AirBnb Data Analysis using Spark

In this article we will use Hadoop and PySpark to perform data analysis on the AirBnb dataset. The dominant big data tools use Apache Hadoop and Apache Spark. Hadoop performance is superior to traditional RDBMS, but data processing using Hadoop has not been maximized. Thus, faster data processing is needed. One way to increase the speed of data processing is to apply spark to process data in HDFS.

Hadoop has several modules, including HDFS and YARN.

1. Hadoop Distributed File System (HDFS)

   HDFS is a distributed storage system for processing large amounts of data in a structured or unstructured manner. There are two HDFS components consisting of NameNode and DataNode. NameNode as a master node that stores the number and location of block data in the form of files and directories in charge of naming, closing and opening files. Meanwhile, DataNode performs the creation, deletion, and replication of data blocks on instruction from NameNode
2. Yet Another Resource Negotiator (YARN)

   Manages and monitors cluster nodes and resource usage to schedule jobs and tasks.

Apache Spark is a data processing framework that can quickly perform processing tasks on very large data sets.

PySpark is a collaboration between Python and Apache Spark to perform real-time large-scale data processing in a distributed environment using Python on the PySpark shell.

Launch Hadoop & Pyspark

start-dfs.sh is a shell script in Hadoop that is used to start the Hadoop Distributed FileSystem (HDFS) daemon process on the cluster. This script runs NameNode on the master machine and DataNode on the slave machine to manage metadata and data storage in the HDFS distributed file system. This script helps to efficiently organize and manage HDFS processes in a Hadoop environment.

start-yarn.sh is a shell script in Hadoop that is used to start the YARN (Yet Another Resource Negotiator) daemon process on the cluster. This script runs ResourceManager on the master machine and NodeManager on the slave machine. YARN is responsible for managing cluster resources and running MapReduce applications.

PySpark is a Python interface for Apache Spark, a fast parallel and distributed data processing framework. PySpark allows users to write Spark applications using the Python programming language. With PySpark, users can utilize the large-scale data processing capabilities of Apache Spark, as well as access data analysis libraries such as GraphX for graph computing and MLlib for machine learning.

```
$ start-dfs.sh
$ start-yarn.sh
$ pyspark
```

Import Libraries

```
import numpy as np
```

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

from pyspark.sql import *
from pyspark.sql import functions as func
from pyspark.sql import SparkSession
from pyspark.ml.stat import Correlation
from pyspark.sql.functions import col, desc, avg

# create the SparkSession
spark = SparkSession.builder.getOrCreate()
```

Usefulness of imported libraries:

- Numpy is used for math operations and manipulations.
- Pandas is used to read, manipulate, and analyze data in tabular format.
- Seaborn is used for statistical data visualization.
- Matplotlib is used to graph and plot data.

After that, this script also imports the PySpark SparkSession module to create Spark RDD, DataFrame, and DataSet. These spark functions play a role in creating DataFrames on CSV files for the purpose of accessing and processing data in the PySpark environment.

Load the Data

The dataset used is New York Airbnb Open Data from Kaggle. The dataset is stored with the variable name df.

Features in the Dataset column include the following:

id	Unique value identifier for each list of datasets.
name	Name of the rental property
host_id	Unique value identifier for each lodging provider
host_name	Name of lodging tenant
neighbourhood_group	City of each lodging location
neighbourhood	Area of each city
latitude	Geographic latitude
longitude	Lintang geografis
room_type	Room types consist of whole houses/apartments, private rooms, and shared rooms
price	Rental price
minimum_nights	Length of stay at the lodging

number_of_reviews	Number of reviews per lodging
last_review	Displays the time of the last incoming review
reviews_per_month	Average value of reviews provided by guests
calculated_host_listings_count	Total list of each host
availability_365	Number of rental rooms available

```
df=spark.read.csv("hdfs://localhost:9000/clarin/AB_NYC_2019.csv",header=True)
df.show()
```

```
+----+--------------------+-------+---------------+------------------+-----------------+--------+---------+----
-----------+-----+-------------+----------------+----------+----------------+------------------------------+--
--------------+
|  id|                name|host_id|      host_name|neighbourhood_group|     neighbourhood|latitude|longitude|
room_type|price|minimum_nights|number_of_reviews|last_review|reviews_per_month|calculated_host_listings_count|avai
lability_365|
+----+--------------------+-------+---------------+------------------+-----------------+--------+---------+----
-----------+-----+-------------+----------------+----------+----------------+------------------------------+--
--------------+
|2539|Clean & quiet apt...|   2787|           John|          Brooklyn|        Kensington|40.64749|-73.97237|   P
rivate room|  149|            1|               9| 2018-10-19|             0.21|                             6|
365|
|2595|Skylit Midtown Ca...|   2845|       Jennifer|         Manhattan|          Midtown|40.75362|-73.98377|Enti
re home/apt|  225|            1|              45| 2019-05-21|             0.38|                             2|
355|
|3647|THE VILLAGE OF HA...|   4632|      Elisabeth|         Manhattan|           Harlem|40.80902| -73.9419|   P
rivate room|  150|            3|               0|       null|             null|                             1|
365|
|3831|Cozy Entire Floor...|   4869|    LisaRoxanne|          Brooklyn|      Clinton Hill|40.68514|-73.95976|Enti
re home/apt|   89|            1|             270| 2019-07-05|             4.64|                             1|
194|
```

The **spark.read.csv()** function is used to read the dataset from kaggle in the form of a CSV file with the name "AB_NYC_2019.csv" located on HDFS.

The **header=True** argument is used to indicate that the CSV file has a header row that will be used as a column name in the DataFrame that will be created.

After the CSV file is successfully read, the script uses the **df.show()** function to display the contents of the DataFrame in tabular form on the console.

Data Profiling and Cleansing

```
# Check the data overview
print('Data overview')
df.printSchema()
```

```
Data overview
root
 |-- id: string (nullable = true)
 |-- name: string (nullable = true)
 |-- host_id: string (nullable = true)
 |-- host_name: string (nullable = true)
 |-- neighbourhood_group: string (nullable = true)
 |-- neighbourhood: string (nullable = true)
 |-- latitude: string (nullable = true)
 |-- longitude: string (nullable = true)
 |-- room_type: string (nullable = true)
 |-- price: string (nullable = true)
 |-- minimum_nights: string (nullable = true)
 |-- number_of_reviews: string (nullable = true)
 |-- last_review: string (nullable = true)
 |-- reviews_per_month: string (nullable = true)
 |-- calculated_host_listings_count: string (nullable = true)
 |-- availability_365: string (nullable = true)
```

The **print()** function is to display the text "Data overview" on the console.

The **df.printSchema()** function is to display the data structure in the df DataFrame. The function will display information about the column name, data type, and nullable status of each column in the DataFrame.

By displaying the data structure using **printSchema()**, we can understand more about the data being processed and ensure that the data type of each column is as expected.

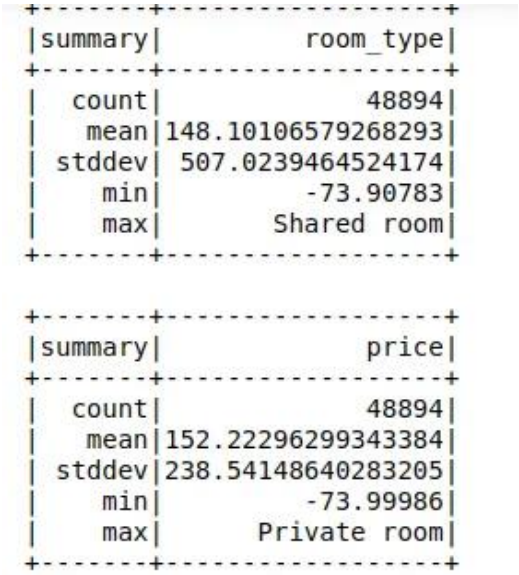
```
#Drop Rows with NULL Values
df.na.drop().show()
```

```
+----+--------------------+-------+---------------+-----------------+---------------+--------+---------+---------+----
----------+-----+-------------+----------------+----------+---------------+----------------------------+--
-------------+
|  id|                name|host_id|      host_name|neighbourhood_group|  neighbourhood|latitude|longitude|
room_type|price|minimum_nights|number_of_reviews|last_review|reviews_per_month|calculated_host_listings_count|avai
lability_365|
+----+--------------------+-------+---------------+-----------------+---------------+--------+---------+---------+----
----------+-----+-------------+----------------+----------+---------------+----------------------------+--
-------------+
|2539|Clean & quiet apt...|   2787|           John|         Brooklyn|    Kensington|40.64749|-73.97237|   P
rivate room|  149|            1|               9| 2018-10-19|           0.21|                           6|
365|
|2595|Skylit Midtown Ca...|   2845|       Jennifer|        Manhattan|       Midtown|40.75362|-73.98377|Enti
re home/apt|  225|            1|              45| 2019-05-21|           0.38|                           2|
355|
|3831|Cozy Entire Floor...|   4869|    LisaRoxanne|         Brooklyn|  Clinton Hill|40.68514|-73.95976|Enti
re home/apt|   89|            1|             270| 2019-07-05|           4.64|                           1|
194|
|5022|Entire Apt: Spaci...|   7192|          Laura|        Manhattan|   East Harlem|40.79851|-73.94399|Enti
re home/apt|   80|           10|               9| 2018-11-19|           0.10|                           1|
0|
|5099|Large Cozy 1 BR A...|   7322|          Chris|        Manhattan|   Murray Hill|40.74767|  -73.975|Enti
re home/apt|  200|            3|              74| 2019-06-22|           0.59|                           1|
129|
```

The dataset obtained from the source has several null values, so each row of data needs to be deleted so that optimal analysis results can be obtained.

```
# Check Statistical Information in Each Column
for col in df.columns:
    df.describe([col]).show()
```

The command serves to display summary statistics of each column in the DataFrame using PySpark by examining the dataset in the form of statistical analysis that helps us know the calculation of each column. The following table displays several columns, namely room type and price so that a summary is obtained.



```
+-------+------------------+
|summary|         room_type|
+-------+------------------+
|  count|             48894|
|   mean|148.10106579268293|
| stddev| 507.0239464524174|
|    min|          -73.90783|
|    max|       Shared room|
+-------+------------------+

+-------+------------------+
|summary|             price|
+-------+------------------+
|  count|             48894|
|   mean|152.22296299343384|
| stddev|238.54148640283205|
|    min|          -73.99986|
|    max|      Private room|
+-------+------------------+
```

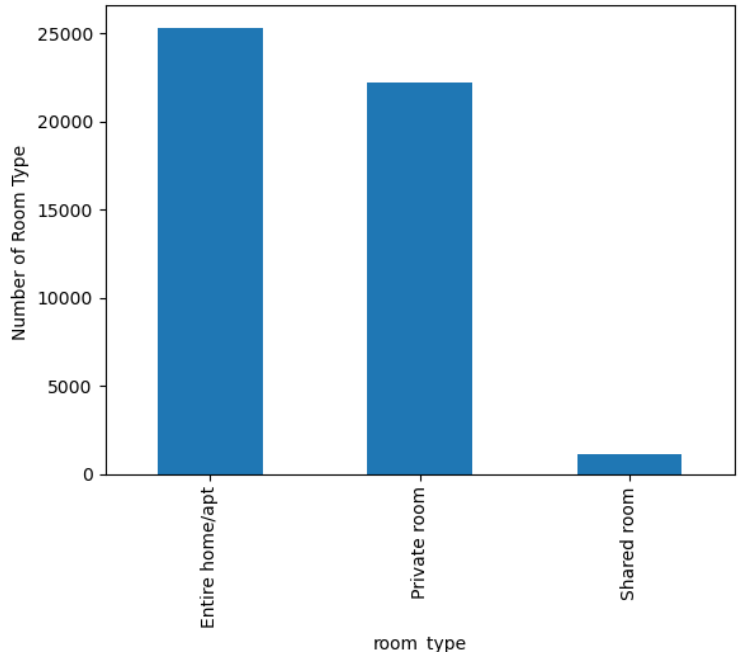
Exploratory Data Analysis

Exploratory Data Analysis (EDA) is a data analysis process that aims to understand the characteristics of the data, find patterns or relationships in the data, and identify anomalies in the data. EDA is conducted using data visualization techniques and descriptive statistics to explore the information contained in the data. The purpose of EDA is to gain a deeper understanding of the data and ensure that the data can be used for the desired analysis purposes. EDA is often the first stage in the data analysis process and can help direct subsequent analysis. Here are some of the results of the data analysis we collected.

What are the amenities that affect the price?

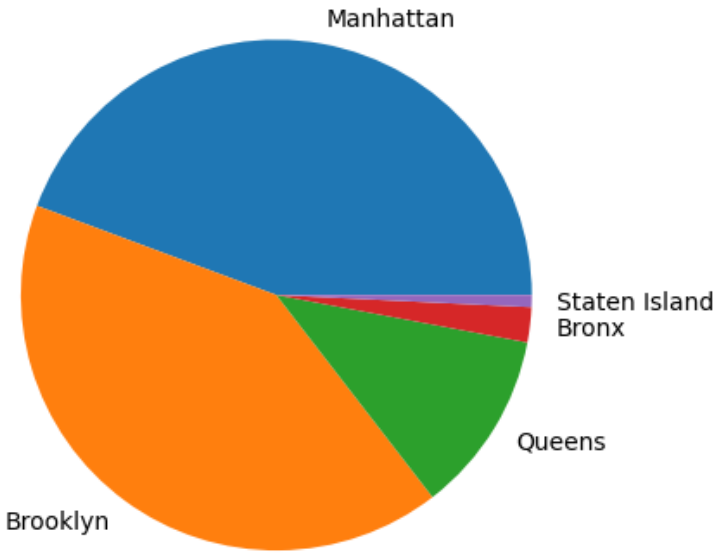
We would like to look at some of the facilities available in influencing the price level of lodging by obtaining the number of room types, neighborhood groups, and minimum nights.

Analysis of Number on Room Type List



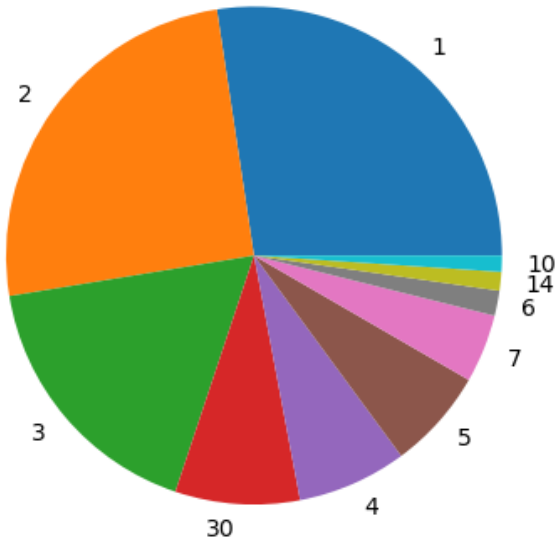
The results of the barplot show that the room types in the entire home have the highest number of **25000**.

Count Analysis on Neighborhood Group List



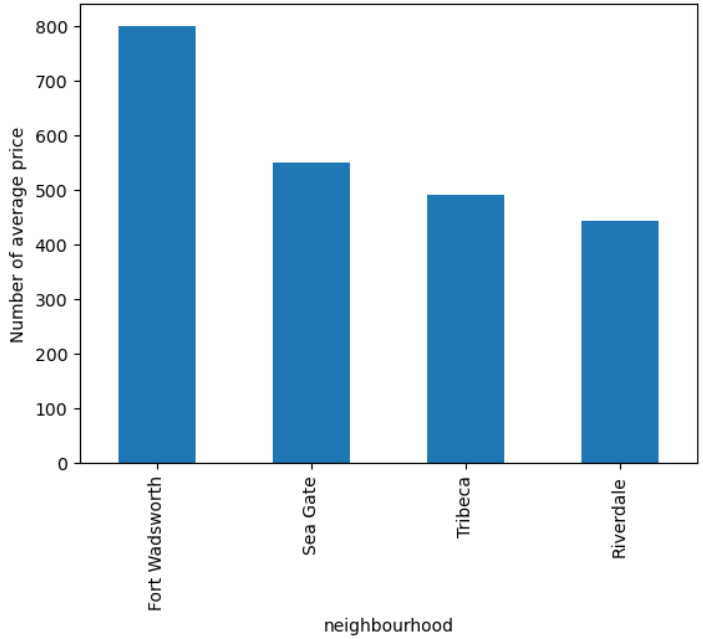
The results of the above diagram obtained Manhattan and Brooklyn as the most Neighborhood Group followed by **Queens**, **Bronx**, and **Staten Island**.

Analysis of the Number on the Minimum Nights List



The diagram above shows the minimum number of nights available is dominated by numbers **1, 2,** and **3** that can be booked by guests.

Neighborhoods with the Highest Average Price in each Neighborhood Groups



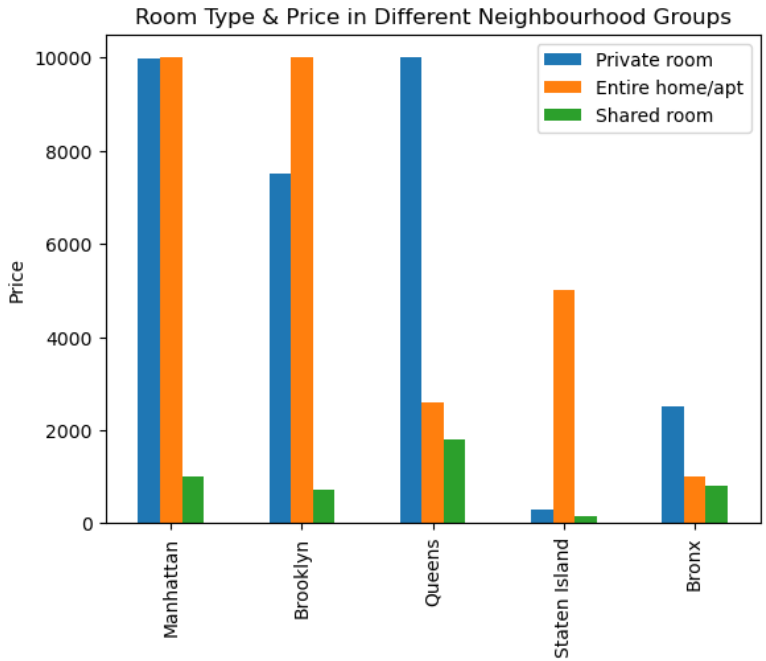
The visualization of the barplot displays the top four neighbourhoods that have the highest average price. The first position is Fort Wadsworth with a price of 800 and Sea Gate is in second place.

Analysis of the Most Expensive Price in Each Neighborhood Group

name	neighbourhood_group	max_price
Luxury 1 bedroom apt. -stunning Manhattan views	Brooklyn	10000
Furnished room in Astoria apartment	Queens	10000
1-BR Lincoln Center	Manhattan	10000
Quiet, Clean, Lit @ LES & Chinatown	Manhattan	9999
Spanish Harlem Apt	Manhattan	9999
2br - The Heart of NYC: Manhattans Lower East Side	Manhattan	9999
Beautiful/Spacious 1 bed luxury flat-TriBeCa/Soho	Manhattan	8500
Film Location	Brooklyn	8000
East 72nd Townhouse by (Hidden by Airbnb)	Manhattan	7703
Gem of east Flatbush	Brooklyn	7500
70' Luxury MotorYacht on the Hudson	Manhattan	7500
3000 sq ft daylight photo studio	Manhattan	6800
Luxury TriBeCa Apartment at an amazing price	Manhattan	6500
SUPER BOWL Brooklyn Duplex Apt!!	Brooklyn	6500
Park Avenue Mansion by (Hidden by Airbnb)	Manhattan	6419
UWS 1BR w/backyard + block from CP	Manhattan	6000
Luxury townhouse Greenwich Village	Manhattan	6000
SuperBowl Penthouse Loft 3,000 sqft	Manhattan	5250
Midtown Manhattan great location (Gramacy park)	Manhattan	5100
NearWilliamsburg bridge 11211 BK	Brooklyn	5000

The table shows that the three neighborhood groups with the highest prices are Brooklyn, Queens, and Manhattan. There are various lodgings available for guests within the same location. However, it offers a wide range of prices that guests can choose from by considering various aspects of the services available from each residence.

Comparison of Room Types and Prices in Different Neighborhood Groups



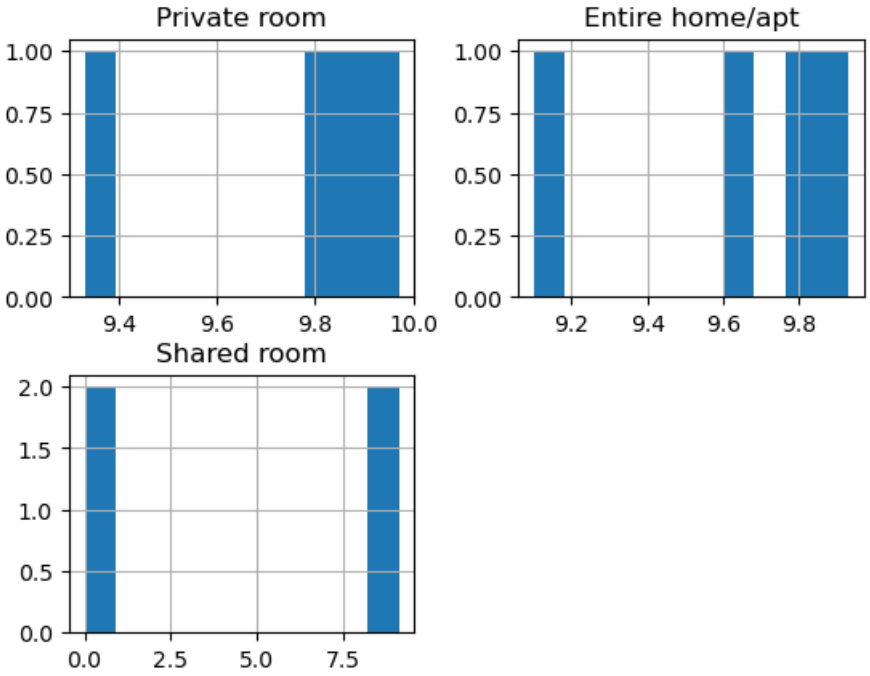
In the type of private room obtained Queens as a neighborhood group with the highest price of 10000. Then for the type of entire home, Manhattan and Brooklyn are obtained as the neighborhood group with the highest price of 10000 while the type of shared room is obtained by Queens with the highest price.

10 Most Reviewed Places in New York City

name	neighbourhood_group	max_review
COZY Room for Female Guests	Brooklyn	9.97
Park Slope Villa	Brooklyn	9.93
A Little Sumptin' on Sumpter	Brooklyn	9.85
Accessible Two Bedded Suite With Kitchen near MSG	Manhattan	9.85
Peaches Paradise.	Queens	9.83
Private room near LGA Airport with queen bed	Queens	9.82
Best Value ♥Memorable Vacation	Manhattan	9.78
Master Bedroom with Full Bath & Manhattan View	Queens	9.74
Cozy and private room close to LGA.	Queens	9.73
Modern 2 Bedroom Walk in. 3 min from Subway !	Brooklyn	9.68

One of the lodging places called COZY Room for Female Guests has the highest number of reviews with 9.97 which is in Brooklyn. The top ten reviews are dominated by Brooklyn and Queens.

How are Monthly Reviews with Room Types in each Neighborhood Group



In the histogram, there are three types of rooms namely private room, entire room, and shared room. The private room and entire home types feature Brooklyn as the highest review while the shared room type features Queens as the highest review with 9.13 reviews.

Conclusion

- Brooklyn, Queens, and Manhattan have the largest number of neighborhood groups. The number of Manhattan is 21,594, Brooklyn is 20,055, and Queens is 5,630. In addition, it has a high price with each price of each neighborhood group being a maximum of 10,000.
- Whole home room types tend to offer higher prices than private rooms and shared rooms. For example, Staten Island has a price of 5000 for an entire home, a private room at 300 and a shared room at 150.
- Neighborhood groups referring to Brooklyn have the most monthly reviews on private room and entire home types with 9.85 and 9.78 reviews respectively.
- Reviews and price have a significant correlation. If the reviews get a high number, the price offered is also high. This can be seen in the Manhattan neighborhood group where the average review is 9 and the average price is above 5000.