WATER QUALITY PREDICTION

ABSTRACT:

We all know water is one of the most essential resource for our living. But as the development is increasing, we are exploiting water by wasting it and treating it with harmful materials which makes water impure and unfit for use. This is the reason it is very important to know the quality of water. This project is based on water quality prediction. In this project, water quality index (WQI) and quality status of water is predicted through some parameters that affects water quality. In this notebook we have performed Data Cleaning steps and did Exploratory Data Analysis. Then we have did some calculations as the data does not contain the column which can be used for prediction. Then we have created 2 models for prediction. The first model is Linear Regression model and The second model is Logistic Regression model. We have done this project with pyspark and other spark packages.

ABOUT THE DATA:

The data contains water quality parameters of different rivers of India. There are 8 parameters and each parameter is the average values measured over a period of time.

AIM:

The main aim of this project is to predict the water quality index , the quality status of the water and various parameter influencing it, using spark machine learning packages.

SPARK & PYSPARK INSTALLATION:

- Install packages required for Spark

  sudo apt install default-jdk
  Scala git -y

- Verify the installed packages

  java -version;
  javac -version;
  scala -version;
  git –version;

- Use the wget command and the direct link to download spark

  wget https://downloads.apache.org/spark/spark-3.0.1/spark-3.0.1-bin-hadoop2.7.tgz

- Now extract using tar

  tar xvf spark-*

- Use this command to move the unpacked directory

  sudo mv spark-3.0.1-bin-hadoop2.7 /opt/spark

- Now configure spark environment using these commands

  echo "export SPARK_HOME=/opt/spark" >> ~/.profile
  echo "export PATH=$PATH:$SPARK_HOME/bin:$SPARK_HOME/sbin" >> ~/.profile

echo "export PYSPARK_PYTHON=/usr/bin/python3" >> ~/.profile

- Open profile using nano and paste these commands

  Nano .profile

  export SPARK_HOME=/opt/spark
  export PATH=$PATH:$SPARK_HOME/bin:$SPARK_HOME/sbin
  export PYSPARK_PYTHON=/usr/bin/python3

- Start standalone spark master server

  Start-master.sh

- To view the Spark Web user interface use

  http://127.0.0.1:8080/

- Test spark shell

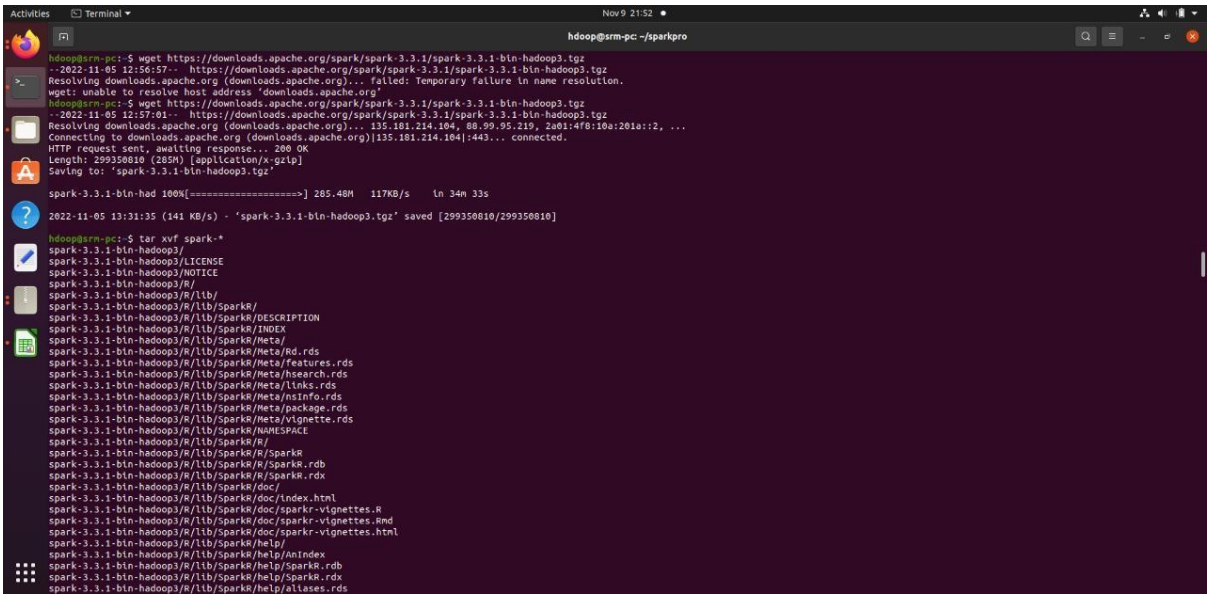
  spark-shell

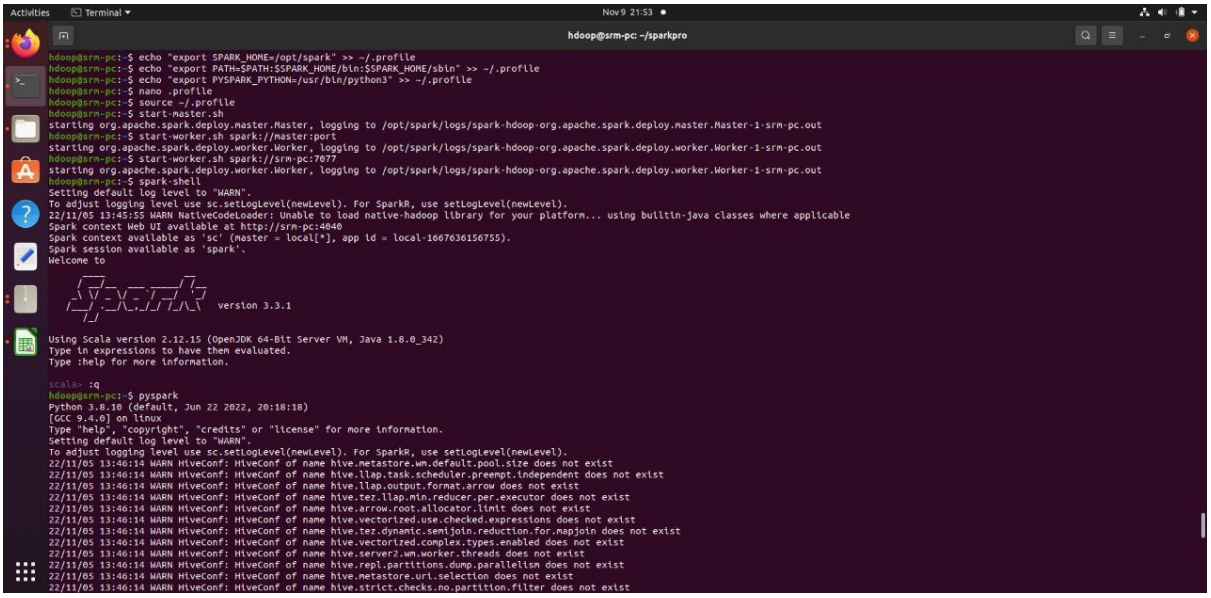
- Install pyspark using

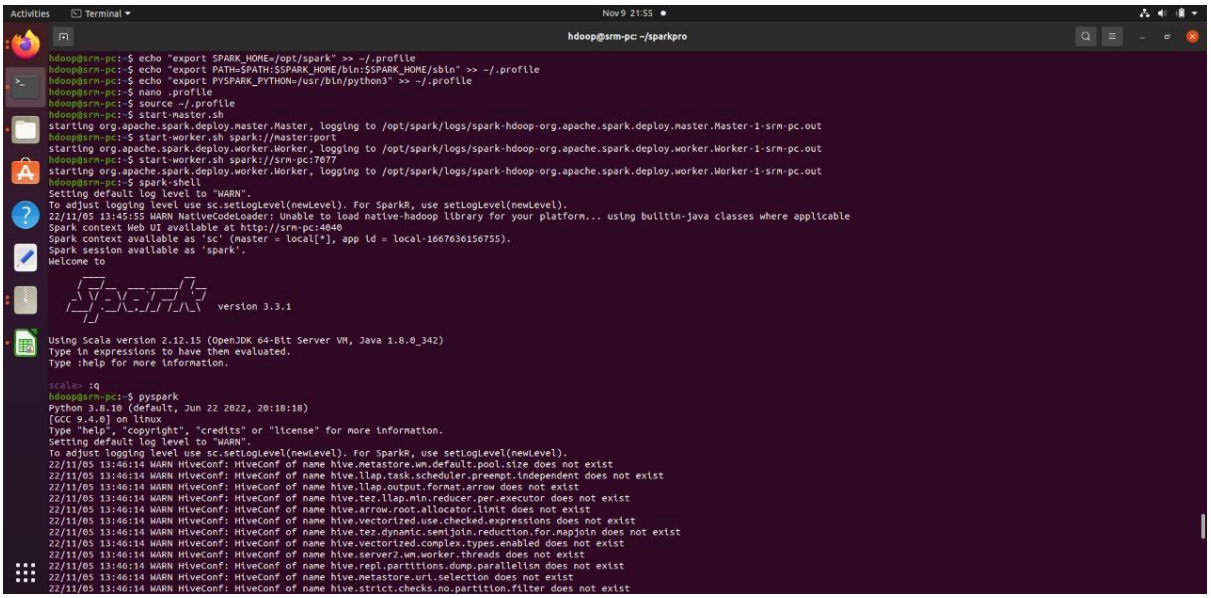
  Python3 pip install pyspark

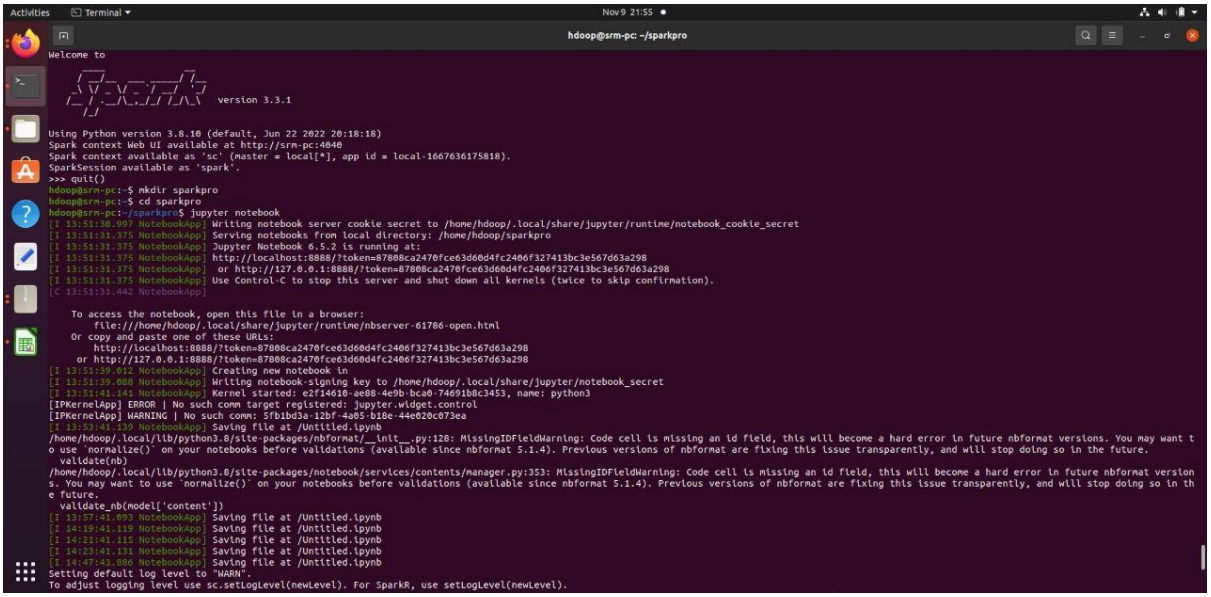
- Test pyspark

  pyspark


Activities
Terminal
Nov 9 21:52
hdoop@srm-pc: ~/sparkpro
hdoop@srm-pc:~$ wget https://downloads.apache.org/spark/spark-3.3.1/spark-3.3.1-bin-hadoop3.tgz
--2022-11-05 12:56:57-- https://downloads.apache.org/spark/spark-3.3.1/spark-3.3.1-bin-hadoop3.tgz
Resolving downloads.apache.org (downloads.apache.org)... failed: Temporary failure in name resolution.
wget: unable to resolve host address 'downloads.apache.org'
hdoop@srm-pc:~$ wget https://downloads.apache.org/spark/spark-3.3.1/spark-3.3.1-bin-hadoop3.tgz
--2022-11-05 12:57:01-- https://downloads.apache.org/spark/spark-3.3.1/spark-3.3.1-bin-hadoop3.tgz
Resolving downloads.apache.org (downloads.apache.org)... 135.181.214.104, 88.99.95.219, 2a01:4f8:10a:201a::2, ...
Connecting to downloads.apache.org (downloads.apache.org)|135.181.214.104|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 299350810 (285M) [application/x-gzip]
Saving to: 'spark-3.3.1-bin-hadoop3.tgz'
spark-3.3.1-bin-had 100%[===================>] 285.48M 117KB/s in 34m 33s
2022-11-05 13:31:35 (141 KB/s) - 'spark-3.3.1-bin-hadoop3.tgz' saved [299350810/299350810]
hdoop@srm-pc:~$ tar xvf spark-*
spark-3.3.1-bin-hadoop3/
spark-3.3.1-bin-hadoop3/LICENSE
spark-3.3.1-bin-hadoop3/NOTICE
spark-3.3.1-bin-hadoop3/R/
spark-3.3.1-bin-hadoop3/R/lib/
spark-3.3.1-bin-hadoop3/R/lib/SparkR/
spark-3.3.1-bin-hadoop3/R/lib/SparkR/DESCRIPTION
spark-3.3.1-bin-hadoop3/R/lib/SparkR/INDEX
spark-3.3.1-bin-hadoop3/R/lib/SparkR/Meta/
spark-3.3.1-bin-hadoop3/R/lib/SparkR/Meta/Rd.rds
spark-3.3.1-bin-hadoop3/R/lib/SparkR/Meta/features.rds
spark-3.3.1-bin-hadoop3/R/lib/SparkR/Meta/hsearch.rds
spark-3.3.1-bin-hadoop3/R/lib/SparkR/Meta/links.rds
spark-3.3.1-bin-hadoop3/R/lib/SparkR/Meta/nsInfo.rds
spark-3.3.1-bin-hadoop3/R/lib/SparkR/Meta/package.rds
spark-3.3.1-bin-hadoop3/R/lib/SparkR/Meta/vignette.rds
spark-3.3.1-bin-hadoop3/R/lib/SparkR/NAMESPACE
spark-3.3.1-bin-hadoop3/R/lib/SparkR/R/
spark-3.3.1-bin-hadoop3/R/lib/SparkR/R/SparkR
spark-3.3.1-bin-hadoop3/R/lib/SparkR/R/SparkR.rdb
spark-3.3.1-bin-hadoop3/R/lib/SparkR/R/SparkR.rdx
spark-3.3.1-bin-hadoop3/R/lib/SparkR/doc/
spark-3.3.1-bin-hadoop3/R/lib/SparkR/doc/index.html
spark-3.3.1-bin-hadoop3/R/lib/SparkR/doc/sparkr-vignettes.R
spark-3.3.1-bin-hadoop3/R/lib/SparkR/doc/sparkr-vignettes.Rmd
spark-3.3.1-bin-hadoop3/R/lib/SparkR/doc/sparkr-vignettes.html
spark-3.3.1-bin-hadoop3/R/lib/SparkR/help/
spark-3.3.1-bin-hadoop3/R/lib/SparkR/help/AnIndex
spark-3.3.1-bin-hadoop3/R/lib/SparkR/help/SparkR.rdb
spark-3.3.1-bin-hadoop3/R/lib/SparkR/help/SparkR.rdx
spark-3.3.1-bin-hadoop3/R/lib/SparkR/help/aliases.rds


Activities
Terminal
Nov 9 21:53
hdoop@srm-pc: ~/sparkpro
hdoop@srm-pc:~$ echo "export SPARK_HOME=/opt/spark" >> ~/.profile
hdoop@srm-pc:~$ echo "export PATH=$PATH:$SPARK_HOME/bin:$SPARK_HOME/sbin" >> ~/.profile
hdoop@srm-pc:~$ echo "export PYSPARK_PYTHON=/usr/bin/python3" >> ~/.profile
hdoop@srm-pc:~$ nano .profile
hdoop@srm-pc:~$ source ~/.profile
hdoop@srm-pc:~$ start-master.sh
starting org.apache.spark.deploy.master.Master, logging to /opt/spark/logs/spark-hdoop-org.apache.spark.deploy.master.Master-1-srm-pc.out
hdoop@srm-pc:~$ start-worker.sh spark://master:port
starting org.apache.spark.deploy.worker.Worker, logging to /opt/spark/logs/spark-hdoop-org.apache.spark.deploy.worker.Worker-1-srm-pc.out
hdoop@srm-pc:~$ start-worker.sh spark://srm-pc:7077
starting org.apache.spark.deploy.worker.Worker, logging to /opt/spark/logs/spark-hdoop-org.apache.spark.deploy.worker.Worker-1-srm-pc.out
hdoop@srm-pc:~$ spark-shell
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
22/11/05 13:45:55 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Spark context Web UI available at http://srm-pc:4040
Spark context available as 'sc' (master = local[*], app id = local-1667636156755).
Spark session available as 'spark'.
Welcome to
version 3.3.1
Using Scala version 2.12.15 (OpenJDK 64-Bit Server VM, Java 1.8.0_342)
Type in expressions to have them evaluated.
Type :help for more information.
scala> :q
hdoop@srm-pc:~$ pyspark
Python 3.8.10 (default, Jun 22 2022, 20:18:18)
[GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.metastore.wm.default.pool.size does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.llap.task.scheduler.preempt.independent does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.llap.output.format.arrow does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.tez.llap.min.reducer.per.executor does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.arrow.root.allocator.limit does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.vectorized.use.checked.expressions does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.tez.dynamic.semijoin.reduction.for.mapjoin does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.vectorized.complex.types.enabled does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.server2.wm.worker.threads does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.repl.partitions.dump.parallelism does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.metastore.uri.selection does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.strict.checks.no.partition.filter does not exist


Activities
Terminal
Nov 9 21:55
hdoop@srm-pc: ~/sparkpro
hdoop@srm-pc:~$ echo "export SPARK_HOME=/opt/spark" >> ~/.profile
hdoop@srm-pc:~$ echo "export PATH=$PATH:$SPARK_HOME/bin:$SPARK_HOME/sbin" >> ~/.profile
hdoop@srm-pc:~$ echo "export PYSPARK_PYTHON=/usr/bin/python3" >> ~/.profile
hdoop@srm-pc:~$ nano .profile
hdoop@srm-pc:~$ source ~/.profile
hdoop@srm-pc:~$ start-master.sh
starting org.apache.spark.deploy.master.Master, logging to /opt/spark/logs/spark-hdoop-org.apache.spark.deploy.master.Master-1-srm-pc.out
hdoop@srm-pc:~$ start-worker.sh spark://master:port
starting org.apache.spark.deploy.worker.Worker, logging to /opt/spark/logs/spark-hdoop-org.apache.spark.deploy.worker.Worker-1-srm-pc.out
hdoop@srm-pc:~$ start-worker.sh spark://srm-pc:7077
starting org.apache.spark.deploy.worker.Worker, logging to /opt/spark/logs/spark-hdoop-org.apache.spark.deploy.worker.Worker-1-srm-pc.out
hdoop@srm-pc:~$ spark-shell
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
22/11/05 13:45:55 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Spark context Web UI available at http://srm-pc:4040
Spark context available as 'sc' (master = local[*], app id = local-1667636156755).
Spark session available as 'spark'.
Welcome to
version 3.3.1
Using Scala version 2.12.15 (OpenJDK 64-Bit Server VM, Java 1.8.0_342)
Type in expressions to have them evaluated.
Type :help for more information.
scala> :q
hdoop@srm-pc:~$ pyspark
Python 3.8.10 (default, Jun 22 2022, 20:18:18)
[GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.metastore.wm.default.pool.size does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.llap.task.scheduler.preempt.independent does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.llap.output.format.arrow does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.tez.llap.min.reducer.per.executor does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.arrow.root.allocator.limit does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.vectorized.use.checked.expressions does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.tez.dynamic.semijoin.reduction.for.mapjoin does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.vectorized.complex.types.enabled does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.server2.wm.worker.threads does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.repl.partitions.dump.parallelism does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.metastore.uri.selection does not exist
22/11/05 13:46:14 WARN HiveConf: HiveConf of name hive.strict.checks.no.partition.filter does not exist


Activities
Terminal
Nov 9 21:55
hdoop@srm-pc: ~/sparkpro
Welcome to
version 3.3.1
Using Python version 3.8.10 (default, Jun 22 2022 20:18:18)
Spark context Web UI available at http://srm-pc:4040
Spark context available as 'sc' (master = local[*], app id = local-1667636175818).
SparkSession available as 'spark'.
>>> quit()
hdoop@srm-pc:~$ mkdir sparkpro
hdoop@srm-pc:~$ cd sparkpro
hdoop@srm-pc:~/sparkpro$ jupyter notebook
[I 13:51:30.997 NotebookApp] Writing notebook server cookie secret to /home/hdoop/.local/share/jupyter/runtime/notebook_cookie_secret
[I 13:51:31.375 NotebookApp] Serving notebooks from local directory: /home/hdoop/sparkpro
[I 13:51:31.375 NotebookApp] Jupyter Notebook 6.5.2 is running at:
[I 13:51:31.375 NotebookApp] http://localhost:8888/?token=87808ca2470fce63d60d4fc2406f327413bc3e567d63a298
[I 13:51:31.375 NotebookApp] or http://127.0.0.1:8888/?token=87808ca2470fce63d60d4fc2406f327413bc3e567d63a298
[I 13:51:31.375 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 13:51:31.442 NotebookApp]
To access the notebook, open this file in a browser:
file:///home/hdoop/.local/share/jupyter/runtime/nbserver-61786-open.html
Or copy and paste one of these URLs:
http://localhost:8888/?token=87808ca2470fce63d60d4fc2406f327413bc3e567d63a298
or http://127.0.0.1:8888/?token=87808ca2470fce63d60d4fc2406f327413bc3e567d63a298
[I 13:51:39.012 NotebookApp] Creating new notebook in
[I 13:51:39.088 NotebookApp] Writing notebook-signing key to /home/hdoop/.local/share/jupyter/notebook_secret
[I 13:51:41.141 NotebookApp] Kernel started: e2f14610-ae88-4e9b-bca0-74691b8c3453, name: python3
[IPKernelApp] ERROR | No such comm target registered: jupyter.widget.control
[IPKernelApp] WARNING | No such comm: 5fb1bd3a-12bf-4a05-b18e-44e020c073ea
[I 13:52:41.135 NotebookApp] Saving file at /Untitled.ipynb
/home/hdoop/.local/lib/python3.8/site-packages/nbformat/__init__.py:128: MissingIDFieldWarning: Code cell is missing an id field, this will become a hard error in future nbformat versions. You may want to use `normalize()` on your notebooks before validations (available since nbformat 5.1.4). Previous versions of nbformat are fixing this issue transparently, and will stop doing so in the future.
validate(nb)
/home/hdoop/.local/lib/python3.8/site-packages/notebook/services/contents/manager.py:353: MissingIDFieldWarning: Code cell is missing an id field, this will become a hard error in future nbformat versions. You may want to use `normalize()` on your notebooks before validations (available since nbformat 5.1.4). Previous versions of nbformat are fixing this issue transparently, and will stop doing so in the future.
validate_nb(model['content'])
[I 13:57:41.093 NotebookApp] Saving file at /Untitled.ipynb
[I 14:19:41.119 NotebookApp] Saving file at /Untitled.ipynb
[I 14:23:41.115 NotebookApp] Saving file at /Untitled.ipynb
[I 14:23:41.131 NotebookApp] Saving file at /Untitled.ipynb
[I 14:47:43.886 NotebookApp] Saving file at /Untitled.ipynb
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).

CODE:

WATER QUALITY PREDICTION

Importing libraries

In [2]:
```
import os
import pandas as pd
import numpy as np
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns
import geopandas as gpd
import warnings
warnings.filterwarnings("ignore")

from pylab import *
from pyspark.sql.functions import udf, concat, col, lit
from pyspark import SparkConf, SparkContext
from pyspark.sql import SparkSession, SQLContext

from pyspark.sql.types import *
import pyspark.sql.functions as F
sc = SparkContext.getOrCreate(SparkConf().setMaster("local[*]"))
from pyspark.sql import SparkSession
spark = SparkSession \
    .builder \
    .getOrCreate()
sqlContext = SQLContext(sc)
```

Reading the dataset and describing

In [4]:
```
df = spark.read.format("csv").option("header", "true").load('/home/hdoop/sparkpro/waterquality.csv')
```

In [5]:
```
df.show(5)
```

```
+------------+--------------------+-----------+----+---+---+------------+---+------------------+--------------+----
----------+
|STATION CODE|           LOCATIONS|      STATE|TEMP| DO| pH|CONDUCTIVITY|BOD|NITRATE_N_NITRITE_N|FECAL_COLIFORM|TOTA
L_COLIFORM|
+------------+--------------------+-----------+----+---+---+------------+---+------------------+--------------+----
----------+
|        1312|GODAVARI AT JAYAK...|MAHARASHTRA|29.2|6.4|8.1|         735|3.4|                 2|             3|
73|
|        2177|GODAVARI RIVER NE...|MAHARASHTRA|24.5|  6|  8|         270|3.1|                 2|            72|
182|
|        2182|GODAVARI RIVER AT...|MAHARASHTRA|25.8|5.5|7.8|         355|4.2|                 9|            59|
133|
|        2179|GODAVARI RIVER AT...|MAHARASHTRA|24.8|5.5|7.8|         371|5.6|              3.55|            90|
283|
|        2183|GODAVARI RIVER AT...|MAHARASHTRA|25.7|5.7|7.9|         294|3.2|              2.69|            45|
132|
+------------+--------------------+-----------+----+---+---+------------+---+------------------+--------------+----
----------+
only showing top 5 rows
```

```
In [6]:  dI . dtypes

Out[6]: [('STAT ION CBDE ',' st ring ') ,
         ('LOCATIONS ',' st ring '),
         ('STATE ',' st rang ' 1 ,
         ('TENP ',' s I ing ') ,
         ('DO ',  ' st  I ng ') ,
         ('pH ',  ' st  ring ') ,
         ('COND UC TI V I TY ',' S't FT rt g ') ,
         ('BOD ',' st ring ') ,
         ('NITRATE  N  NIJ  RITE  N ',  ' st  rang ') ,
         ('FECAL COLIFORM ','st ring '),
         ('T0TAL"C0LI FDRN ',  ' st ring ') ]
```

```
In [7]:  dt.nead()

 '?" +.|' "   ROw (STATION   CODE=' 1312 ',   LDCA+IDNS=' GODAVARI   AT  3AYAKNADI   DAM,   AURNAGABAD , MAHARASHTRA ',  5TA^E= ' NAHARASH+ RA ',  TENP=' 2
  9.2 '   DO= '6. 4'     pH= ' 8.1 '   CONDUCTIVITY= ' 735 '   BOD= ' 3. 4 '   NITRATE N NITRITE  N= ' 2 '   FECAL  COLIFDRH= ' 3 '   TOTAL  COLIFORM
 ='73'1                '                  '                                                                                                          '
```

```
In [l0]:  d{.describe()

El .it " I'd' " Data rane [ s unrra ry: st rang , STATION CODE : s t ring . LOCATIONS : st rin.g          STATE : st ring , TEMP : st ring , DD : s t ring . pH : st r
  ing ,  CONDUCT IVI^Y :  st rang ,  BOD :  st  ri  ng ,  NITRATE_N_NI TRITE  N  st  ring ,  FECAL  C0LI FORM :  st  ring ,  +0TAL_C DLI FORM :  st  ring ]
```

Data cleaning

```
In lll]:  f rom py s parh . so t. t ype s znpo rt F toatType
```

As we can see aoove that all the columns have string data types, out fDr the calculation of water quality index we need to convert tnem in float data type. So we will conven the reauired columns to flDat data type

```
IU   12 ]:  d   =  df . wet hCo\umn I "TEr'1P'  , df [ ' TE'1.' " ]  ca st (Fl oat Type ( ) ) )
            d = df . wit hCo\umn I " pH ' , df [ " pH. " ] . c a st (F\ a a t Type ( 111
            d   = df . wet hCo\umn I "DO ' , df [ "DO " ] . c a st (F\ a a t Type ( 111
            d  = df . wet hCo\umn I " C0IJDUCTI\'ITY", df [ 'CONDUC  I CITY" ]  cast  FT oat Type ( ) ) )
            d  =  df . wet hCo\umn I " BOD ", df [ 'BOD " ] . ca st (Fl a a t Type ( 1 ) )
            d = d f . wet hCo\ umn I " NITF' TE  £J NI TRITE IN ' , d [ " I4 TTéATE IN NITRITE £J " ] . ca st *boat type ( ) ) )
            d =  d f . wit hCo\ umn I " FEC ML CDLI F0FJU' , df [ ' FECAL COLIFO F.H' ] . cast I Ft DatType ) ) )
```

```
In [13]:  d . d types

Out[13]: [('STATI 0N CODE,'      ' st ring ') ,
          ('LOCATIONS ',' st ri ng ') .
          ('STATE ',' st rang ',1
          I ' GEN" ' ,' +toat ' › ,
          ('DO ',' +\oat ') ,

          (' CONDUCTIVITY ',' fl oat ') ,
          ('BOD ',' ll oat '),
          (' NITRATE  N NI'RITE  N ' .  ' II  oat  ') ,
          (' FECAL_C0LIFORN ',' float ') ,
          ('T0TAL_E0LIF0RM ',          ' ) )
```

Now the column TOTAL COLIFORM is not required so we will drop this column.

[14] : d f=dl . d rap I ' TOTAL COL I F 0Rf•I ')

Now we have to check all the rDwn for nulla values and remove all the rDws whicn contain any null value in it.

So for applying a SOL query we first have tD register it has a virtual temporary table and then we will issue SOL query.

We are doing this because it is impDrta nt to perform data cleaning as it will make our mDdel to wDrk with better prediction and accuracy.

I [15]' df.createOrReplaceTempViewC"d'

I h [lfi] : d f c lean = s pa rk. s qt (' ' ' Seve ct ' f i ori d f s‹ I \dJe I e r ENR zs not u\ I and D0 z s not nu T
and pH zs not nul I and BBD z s not iJ u\ T a nd c0NDUCTI\'I Y i s n at n ri I \
and NITRATE N NITP I TE N 1 s n at nu T\ and FECAL C DL I F0éffl is n0t iJ u\ I ' ' ')

[l7] : d I . s fi o I S1

STATION C0DE	L0CATI ONS	STATE	TEMP	DO	pH	CONDUCTIVITY	BOD	NITRATE_N_NITRITE_N	FECAL_COLIFORM
13 I2	GODAVARI AT AYAK. . .	NAHARASHTRA	29.2	[illegible] 4	8 . L	735. 0	3. 4	2 . 0	3. 0 I
2 L77	GODAVARI RIVER NE . . .	NAHARASHTRA	24. 3	6. 0	8 . 0	270 . 0] 3 . 1	2 . 0	7 2. 0 I
2 L82	GODAVARI RIVER AT. . .	NAHARASHTRA	23. B	3.5 7 . B		35S . 0] 4 . 2	**9.0**	b9. 0 I
2 L79	GODAVARI RIVER AT. .	NAHARASHTRA	24. B	[illegible] 7 . B		37 1 . 0] 5 . 6	**3.55**	90 . 0 I
2 L83	GODAVARI RIVER AT. .	NAHARASHTRA	25.7	[illegible] 7 . 9		294 . 0] 3 . 2	2 . 69	45. 0 I

only showing top 5 rows

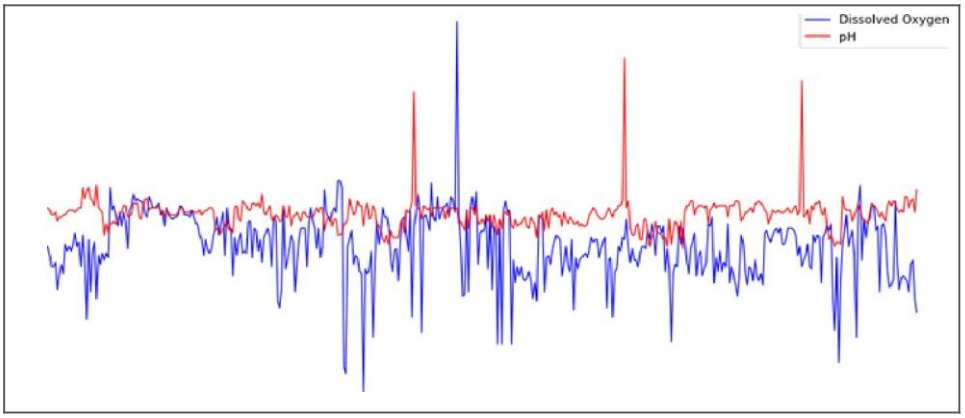
EDA - Exploratory data analysts

In [18]: df clean . c rea te0rRepl aceTempView (" a f « I " }

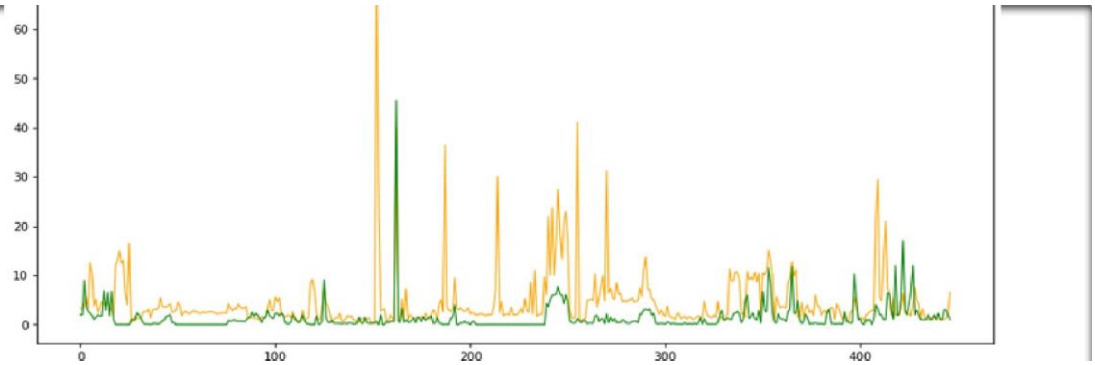
In [20]: do = s p *a rd* . s *Q\ ! ' S e L< c** D0 ' -oz ':1f sql ")
do = do . rdd . nap (lambda row : row . DO) . cot I e c t (}
ph = s p *a rd* . s *Q\ ' S e L ec* cH- -oz ':1f sql ")
ph = ph . rdd . nap (lambda row : row . pH) . cot I e c t (}
bod = spa rk . sql (" S.•t e 'z t B0? *I i- on* d' sql ')
bod = bad . rdd . nap f **anbda** rDw **row.** BOD) . c a \ \ ec t ( )
nn = s p *a rd* . s *Q\ ! ' S e L< c** f. I Th A.TE f. IJ1 TRI T E IJ f ron df sq". ')
nn = nn . *rdd* . rn*a* p *i* \ *as bd a* row : row . NITRATE N NITRITE N) . c of\.ec t I)

Data visualization

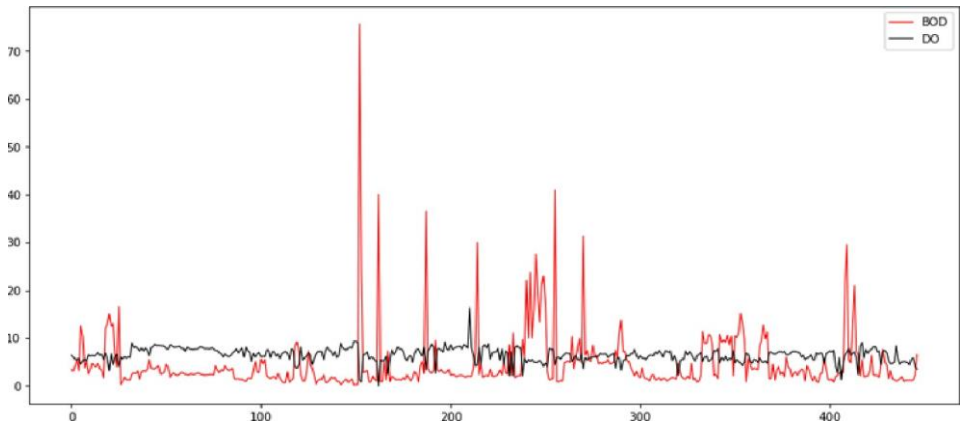
2 a { 2 l] fig . ax = pit . subp\.ots f nun=None , figs 1ze= (L4, fi) , dpi=80 , f acec a \.or= ' ' , ed g e c of a r= L:)
size=teri (dD)
ax . plot I rarige f 0. size}, do, co\.or= ' L't'.ie , anlnat ed=True , I i newidth=I , I abeI= ' Di s • a". 'zed C x y;e n ')
ax . plot (rarige f 0. size}, ph, co\.or= ' i-ed ' , animated=T rue , I inewidth=L , I abet = *H ')
legend=ax . legend (}



In [22]:
```
fig , ax2 = pit . s ubp \ ot s ( num=None , 1'igsize= ( L4, G } , dp i=8 0, I ac ec a\ a r—— +' ' ,  edgeco\.or= ' k' }
ax2. plot ( i ange \'8 , size , bad , cDIo r=' a i z ng e ,  a nina t ed=T rue ,  I ineu'idt h=I ,  \.abe\.= ' BOX ' }
ax2. plot ( i ange \'8 , size , nn , co\o r—— 'g re.°n ' anina t ed=T r ue , \ i ne\ai d th=L , label= ' 'Jf. ' }
legend=ax2. legend I )
```

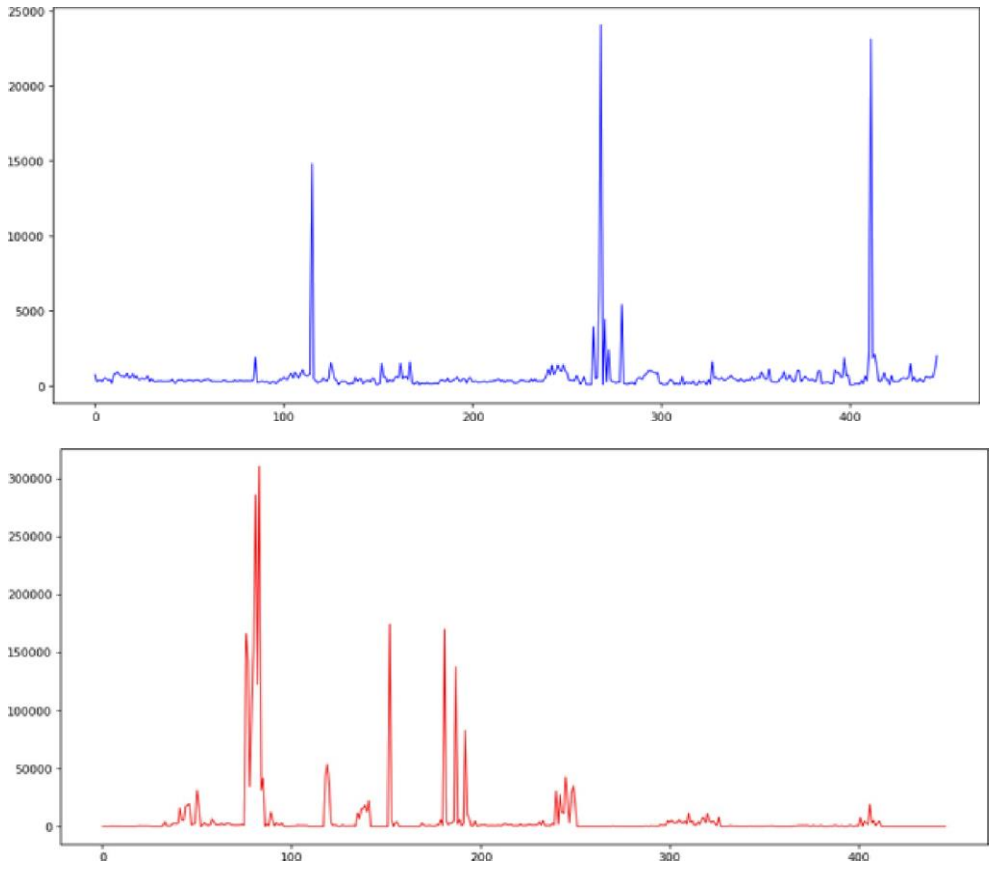


In [23]:
```
fig . ax3 = p\ t . s u bptots num=No ne , f igs i ze= ( L4, G } , dpi=80.  ! a c ec a l a r——  v" ,  edgeco l a r= ' k ' }
ax3. plot ( i an g e ( , size } .  bad ,  c DT o r=  ' reo ' ,  an i ri at ed=True .  Si newid t h=I ,  \abe\= ' B OD ' }
ax3. plot ( i an g e ( , size I , do , c a l a r—.- l a c k ' a nina ted=T rue , \ in evi d th=E , label= ' DC ' }
tegend=ax3. l e gen d f )
```



```
In [ 2d I :   I  com  = spar1‹. sql -I 5e\ec t CO/@UCTIVITY f rd d f sql " )
              2  com = con. rdd. eap (£ aaAda rou : re. CB@U£ TIYITY ). collect ( )
                 fee = spar-k. sql -f Setec I FECAL COLIF0£'¥I Iron d f sql " )
              4  fee  =  fee. rdd. eap (£ aaAda rou :  re.  FECAL_C6LI F0f¥'¥) . collect ( }

In [ 23I :    I  Iig. ax  = pit . subptot s (nu•=Me . i igsize= t l4. 61 . g  i we .  Iacecofor= 'u' .  ed9ecofor= 'k' 1
              2  an . plot ( r6ozf ( B, size) , con ,  color= 'b I ue ' , animted=7rue , 1ineuidth=i)
              2  fig,arZ = pltsuplots(num=6bne,figsize=(Lt,6}, Opi=B0, facecoIor-w', edgecolor='k')
              J  axe.plot(rangel8.size). 7ec. color='r*d'. aninated=7me. Ii  idtE=l)
```

0u t [25 j : [zsat plott ib . lines. Li ne2D at 8x7fede75ae6ae• I



Feature engineering

In [26]:
```
d f=d f clean . t oPand a s ( I
d f . dtypes
```

Out[26].
```
STATION CODE              obj ect
L0CATIONS                 ob j e ct
STATE                     object
TEMP                     flQat32
D0                       float32
pH                       float32
C0NDUCTIVI^Y             float32
BOD                      float32
NITRATE NNITRITE N       float32
FETAL COLIFORM           float32
dtype: object
```

In [27]:
```
sta rt-0
end-94B
station-df . i\ oc [ s ta rt : end , 0]
\ocation=d f . ioc [st a rt: end , I ]
stat e=dT il a c [ sta rt : end , 2]
do= df.iloc [start:end ,4].astype(np.float641
value-0
ph - d I . hoc [ start : end , 3]
co - d I . i\oc [ start : end , 6 I . as type ( np . I \o at 64
boo = dT . il a c [ sta rt : end , 7 ] . a stype ( np. II oat64)
na= d f itoc [sta rt : end , 8 ] . a stype ( n p . I I oa t64)
I c=d f . i\oc [ 2 : end , 9 ] . a stype ( np . ft Dat64)
```

In [28]:
```
df=pd.concat([station,location,state,do,ph,co,bod,na,fc],axis=1)
df. columns = ['station','location','state','do','ph','co','bod','na','fc']
```

the Water Quality Index is calculated by aggregating the ouality with the weight linearly:

WQI = ∑ (qn x Wn)

where qn =Ouality rating for the nth Water quality Wn= unit weight for the nth oaramelers

In [29]:
```
df I iJ EH ] =d I . ph . apply { \ anibda x: { 10'3 if ( 8. 5>=x>=7)
                                                el se (TO if (6. 3>=x>=8 . 5 ) a r (3 . 3—=x>=6 . 8 )
                                                    else ( 6'3 if ( B B>=x>=8 . 6 ) a r ( b . 8>=x>=6 . T )
                                                        et se ( '3 1+ ( 3>=x>=0 . 8 ) a r ( 6 .7>=x>=6. 3 )
                                                            e \ se C )))))
```

In [30]:
```
df I inc a ] =d I . do . apply ( \ ambda x: ( I'00 If ( x>=6 )
                                                el se ( a0 if (s>=x>=5. 1)
                                                    else ( 6'3 if ( 5>=x>=4. )
                                                        et se ( ='3 1+ (4>=x>=fi )
                                                            else 0)))))
```

In [5I]:
```
df I iJ c a ] =d T . Ie . apply ( \ ambda x : ( I'30 If ( ñ>=x>='3)
                                                el se (â0 if (50>=x>=3 )
                                                    else ( 6'3 if ( 5C0—=x>=5'0)
                                                        et se ( ='3 1+ ( 100'30>=x>=-i OC}
                                                            else 0)))))
```

In [*2 :
```
df I iJ »‹Io ' ] =df . bad . apply ( I aribda x : ( L'30 If ?>=x>='3 )
                                                el se ( SI? if (3>=x>=*)
                                                    else ( 6'3 if ( 8C>=x>=s )
                                                        et se ( I'3 1+ ( 1. 5>=x>=8'3)
                                                            e \ se C )))))
```

In
```
df I wee ] =d I . co . apply ( \ ambda x: ( I'00 If * 5>=x>=0 )
                                                else(o if (15†>=x>= }
                                                    else(60 iC (2‹S>:x>=15e)
                                                        else( i* %té>=x>=225)
                                                            else011) l
```

In [5*]:
```
df I iJ la ] =d I . na . apply ( \ ambda x: ( I'30 If ( 2 0>=x>=0 )
                                                el se ( SO if {50>=x>=2 0 )
                                                    else ( 6'3 if ( ICI?—=x>=5'0)
                                                        et se ( ='3 i* ( 2C 0>=x>= '3C ) et
                                                            se C } ) } ) }
```

Mow we appl y the formula of wqi by first multiplying all the qu ality rating with its weig ht and then summed all the values

In [36]:
```
d f [ n,'th ' ] =d f . npH  *  0. 165
d f [ ' v,'d a ' ] =d f . ndo  * 0. 28 1
d I [ ' u bd a ' ] =o f . n bdo *  0 . 2 34
d I [ ' sec ' ] =d I . nec' D 009

d f [ '\-. c D ' ] =d I . nco * 0 . 28 I
d f [ '\-.‹  ' ] =d I . wph+d I wdo•df . wbdD*d f . wee d . una*d f . wco
```

	station	location	stie do ph	oo bod ua k npH . nboo nec uua wph wdo wbdo weo wna woo
0	1342	GO DAVARI AT JAYAKWADI M AURMAGABAD MAHARAS A	MAHARASHTRA 6.^ 8 I 735.0 3.4 2.00 NaN DO	80 0 1 0D I 6.5 28.10 18.72 D.DO 2.8 0 00
1		GODAVARI RIVER NLAH SOMESHWAR TEMPLE.	MAHARASHTRA 60 80 27D.0 31 200 HaN	80 40 1 OD 16.5 28.10 18.72 D.36 2.8 0 00
2	B2	GODAVARI RIVER AT SAIKHEDA.	MAHARASHTRA 5 5 7 8 355 0 4 2 9 00 59 0 DO	80 0 10D 165 224B 1872 DDO 2.8 i686
3		GODAVARI RIVER AT HANUMAN GHAT, NASHIK CITY.	MAHARASHTRA 5.5 7 8 371.0 5.6 3.55 90 0 DO	80 0 10D 16.5 22M8 1£72 D.D0 2.8 1686
4	B3	GODAVARI RIVER AT NANDUR-MADMESHWAR DAM.	MAHARASHTRA 5.7 7 4 294.0 3.2 2.69 45 0 DO	80 40 1 OD 16.5 22.4B 18.72 D.36 2.B 224B
442	29s0	GAPE- SAGAR LAKE, DUNGARPUR, RAJASTHAN	R JAB+HAN 4.^ 8 1 5 3B.0 1 2 1.00 5 0 DO	100 0 1 0D 16.5 1 6.86 23.40 D.DO 2.8 2810
443	2941	LAKE JAISAMAND, SALUMBER, UDAIPUR, POINT	RAJA0*HAN 5.6 8 4 5 91.0 1.1 3.00 4 0 DO	T00 0 1 OD 1 6.5 22.4B 23.40 D.DO 2.8 28 10
444	2942	LAKE JAISAMAND, SALUMBER, UDAIPUR, POINT	RAJA3*HAN 5 8 8 5 58B 0 1 2 3 00 4 0 DO	y00 0 1 0D 16 5 22 4B 23 40 D DO 28 10
445	2953	LODHA TALAB, BANSWARA-DUNGARPUR ROAD, BANSWAR...	RAA5*HAE 4.1 7 9 11330 2.3 2.00 4D0	IOO 0 10D 16.5 0.00 23.40 DDO 2.8 2248

Now we classify the water on the basis of their water quality index

In 137]
```
d I [ ' at Qty ] =d f . wqi . apply I lacbda x I ' Ex c tent ' if I 25>=x>=0)
                                                  else ( ' GoDd  if  I  50>=x>=26)
                                                       else ( PDo r If ( 75>=x>=3I \
                                                            else ( ' '.'e i y Poo r if I 1D0>=x >=76 1
                                                                 else ' Un s u itab\ e ) ) ) ) )
```

In [38]
```
s pa rk  dl  =  sq I Cont ext  . c reat eDataF rane (df I
```

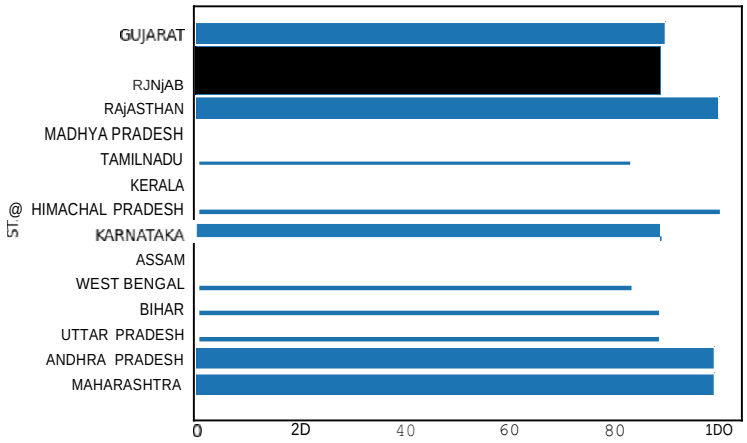
In [39]:
```
s pa rk dl . show ( )
```

|station| location| state} do ph| ‹• I ood
^•I +‹I•• I ndo | mo | n bdo | nec | nna mph >do u odo web I
wna | wco sq i | qua I it y

1312 | GODAVARI AT AYAK. . . | HAHARASH RA | 6. 40o09009 3367432 | 8. 100000381 69727 | 7J3 .0 Is .4coooo s5JO74316
2.0 | NaN | LOO | 100 | 0 | B0 1 0 | L00 | I6. 3 | 28.1 | 18.720oOoO00000002 0.0 | 2.800
0000000000003 | 0 0 | 66. 12 | Poo r
2177 | GODAVARI RIDER NE . . . | MAHARASH+RA | 6. 0 8. o 1270. 0 13.09999990 6323684 I
2. 0 NaN | U0 | ION | 0 | BO 1 40 | U0 | I6. 5 28 . 1] 18 . 720 000000000002 0.36 | 2.800
0000000000003 | 0 0 | 66. 4B | Poo r
2182 | GODAVARI RIDER AT. . . | HAHARASH+RA | 5.5 | 7 . 8000@0190734863 | 3^\- ^ 1 4 99999809265137
9. 0 1 59. 0 | L00 I BO 1 60 | B0 1 0 | L00 | T6. 5 | 22. 480000000000004 | 18.720oOoO00000002 I 0.0|2.800
0000000003\16.860000000000003| 77.36|Very Poor|
2179|G0DAVARI RIDER AT. . . | MAHARA9H*RA| 5.5|7.800000190734863|371.*I *9999990463236B| 3.Z499999
32316284| 90.01 100 80 601 80| OIT OO 16.5122.480000O0o0o0p |1B.720000000000002 0.0|
2.8000000000000003 16.860000000000003| 77.36|Very Poor|
2183|G0DAVARI RIDER AT. . . | MAHARASH*RA|5.699999809265137|7.900000095367432|294 "1 *000000^76B3716| 2.6900000
37220459| 45.01100 80 BOI 80| 401 T00 lc.5122.48000000000000p|1B.720000000000002 0.36|
2.8000000000000003|22.480000000000004|
| 2181|GODAVARI RIVER AT...| MAHARASHTRA|
52316284|131.0|100| 60| 60| 60| 0|100|
2.8000000000000003|16.860000000000003|

83.34|Very Poor|
4.5| 7.5|513.0|12.600000381469727| 2.2999999
16.5|16.860000000000003|14.040000000000001| 0.0|
67.06| Poor|

```
In [40]: . spa rk_df . c reate0 rRepla ceTempview ( " d sql" )

In [41]: StatB = spa rk . sql " Select s tate f rom d s ql " I
         State = State . rdd . map ( t anbda row : row . stateI . col tec t ( )

In [42]: . ñqi = spa rx . s qt ( " Sel ect wqi f rom d f sqI " )
         Wqi = Woi.rdd.map(lambda row    row.wqi).collect()

In [43]: . pit . ba rh ( State, ¥/qz )

         pit . xtabet ( "WOI " )
         pit . ytabet ( " STATES " )

         plt.show()
```



Modeling

Now we agply machine learning algorithms la predict the data.

Linear Regression Model

First we convey the required data to predict WQI into vector form by using VectorAssembler.

Then we normalize our oata by using Normalized.

```
In [44]: from pyspark. nil.feature 1nport vectorAssembler
         from pyspark. nil.feature 1nport Norxialzzer

         vectorAssembter = vectorAssemD\erIinputCol s=["npH","ndo","nbdo","nec","nna","nco" , outputCol="features")
         norealizer = Normal1zer(inputcoT=' features ',outputCol="*eatures 'norm"I
```

Applying Linear Regresion model

```
In [ ^3 ]: f rom pys pa rk. nil . reg res sion lnporl Lznea rReg ressnon
           l r = Line a rReg res s ion ( +eatu resC a\ =" 'eatu res no rn " , la bet Cal = "wqi " , maxlte r=10, regPa ram=0 . 3, elasticNetPa rani=0 . 2)
```

Now we streamline processes into

```
In [46]: from pyspark.ml import Pipeline

In l^7]: pipeline = Pipeline \ stages= [vecto rAs senbte r , no rna tizer, \ r] )
```

Now we Gplil the data into train and test dala

```
In [48]: t rain_da ta , test_data=s pa rk_df . randonspl zt ( [0. 8, 0.2] )
```

Making predictions using the model

In [51]:
```
predi ctions - model.t ransTo rm\t      data)
```

I [52]'
```
predictions.select('vqi','p›eJictzon'l.show(l
```

```
                wqi|        predictzon

|82.03999999999999|B2.1238308668°088|
              82.4| Bl.9274096394131 B|
             66.12|67.34565604943359
              66.12|67.3456b604943359|
              82.4|Bl.92740963941318|
              77.72|77.78171638043395|
   77.36000000000001|77.89313076930208
             66.1Z|67.3456b604943359|
|82.03999999999999|B2.1238308668p08  B|
             66.12|67.34565604943359
              66.12|67.3456b604943359|
|82.03999999999999|B2.1238308668°088|
 82.03999999999999|B2.12383086684088
 |82.03999999999999|B2.1238308668q088|
              66.12|67.3456b604943359|
              77.36|78.5041637745495^|
             82.98| B3.11194223770477
              82.4|Bl.92740963941318|
              77.9|77.9108613083°056
             93.28|91.66703037750938|

Only showing top 20 rows
```

Now we check the perforzriance of our model

In [53]:
```
a=nodet . stages [ 2] . s unna ry . r2
```

Out[53]: 0.9753564789532392

Logistic Regression Model

Here we are creating a logi stic regres sion model because we dank have to oredict a continuous value

In [54]:
```
Trom pysoark.ml.feature imoort Siringlndexer
```

As our quality column contains values in string format so first we indexed tnem using Stringlndezer

Then data is converted which are required to predict water quality into vector form by using VectorAssembler

Then we normalize our data by using Normalized.

In [55]:
```
zndexer = StringIndexer(inputCol="quality",outputCo\=' \a›e°.)
vectDrAssemD\er2 = vectorAsseirbier inputCol s=[ iJpH", nda","rıDda","roec ,"iJna ,'ncD',"\ i "], outputco\="feat'.ires2 )
```

Applying Logistic RegressiDn MDdel

In [56]:
```
from pyspark.ml.classification import LogisticRegression
lor = LogisticRegression(featuresCol="features norm2",labelCol="label",maxIter=10)
```

In [57]:
```
pi peti ne2 - Pipe\ine stages=[indexer, vectarAssenbler2, normasiz er2, to r] )
```

Now we split the data into train and test data

I lv|58]'
```
\ rain data , test_data= s pa rk d f  rand omsplit | 0 . 8, 0.2 ] )
```

Fitting the data into the model

In [60]:
```
model2 = pioeline2.fitltrain datal
```

Making predictions using the mOdel

In [611:
```
D red ic tions 2 = modet2 . I ransfo rm ( t rain data )
```

In [6Z1:
```
D redic tions 2 . select ( ' label " , " p redict ion " ) . s how ( I
```

```
1 a bet | p re d ict to n
0 .i3            0.0
0 .i3            0.0
0 .i3            0.0
O.fl |           O.O|
```

ont y showing I aa 2 g r ows

Now we check the performance of our model

In [64]:
```
f ron pyspa rk . md . eva I ua I ton lnport Nu111 c las sClas sT f 1 c at lonEva I ua I a r
. eva\ = Nult ictass Cha s s1 f i ca l i on Eva I ua I o r ( ) . set Net r'icN ame ' ac cu racy' I . set L abet Co l ( ' la bed ' ) . set Pred ic 11onCa I ( ' p redi
o=evaI . eva I uat e (p red 1 c t tonsZ )
```

Out[64]: 0.9972144846796658

As the quality column is in string fomat so we convert our predicted data which are in numbers to their real string values and compare tnem with the actual data

In [65]:
```
names = I "be ry Poo r " , " Poo r" , " Good " , " U ns uita bye " , " Excel bent " ]
```

In [68]:
```
p red 1c II ons 2 . c rea t eOr Rept ac eTemp\it em ( " p i ed ct ion s 2 sql ")
```

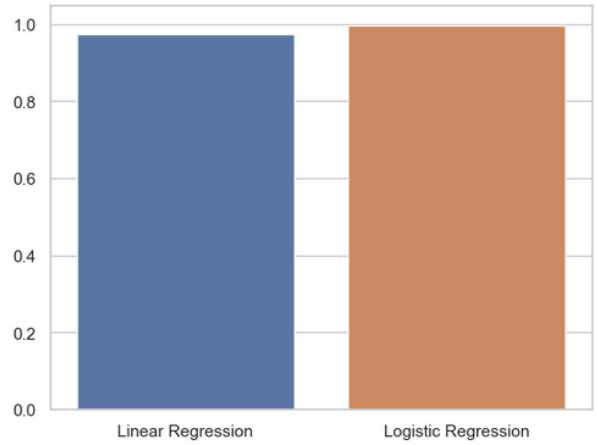
In [671:
```
p red = spark. sql ( 'Seles I p edict ion I on predic ns2 sq I' )
. p red = p red . rdd . nap ( \ ambda row : Int ( row . p red ic t i on ) ) . cot Dec t ( )
qua = spark. sql ( " Select quad it y Iron p red ie tzons2 sq I " )
- qua = qua . rdd . nap ( \anbda row : row. qua I i I y) . cod \ ect ( )
```

In [69]:
```
for x In range (100 T:
    print ( " P red ie ted: " , naoes [ pred I x ] ] , " Ac tual : " , qua I x I )
```

```
Pred1c ted : Ve ry Poo r Act uat : Very Poo r
Pred1c ted : Ve ry P oo r Act uat : Very Pa a r
Pred be ted : PDo r Actual : PDo r
Pred1c ted : Poo r Actual : Poo r
Pred be ted : PDo r Actual : PDo r
Pred1c ted : Poo r Actual : Poo r
Pred be ted : ve ry Poo r Act uat : very Poo r
Pred1c ted : Ve ry P oo r Act uat : Very Pa a r
Pred be ted : ve ry Poo r Act uat : very Poo r
Pred1c ted : Poo r Actual : Poo r
Pred be ted : ve ry Poo r Act uat : very Poo r
Pred1c ted : Ve ry P oo r Act uat : Very Pa a r
Pred ?c ted : Poo r Actual : Poo r
Pred1c ted : Ve ry P oo r Act uat : Very Pa a r
Pred ?c ted : Ve ry P oo r Act uat : Very Pa a r
Pred1c ted : Ve ry P oo r Act uat : Very Pa a r
Pred1c ted : Ve ry P oo r Act uat : Very Pa a r
Pred1c ted : Ve ry Poo r Act uat : Very Poo r
Pred1c ted : Ve ry P oo r Act uat : Very Pa a r
Pred1c ted : Ve ry Poo r Act uat : Very Poo r
Pred1c ted : Ve ry P oo r Act uat : Very Pa a r
Pred be ted : ve ry Poo r Act uat : very Poo r
Pred1c ted : Ve ry P oo r Act uat : Very Pa a r
Pred be ted : ve ry Poo r Act uat : very Poo r
Pred1c ted : Ve ry P oo r Act uat : Very Pa a r
```

In [70]:

```
sns.set_theme(style='whitegrid')
scores = pd.DataFrame(data=[[a,b]],columns=['Linear Regression', 'Logistic Regression'])
sns.barplot(data=scores);
```



RESULT:

The program is executed successfully.