databricksProject_01

DSCI 417 –Project 01

Analysis of NASA Server Logs

Sean Graham

Part A: Set up Environment

This part of the project will set up the enviornment.

The following cell imports the necessary packages and tools.

```
import matplotlib.pyplot as plt
from pyspark.sql import SparkSession
```

The following cell creates the SparkSession and SparkContext objects.

```
spark = SparkSession.builder.getOrCreate()
sc = spark.sparkContext
```

Part B: Load and Process Data

This part of the project will load and process data from a file containing one month of server log data collected from NASA.gov in August 1995.

The following cell will load the contents of the server log file into an RDD named nasa_raw and print the number of elements in this RDD.

```
nasa_raw = sc.textFile('/FileStore/tables/NASA_server_logs_Aug_1995.txt')
print(nasa_raw.count())
```

```
1569888
```

The following cell will display the first few elements of the RDD.

```
for line in nasa_raw.take(10):
    print(line)
```

```
in24.inetnebr.com [01/Aug/1995:00:00:01] "GET /shuttle/missions/sts-68/news/sts-68-mcc-05.txt" 200 1839
uplherc.upl.com [01/Aug/1995:00:00:07] "GET /" 304 0
uplherc.upl.com [01/Aug/1995:00:00:08] "GET /images/ksclogo-medium.gif" 304 0
uplherc.upl.com [01/Aug/1995:00:00:08] "GET /images/MOSAIC-logosmall.gif" 304 0
uplherc.upl.com [01/Aug/1995:00:00:08] "GET /images/USA-logosmall.gif" 304 0
ix-esc-ca2-07.ix.netcom.com [01/Aug/1995:00:00:09] "GET /images/launch-logo.gif" 200 1713
uplherc.upl.com [01/Aug/1995:00:00:10] "GET /images/WORLD-logosmall.gif" 304 0
slppp6.intermind.net [01/Aug/1995:00:00:10] "GET /history/skylab/skylab.html" 200 1687
piweba4y.prodigy.com [01/Aug/1995:00:00:10] "GET /images/launchmedium.gif" 200 11853
slppp6.intermind.net [01/Aug/1995:00:00:11] "GET /history/skylab/skylab-small.gif" 200 9202
```

The following cell will process each line of server information by removing the double quotes, tokenizing the strings, replacing the hyphens that appear for the number of bytes with zeros, and coercing the values into appropriate datatypes.

```
def process_row(row):
    row = row.replace('"', '')
    tokens = row.split(' ')
    if tokens[-1] == '-':
        tokens[-1] = tokens[-1].replace('-', '0')
    tokens[5] = int(tokens[5])
    return tokens

nasa = nasa_raw.map(process_row)

nasa.persist()

for row in nasa.take(10):
    print(row)
```

```
['in24.inetnebr.com', '[01/Aug/1995:00:00:01]', 'GET', '/shuttle/missions/sts-68/news/sts-68-mcc-05.txt', '200', 1839]
['uplherc.upl.com', '[01/Aug/1995:00:00:07]', 'GET', '/', '304', 0]
['uplherc.upl.com', '[01/Aug/1995:00:00:08]', 'GET', '/images/ksclogo-medium.gif', '304', 0]
['uplherc.upl.com', '[01/Aug/1995:00:00:08]', 'GET', '/images/MOSAIC-logosmall.gif', '304', 0]
['uplherc.upl.com', '[01/Aug/1995:00:00:08]', 'GET', '/images/USA-logosmall.gif', '304', 0]
['ix-esc-ca2-07.ix.netcom.com', '[01/Aug/1995:00:00:09]', 'GET', '/images/launch-logo.gif', '200', 1713]
['uplherc.upl.com', '[01/Aug/1995:00:00:10]', 'GET', '/images/WORLD-logosmall.gif', '304', 0]
['slppp6.intermind.net', '[01/Aug/1995:00:00:10]', 'GET', '/history/skylab/skylab.html', '200', 1687]
['piweba4y.prodigy.com', '[01/Aug/1995:00:00:10]', 'GET', '/images/launchmedium.gif', '200', 11853]
['slppp6.intermind.net', '[01/Aug/1995:00:00:11]', 'GET', '/history/skylab/skylab-small.gif', '200', 9202]
```

Part C: Most Requested Resources

This part of the project will determine which resources were requested the most frequently.

The following cell will determine which resources were requested the most frequently.

```
count_by_resource = nasa.map(lambda x : (x[3], 1)).reduceByKey(lambda x, y : x + y).sortBy(lambda x : x[1], ascending=False)

for row in count_by_resource.take(10):
    print(row)
```

```
('/images/NASA-logosmall.gif', 97410)
('/images/KSC-logosmall.gif', 75337)
('/images/MOSAIC-logosmall.gif', 67448)
('/images/USA-logosmall.gif', 67068)
('/images/WORLD-logosmall.gif', 66444)
('/images/ksclogo-medium.gif', 62778)
('/ksc.html', 43688)
('/history/apollo/images/apollo-logo1.gif', 37826)
('/images/launch-logo.gif', 35138)
('/', 30347)
```

Part D: Most Common Request Origins

This part of the project will determine which servers are the origins for the greatest number of requests.

The following cell will determine which servers are the origins for the greatest number of requests.

```
count_by_origin = nasa.map(lambda x : (x[0], 1)).reduceByKey(lambda x, y : x + y).sortBy(lambda x : x[1], ascending=False)

for row in count_by_origin.take(10):
    print(row)
```

```
('edams.ksc.nasa.gov', 6530)
('piweba4y.prodigy.com', 4846)
('163.206.89.4', 4791)
('piweba5y.prodigy.com', 4607)
('piweba3y.prodigy.com', 4416)
('www-d1.proxy.aol.com', 3889)
('www-b2.proxy.aol.com', 3534)
('www-b3.proxy.aol.com', 3463)
('www-c5.proxy.aol.com', 3423)
('www-b5.proxy.aol.com', 3411)
```

Part E: Request Types

This part of the project will analyze records based on their request type. It will start by confirming that there are three different request types.

The following cell will analyze records based on their request type. It will start by confirming that there are three different request types.

```
req_types = nasa.map(lambda x : x[2]).distinct().collect()
print(req_types)
```

```
['GET', 'HEAD', 'POST']
```

The following cell will count the number of requests of each type.

```
for type in req_types:
    number = nasa.filter(lambda x : type in x).count()
    print('There were', number, type, 'requests.')
```

```
There were 1565812 GET requests.
There were 3965 HEAD requests.
There were 111 POST requests.
```

The following cell will determine the average number of bytes returned to the client for each request type.

```
avg_bytes = nasa.map(lambda x : (x[2], (x[5], 1))).reduceByKey(lambda x, y : (x[0] + y[0], x[1] + y[1])).map(lambda x: (x[0], round(x[1][0]/x[1][1]))).collect()
```

```
for element in avg_bytes:
    print(element)
```

```
('GET', 17134)
('HEAD', 0)
('POST', 495)
```

Part F: Status Codes

This part of the project will analyze the status codes returned by the server.

This following cell will analyze the status codes returned by the server

```
status_codes = nasa.map(lambda x : x[4]).distinct().sortBy(lambda x : x[0], ascending=True).collect()
print(status_codes)
```

```
['200', '302', '304', '404', '403', '501', '500']
```

The following cell will determine which status codes appear for each request type.

```
for type in req_types:
    list = nasa.filter(lambda x : type in x).map(lambda x : x[4]).distinct().sortBy(lambda x : x[0], ascending=True).collect()
    print('Status codes for', type, 'requests:', list)
```

```
Status codes for GET requests: ['200', '304', '302', '404', '403', '500']
Status codes for HEAD requests: ['200', '302', '404']
Status codes for POST requests: ['200', '404', '501']
```

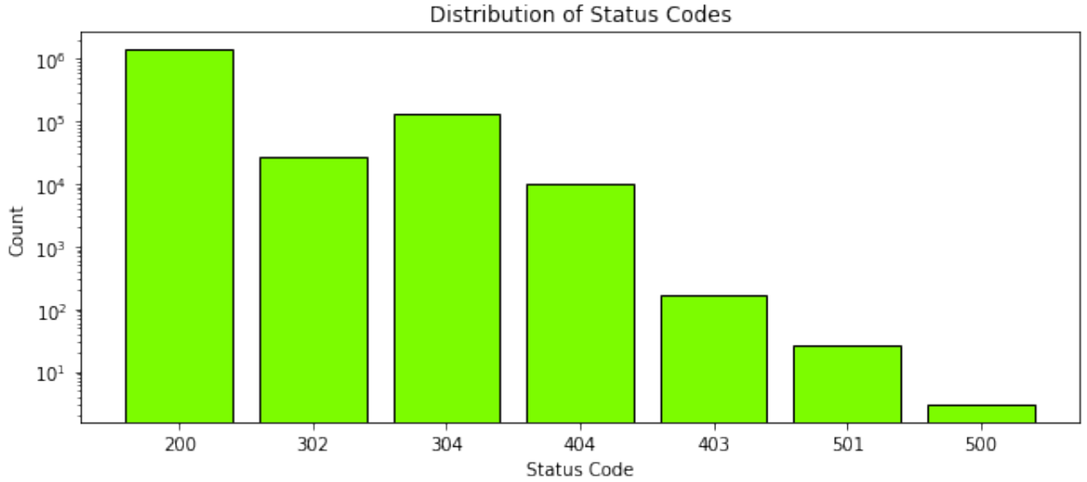
The following cell will count the number of requests resulting in each status code.

```
code_counts = []

for code in status_codes:
    number = nasa.filter(lambda x : code in x).count()
    code_counts.append(number)

plt.figure(figsize=[10,4])
plt.bar(status_codes, code_counts, color='lawngreen', edgecolor='k')
plt.title('Distribution of Status Codes')
plt.xlabel('Status Code')
plt.ylabel('Count')
plt.yscale('log')
plt.show
```

Out[13]:




```
<function matplotlib.pyplot.show(*args, **kw)>
```

Part G: Request Volume by Day

This part of the project will determine the number of requests received by the server during each day in August 1995.

The following cell will determine the number of requests received by the server during each day in August 1995.

```
counts_by_day = nasa.map(lambda x : (x[1][1:3], 1)).reduceByKey(lambda x, y : x + y).sortBy(lambda x : x[0], ascending=True)

for row in counts_by_day.take(5):
    print(row)
```

```
('01', 33996)
('03', 41388)
('04', 59557)
```

```
('05', 31893)
('06', 32420)
```

The following cell will create a bar chart to display the distribution of requests by the day of the month.