Big Data Concepts Project Report

10/21/2021

-Submitted by **Uday Kiran Sharaff**

for INFO I 535

Contents:

Introduction:

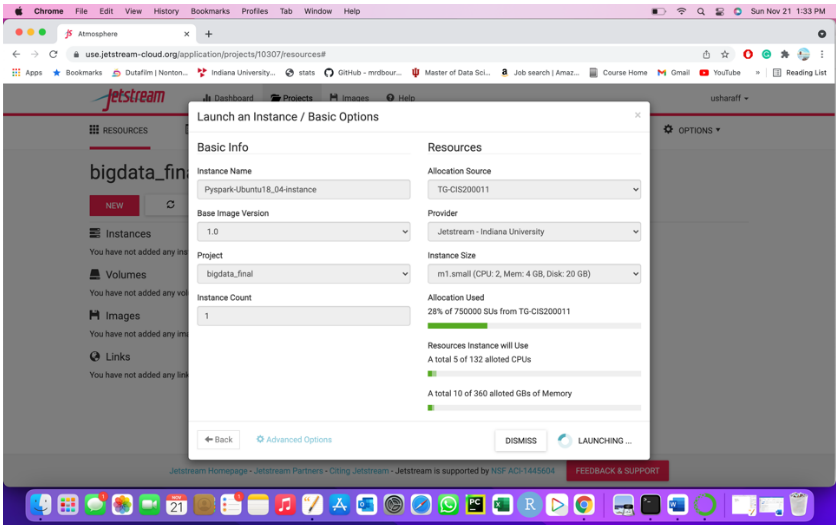
This project analyses on Real time Data published by Capital Bike share. Capital bike share is a bike sharing company mainly operating in Washington DC and some surrounding counties. The objective of this project is to see how the usage of this service changed from start of this year to now. For this, data from Jan 2021 and Oct 2021 is used to compare and analyze. The data is huge with 100070 rows in Jan 2021 and 328300 rows in Oct 2021. The comparisons are done on the number of users, type of users, type of bikes used and most frequent start and end destination pairs.

Background:

The way of life has significantly changed all over the world in this year and most of it is still recovering from the global pandemic. When I was going through the recommended datasets, I came across this data set and wanted to check how this company is affected by the pandemic and how the users changed over this year. So, I chose this dataset, I took data from Jan 2021 which is start of this year when the pandemic was still at its peak and Oct 2021 as it is the most recent data published and would be an apt representation of changed situation after all the vaccinations and reduced cases.

Methodology:

I used a virtual machine on Jetstream by IU for this project. First I have created a new project and added an instance **Pyspark-Ubuntu18.04-instance** with pre-installed python3, scala, spark3, Hadoop for processing.



Launch an Instance / Basic Options

Basic Info

Instance Name: Pyspark-Ubuntu18_04-instance

Base Image Version: 1.0

Project: bigdata_final

Instance Count: 1

Resources

Allocation Source: TG-CIS200011

Provider: Jetstream - Indiana University

Instance Size: m1.small (CPU: 2, Mem: 4 GB, Disk: 20 GB)

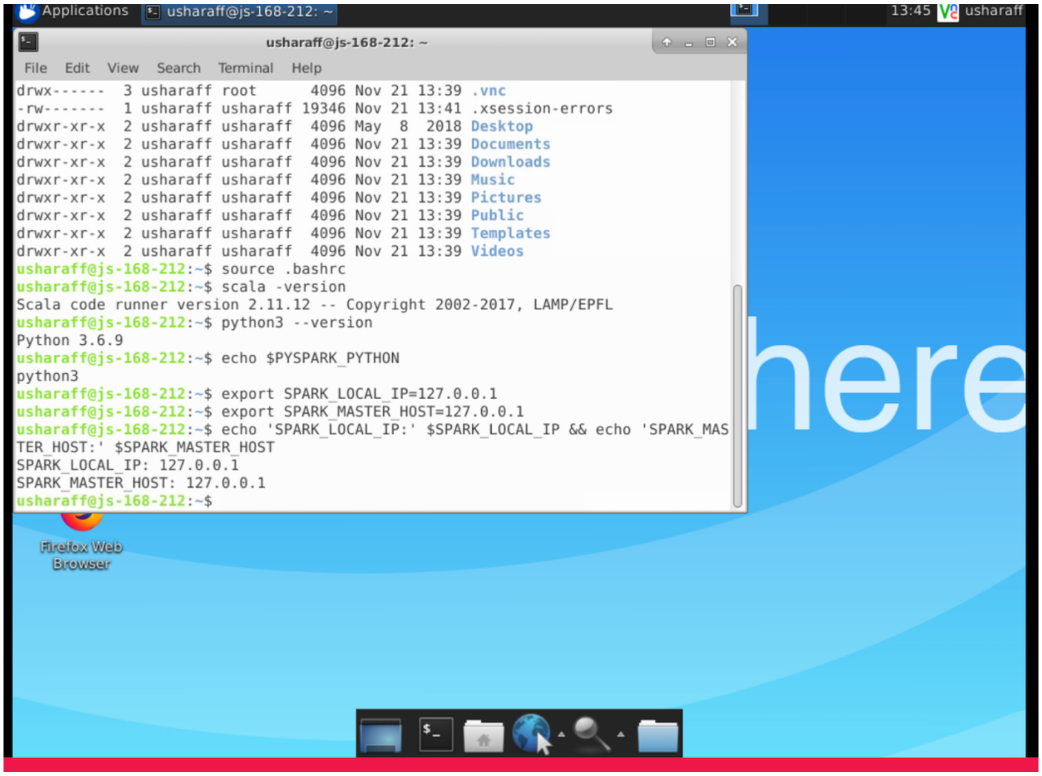
Allocation Used: 28% of 750000 SUs from TG-CIS200011

Resources Instance will Use: A total 5 of 132 alloted CPUs

A total 10 of 360 alloted GBs of Memory

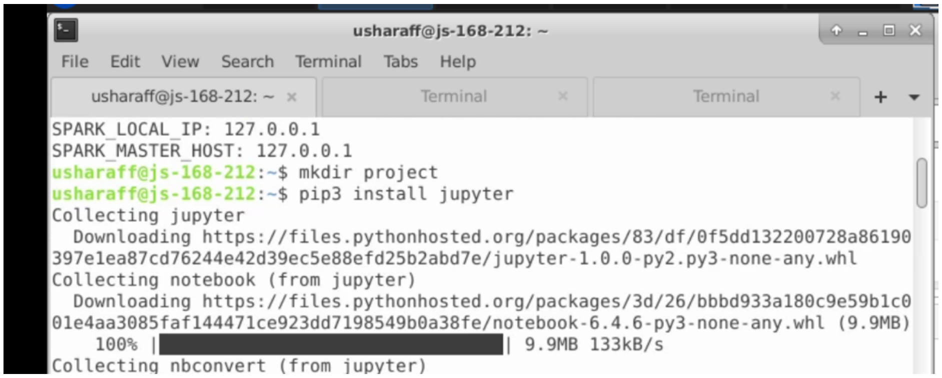
Back
Advanced Options
DISMISS
LAUNCHING ...
bigdata_final
RESOURCES
OPTIONS
NEW
Instances
Volumes
Images
Links
Dashboard
Projects
Help
usharaff
use.jetstream-cloud.org/application/projects/10307/resources#
Jetstream Homepage - Jetstream Partners - Citing Jetstream - Jetstream is supported by NSF ACI-1445604
FEEDBACK & SUPPORT

Once the VM is deployed I used the web desktop. I verified if all the required software like python and pyspark are installed. Exported additional environment variables for Spark – local IP and master.



```
drwx------  3 usharaff root      4096 Nov 21 13:39 .vnc
-rw-------  1 usharaff usharaff 19346 Nov 21 13:41 .xsession-errors
drwxr-xr-x  2 usharaff usharaff  4096 May  8  2018 Desktop
drwxr-xr-x  2 usharaff usharaff  4096 Nov 21 13:39 Documents
drwxr-xr-x  2 usharaff usharaff  4096 Nov 21 13:39 Downloads
drwxr-xr-x  2 usharaff usharaff  4096 Nov 21 13:39 Music
drwxr-xr-x  2 usharaff usharaff  4096 Nov 21 13:39 Pictures
drwxr-xr-x  2 usharaff usharaff  4096 Nov 21 13:39 Public
drwxr-xr-x  2 usharaff usharaff  4096 Nov 21 13:39 Templates
drwxr-xr-x  2 usharaff usharaff  4096 Nov 21 13:39 Videos
usharaff@js-168-212:~$ source .bashrc
usharaff@js-168-212:~$ scala -version
Scala code runner version 2.11.12 -- Copyright 2002-2017, LAMP/EPFL
usharaff@js-168-212:~$ python3 --version
Python 3.6.9
usharaff@js-168-212:~$ echo $PYSPARK_PYTHON
python3
usharaff@js-168-212:~$ export SPARK_LOCAL_IP=127.0.0.1
usharaff@js-168-212:~$ export SPARK_MASTER_HOST=127.0.0.1
usharaff@js-168-212:~$ echo 'SPARK_LOCAL_IP:' $SPARK_LOCAL_IP && echo 'SPARK_MAS
TER_HOST:' $SPARK_MASTER_HOST
SPARK_LOCAL_IP: 127.0.0.1
SPARK_MASTER_HOST: 127.0.0.1
usharaff@js-168-212:~$
```

Then I downloaded CSV files from https://s3.amazonaws.com/capitalbikeshare-data/index.html onto the web desktop. Created a directory for Project. I installed jupyter notebook into the web desktop. Opened a new ipynb file in Jupyter notebook to start processing.

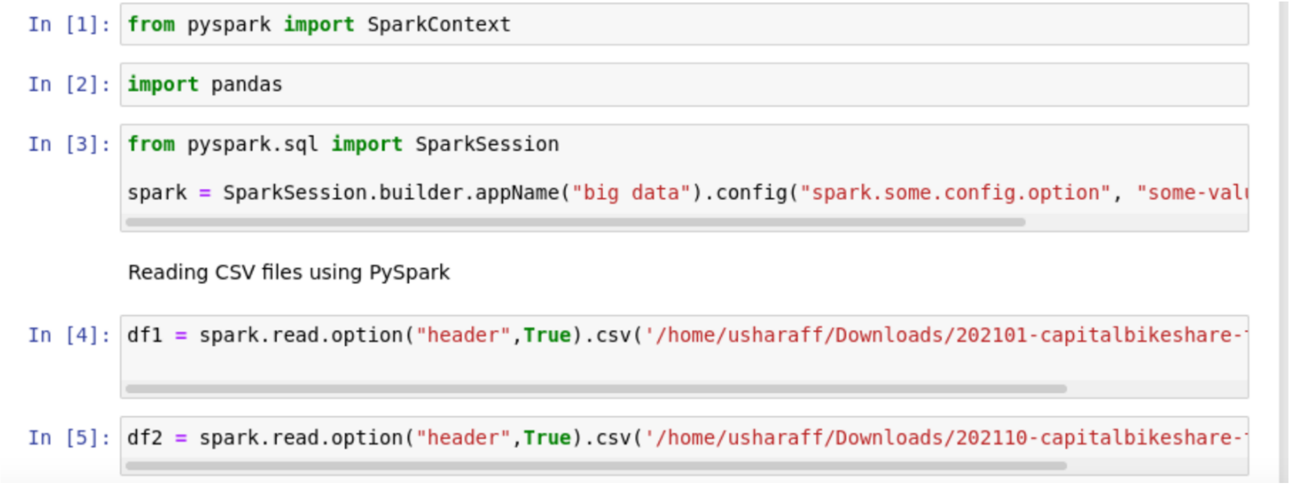


```
SPARK_LOCAL_IP: 127.0.0.1
SPARK_MASTER_HOST: 127.0.0.1
usharaff@js-168-212:~$ mkdir project
usharaff@js-168-212:~$ pip3 install jupyter
Collecting jupyter
  Downloading https://files.pythonhosted.org/packages/83/df/0f5dd132200728a86190
397e1ea87cd76244e42d39ec5e88efd25b2abd7e/jupyter-1.0.0-py2.py3-none-any.whl
Collecting notebook (from jupyter)
  Downloading https://files.pythonhosted.org/packages/3d/26/bbbd933a180c9e59b1c0
01e4aa3085faf144471ce923dd7198549b0a38fe/notebook-6.4.6-py3-none-any.whl (9.9MB)
    100% |████████████████████████████████| 9.9MB 133kB/s
Collecting nbconvert (from jupyter)
```

I used two CSV files, the data on users of capital bike share in Jan 2021 and Oct 2021, start of year and most recent data.

I used pyspark to load, transform and query the data as the data is huge. Pyspark will be faster for dealing with Big data sets because it uses in-memory columnar storage and does batch processing.

I imported necessary libraries in the jupyter notebook, loaded the two csv files into 2 data frames using pyspark.

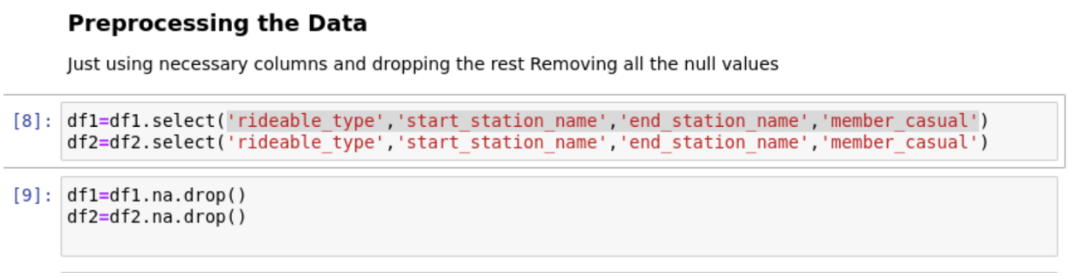


```
In [1]: from pyspark import SparkContext

In [2]: import pandas

In [3]: from pyspark.sql import SparkSession

        spark = SparkSession.builder.appName("big data").config("spark.some.config.option", "some-valu
```

Reading CSV files using PySpark

```
In [4]: df1 = spark.read.option("header",True).csv('/home/usharaff/Downloads/202101-capitalbikeshare-

In [5]: df2 = spark.read.option("header",True).csv('/home/usharaff/Downloads/202110-capitalbikeshare-
```

Then for preprocessing the data I just took the necessary columns 'rideable_type', 'start_station_name', 'end_station_name', 'member_casual'.
I dropped the rows with null/None values.



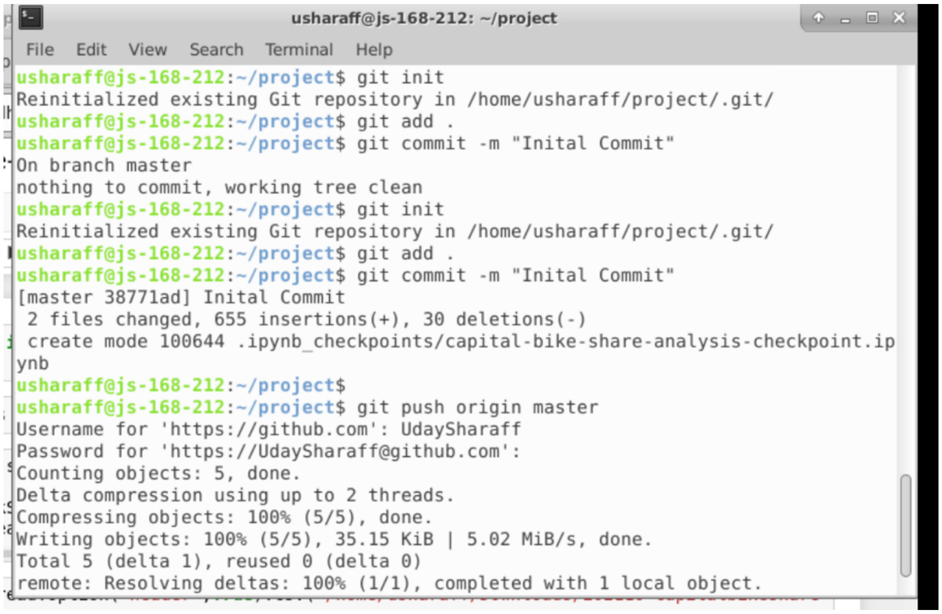
Preprocessing the Data

Just using necessary columns and dropping the rest Removing all the null values

```
[8]: df1=df1.select('rideable_type','start_station_name','end_station_name','member_casual')
     df2=df2.select('rideable_type','start_station_name','end_station_name','member_casual')
```

```
[9]: df1=df1.na.drop()
     df2=df2.na.drop()
```

I linked my Github account to webdesktop and uploaded and updated the code into the repository regularly for backup and tracking.

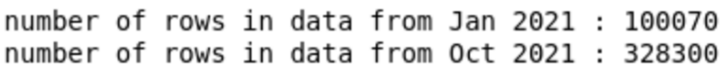


usharaff@js-168-212: ~/project
File Edit View Search Terminal Help

```
usharaff@js-168-212:~/project$ git init
Reinitialized existing Git repository in /home/usharaff/project/.git/
usharaff@js-168-212:~/project$ git add .
usharaff@js-168-212:~/project$ git commit -m "Inital Commit"
On branch master
nothing to commit, working tree clean
usharaff@js-168-212:~/project$ git init
Reinitialized existing Git repository in /home/usharaff/project/.git/
usharaff@js-168-212:~/project$ git add .
usharaff@js-168-212:~/project$ git commit -m "Inital Commit"
[master 38771ad] Inital Commit
 2 files changed, 655 insertions(+), 30 deletions(-)
 create mode 100644 .ipynb_checkpoints/capital-bike-share-analysis-checkpoint.ip
ynb
usharaff@js-168-212:~/project$
usharaff@js-168-212:~/project$ git push origin master
Username for 'https://github.com': UdaySharaff
Password for 'https://UdaySharaff@github.com':
Counting objects: 5, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 35.15 KiB | 5.02 MiB/s, done.
Total 5 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
```

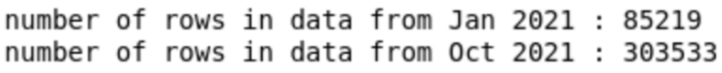
Results and Discussion:

Before preprocessing:



```
number of rows in data from Jan 2021 : 100070
number of rows in data from Oct 2021 : 328300
```

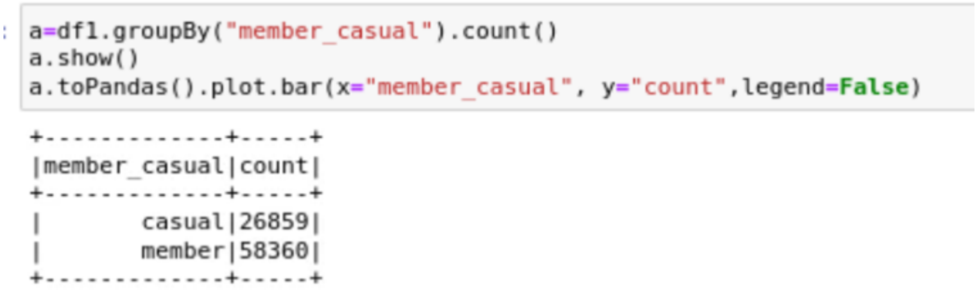
After preprocessing:



```
number of rows in data from Jan 2021 : 85219
number of rows in data from Oct 2021 : 303533
```

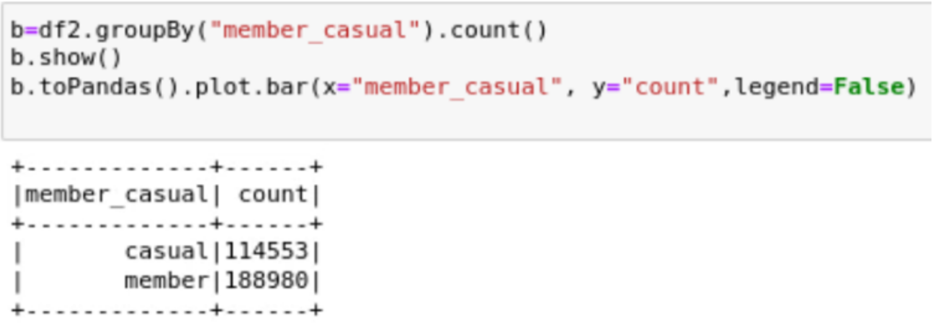
After removing the Nones the January data has 85219 rows and October data has 303533 rows, showing the users significantly increased from January almost 3.5 times.

Then I checked for distribution of members vs casual users in both January and October. For this I used groupby and count offered by pyspark.sql. And plotted the plots using pandas.plot .



```
a=df1.groupBy("member_casual").count()
a.show()
a.toPandas().plot.bar(x="member_casual", y="count",legend=False)
```

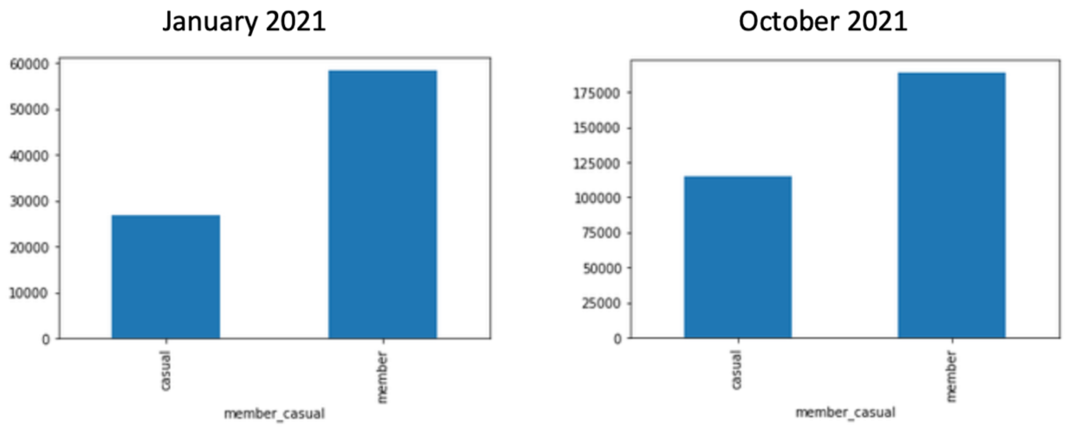
```
+-------------+-----+
|member_casual|count|
+-------------+-----+
|       casual|26859|
|       member|58360|
+-------------+-----+
```



```
b=df2.groupBy("member_casual").count()
b.show()
b.toPandas().plot.bar(x="member_casual", y="count",legend=False)
```

```
+-------------+------+
|member_casual| count|
+-------------+------+
|       casual|114553|
|       member|188980|
+-------------+------+
```

Bar Plots

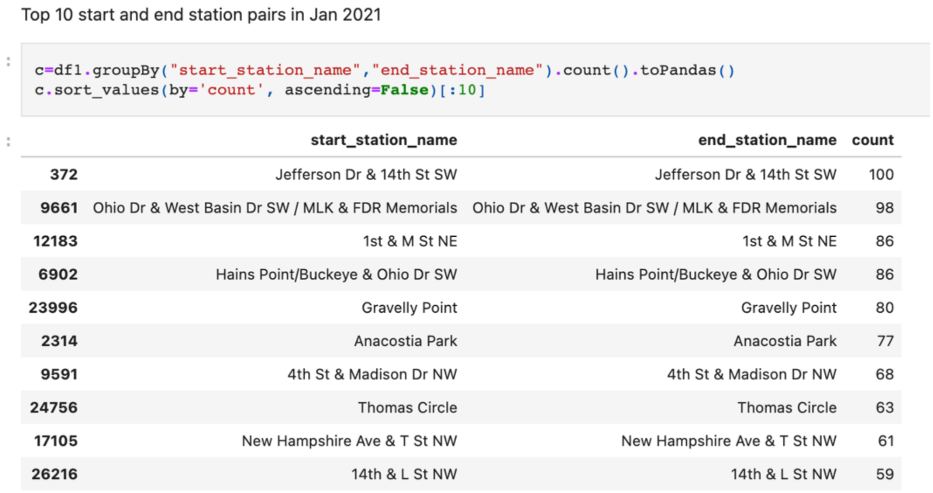


January 2021

October 2021

Even though both users increased over the year, the ratio of casual users to members increased from January to October, looks like many casual users are trying to use this service.

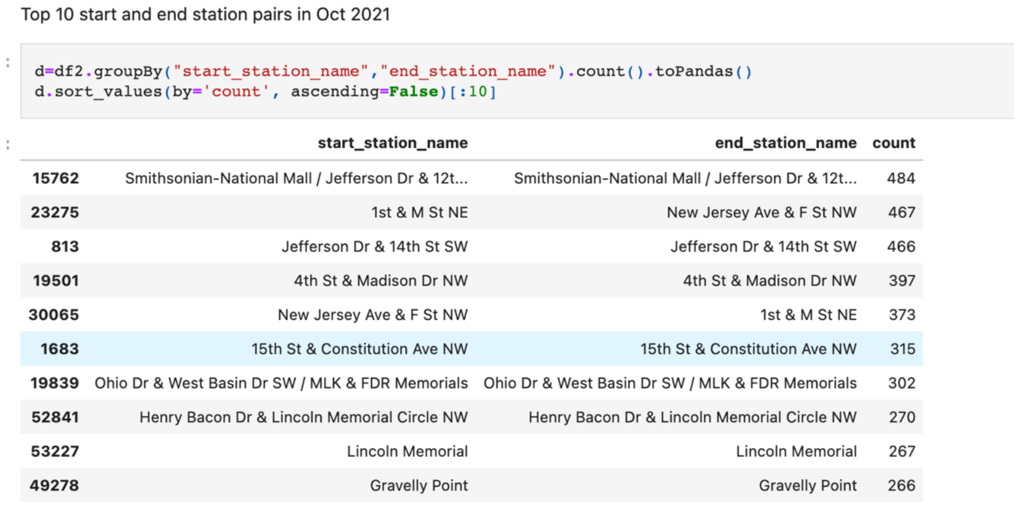
I looked at the most frequent start and end destination pairs to see how the daily commute looks like. For this I grouped the data in start and end destination pairs using group by on both the columns and took the top 5 counts after sorting the data frame on count.



Top 10 start and end station pairs in Jan 2021

```
c=df1.groupBy("start_station_name","end_station_name").count().toPandas()
c.sort_values(by='count', ascending=False)[:10]
```

	start_station_name	end_station_name	count
372	Jefferson Dr & 14th St SW	Jefferson Dr & 14th St SW	100
9661	Ohio Dr & West Basin Dr SW / MLK & FDR Memorials	Ohio Dr & West Basin Dr SW / MLK & FDR Memorials	98
12183	1st & M St NE	1st & M St NE	86
6902	Hains Point/Buckeye & Ohio Dr SW	Hains Point/Buckeye & Ohio Dr SW	86
23996	Gravelly Point	Gravelly Point	80
2314	Anacostia Park	Anacostia Park	77
9591	4th St & Madison Dr NW	4th St & Madison Dr NW	68
24756	Thomas Circle	Thomas Circle	63
17105	New Hampshire Ave & T St NW	New Hampshire Ave & T St NW	61
26216	14th & L St NW	14th & L St NW	59



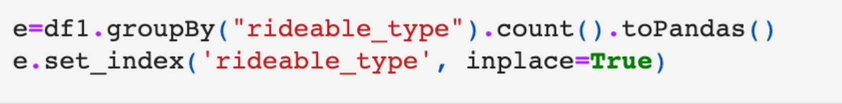
Top 10 start and end station pairs in Oct 2021

```
d=df2.groupBy("start_station_name","end_station_name").count().toPandas()
d.sort_values(by='count', ascending=False)[:10]
```

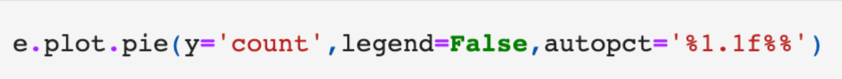
	start_station_name	end_station_name	count
15762	Smithsonian-National Mall / Jefferson Dr & 12t...	Smithsonian-National Mall / Jefferson Dr & 12t...	484
23275	1st & M St NE	New Jersey Ave & F St NW	467
813	Jefferson Dr & 14th St SW	Jefferson Dr & 14th St SW	466
19501	4th St & Madison Dr NW	4th St & Madison Dr NW	397
30065	New Jersey Ave & F St NW	1st & M St NE	373
1683	15th St & Constitution Ave NW	15th St & Constitution Ave NW	315
19839	Ohio Dr & West Basin Dr SW / MLK & FDR Memorials	Ohio Dr & West Basin Dr SW / MLK & FDR Memorials	302
52841	Henry Bacon Dr & Lincoln Memorial Circle NW	Henry Bacon Dr & Lincoln Memorial Circle NW	270
53227	Lincoln Memorial	Lincoln Memorial	267
49278	Gravelly Point	Gravelly Point	266

All the top 10 rows in Jan have same start and end stations while there are some with different start and end in Oct. In Jan, commuters would have taken round trips from the station nearest to their home as many places are closed then, which is not the case in October. The popular tourist attractions like Smithsonian museums and Lincoln memorial are not in top 10 trips in Jan and appeared in the top in Oct. Indicating tourism is rising in the capital and many people are going outdoors.

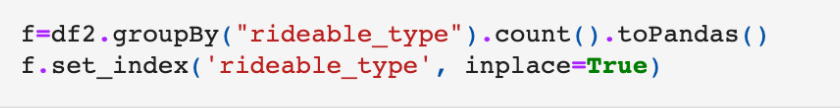
Finally I wanted to look at the change in the types of bikes commuters used and how it changed from Jan. For this I grouped by bite type column and calculated the count to plot a pie chart.



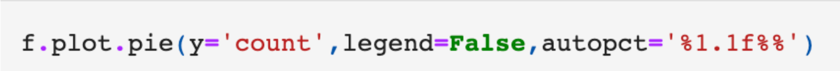
```
e=df1.groupBy("rideable_type").count().toPandas()
e.set_index('rideable_type', inplace=True)
```



```
e.plot.pie(y='count',legend=False,autopct='%1.1f%%')
```

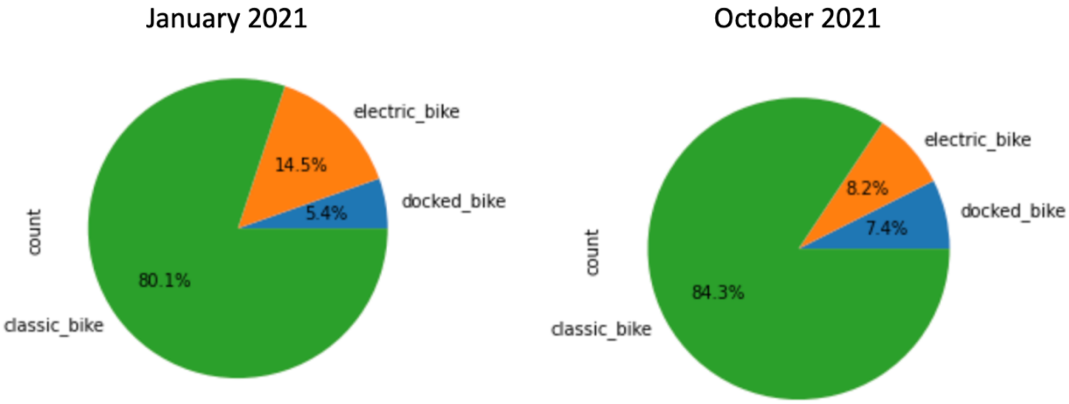


```
f=df2.groupBy("rideable_type").count().toPandas()
f.set_index('rideable_type', inplace=True)
```



```
f.plot.pie(y='count',legend=False,autopct='%1.1f%%')
```

Pie charts



The percent use of classis bikes and docked bikes increased while the use of electric bikes is decreased significantly. Many are opting for classic bikes over electric bikes.

I incorporated concepts taught in class in this project, I used virtualization by using VM in Jetstream and working on project in the webdesktop tool. Processing and Analytics module is used as I am loading and transforming the data using Pyspark analyzed and visualized data using python in Jupyter Notebook. There is an implicit pipeline implemented here:
Setting up the environment > Loading the data > Preprocessing, Cleaning, and transforming > Analyzing and visualizing Data > Interpreting results > Backup and maintaining using Git repo etc.,

I faced some challenges during this project, I had to recreate the pyspark instance as for some reason my jupyter notbook was not installing in my VM resulting in segmentation default error, after some research and I got it to install, the assignment Complete - Analyzing data with PySpark was helpful. I was pretty new to pyspark and learning and using it throughout the project was a bit challenging. Apart from this the project was really engaging and I learned a lot.

I have uploaded the ipynb file where I coded along with this report. It can also be found at https://github.com/UdaySharaff/Capital_BikeShare/blob/master/capital-bike-share-analysis.ipynb

Conclusion:

Working on this project refreshed many topics thought through out the course. Analyzing the data showed some pretty interesting results. The use of capital ride share service has changed significantly from start of this year clearly showing the effects of the pandemic and how the service bounced back. The rides taken in countries capital using this service increased about 3.5 times. The ratio of casual users to members increased, the use of classic bikes increased and electric bikes decreased, and there are changes in patterns of most frequent start and end destination pairs. The pandemic affected many businesses and is good to see this particular bike sharing survived and bloomed back into action.

References:

- https://sparkbyexamples.com/pyspark-tutorial/
- https://portal.xsede.org/jetstream
- https://www.capitalbikeshare.com/system-data
- https://towardsdatascience.com/how-to-get-started-with-pyspark-1adc142456ec
- https://www.fox5dc.com/news/dc-among-most-desired-travel-destinations-for-2021-as-covid-restrictions-loosen
- Complete - Analyzing data with PySpark