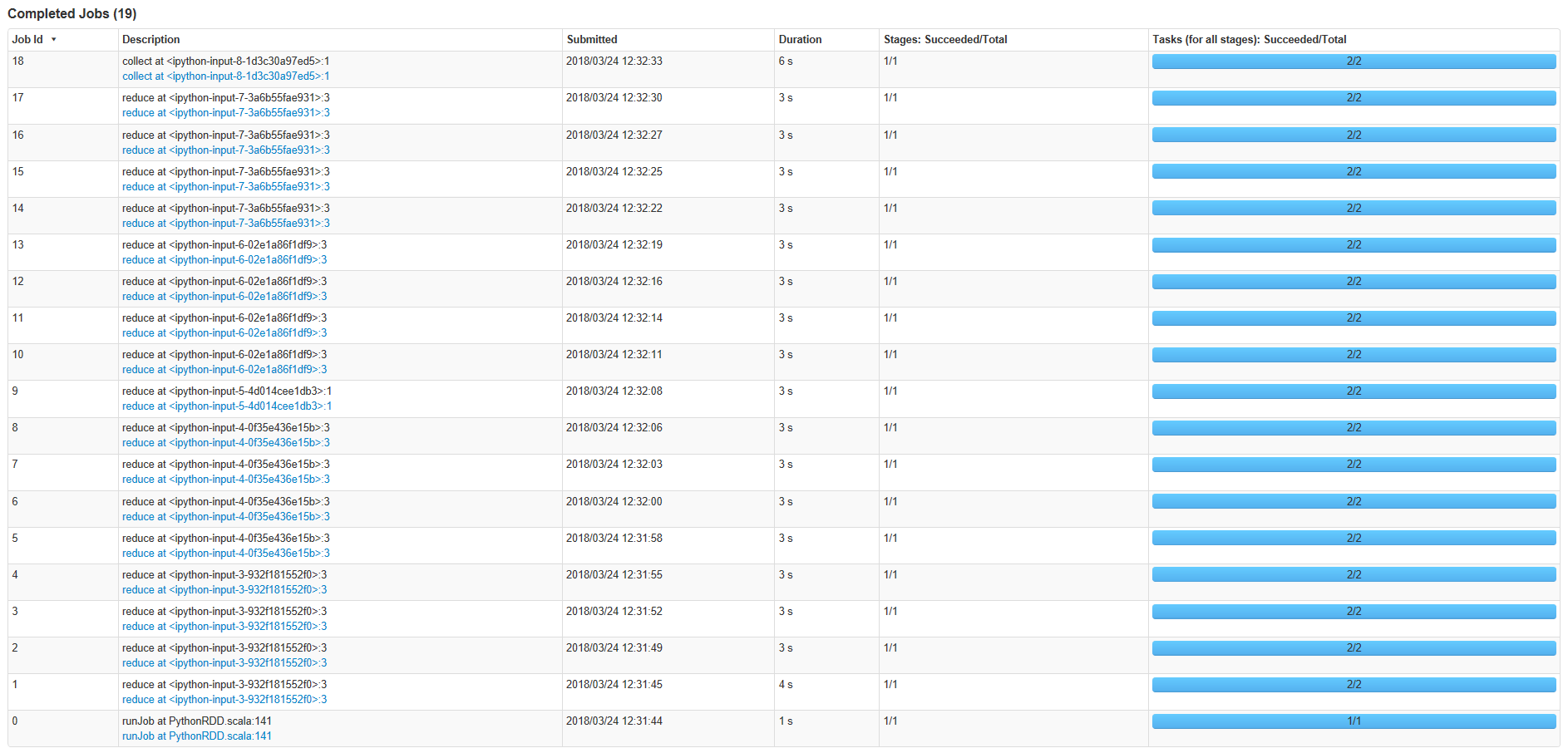
## (40pt) (3) Perform min-max normalization on the columns to generate normalized output

<https://drive.google.com/file/d/1NSMJSyiVYG_tte4tob8ceHgJyfqYbybG/view?usp=sharing>

## 執行時間







總共59s