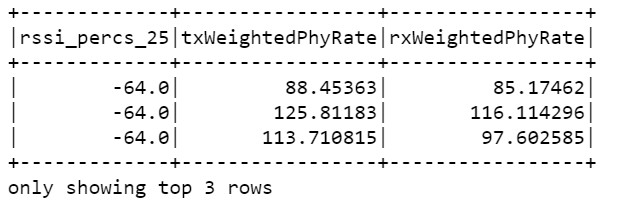
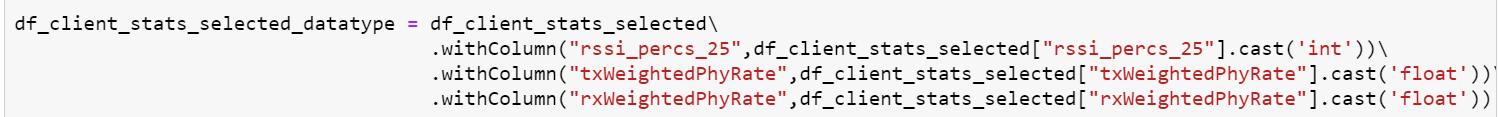
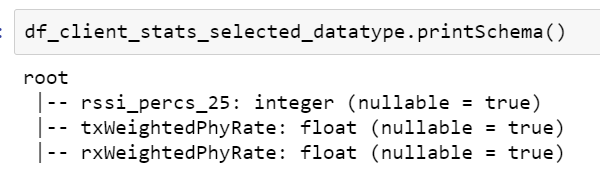
Pipeline – Pseudo Code for Client\_Stats Analysis

1. Read client\_stats data from given csv file (client\_stats \_ sample\_0225part1.csv ) and select the columns rssi\_percs\_25, txWeightedPhyRate, rxWeightedPhyRate.

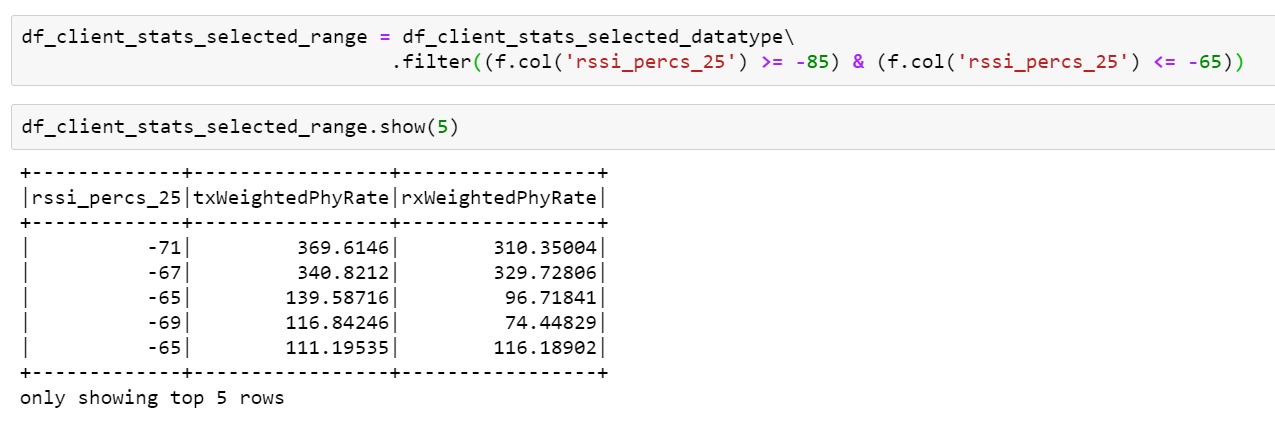


1. Convert the datatype of fields rssi\_percs\_25, txWeightedPhyRate, rxWeightedPhyRate to integer/float as they are in string format.

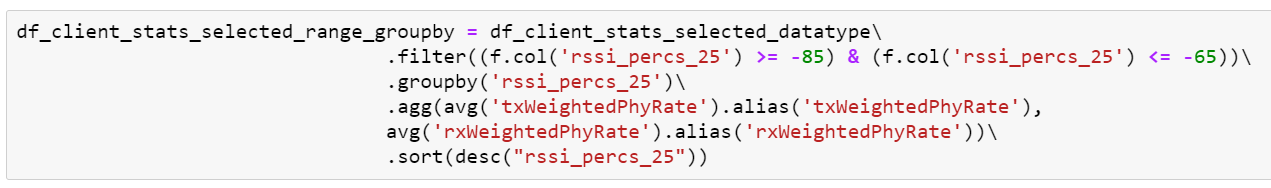




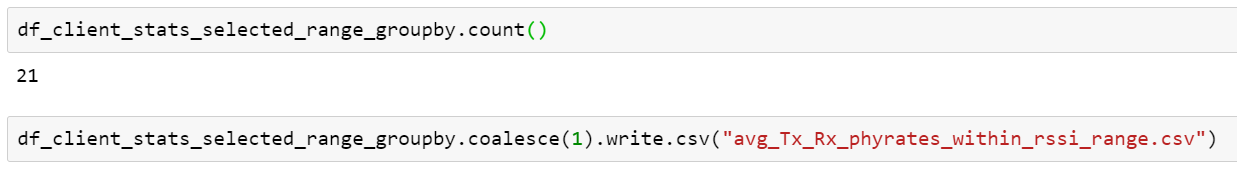
1. Filter the dataframe with the range of RSSI values between -85 to -65.



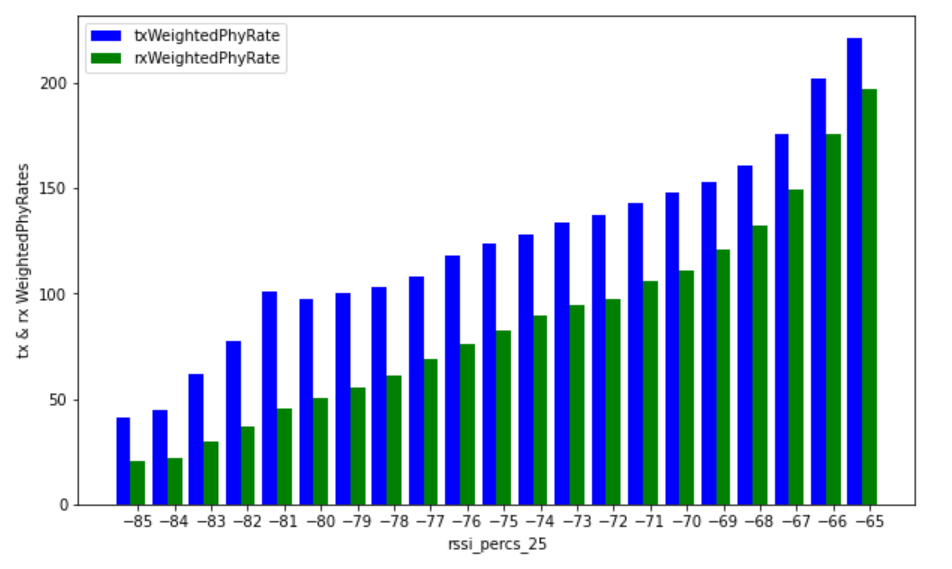
1. Filter the spark data frame with the range of RSSI values between -85 to -65 and find out the averages of Tx & Rx Phyrates. (optional step: Sort data frame by RSSI column finally)



1. Check for count of spark data frame and export it to csv file.



1. The job would run on daily basis and the output data frame must be written to elastic search via ansible playbooks & Kafka topic.
2. Retrieve the data from ES and produce the below graph in Kibana.



Flow Diagram:

HDFS

Kibana

Elastic Search

Kafka topic

Ansible Playbooks

1. Read data from hdfs and process it as per above steps and push back the resulted dataframe to hdfs.

2. Create an Ansible playbook with ES index and Kafka topic.

3. When the scheduled job along with it's configurations gets executed in Airflow, Kafka topic sends the data to Elastic Search.

4. Data gets stored in ES with an index and can be retrieved in kibana to plot a graph.