ANALYSIS OF WATER RESOURCES AND WATER MANAGEMENT FOR CHENNAI CITY

CSE3020 – DATA VISUALIZATION

J – COMPONENT

FINAL REPORT

SUBMITTED TO

PROF. SIVANESAN S.

SLOT: L13 + L14

SUBMITTED BY

NAME	REGISTRATION NUMBER
MOHD UMAR	18BCE0196
PRATHAM SHARMA	18BCE0343

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

DECLARATION

We hereby declare that the project titled **"Analysis of Water Resources and Water Management for Chennai City"** submitted by us, for the award of the degree of Bachelor of Technology in Computer Science and Engineering to VIT is a record of bonafide work carried out by us under the supervision of Prof. Sivanesan S.

We further declare that the work reported in this project has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place: Vellore

Date: 05/11/2020

Mohd Umar (18BCE0196)

Pratham Sharma (18BCE0343)

CERTIFICATE

This is to certify that the project titled **"Analysis of Water Resources and Water Management for Chennai City"** submitted by **Mohd Umar (18BCE0196), Pratham Sharma (18BCE0343),** SCOPE, VIT, for the award of the degree of Bachelor of Technology in Computer Science, is a record of bonafide work carried out by him under my supervision during the period, 13.07.2020 to 05.11.2020, as per the VIT code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The report fulfils the requirements and regulations of the University and in my opinion meets the necessary standards for submission.

Place: Vellore

Date: 05/11/2020

ACKNOWLEDGEMENTS

We would first like to acknowledge the help and guidance provided by Prof. Sivansesan S. sir during the implementation of this project. Without his encouragement we would not have been able to work on this topic.

We would also like to acknowledge the University Management and our respective School Deans for giving us an opportunity to carry out our studies at the University

Mohd Umar (18BCE0196)

Pratham Sharma (18BCE0343)

CONTENTS

Analysis of Water Resources and Water Management for Chennai City

ABSTRACT

Chennai, formerly known as Madras, is the capital of Tamil Nadu, a state on the eastern coast of India. Located on the Coromandel Coast off the Bay of Bengal, the city of Chennai is one of the largest cultural, economic and educational centres of south India.

It is also one among the 100 cities to be developed as a smart city under the Smart Cities Mission launched by PM Narendra Modi in 2015. However, in spite of being a large metropolitan city and being developed as a smart city, Chennai faces huge problems in terms of water resource availability.

Last year (2019), Chennai faced an acute water shortage. On 19 June 2019, Chennai city officials declared that "Day Zero", or the day when almost no water is left, had been reached, as all the four main reservoirs supplying water to the city had run dry. Families and households had to rely on alternative water sources such as distant, unreliable public water pumps, and costly private water tankers.

Through this project, we will try to

- Visualize the water requirement/usage of the city of Chennai and surrounding areas using historical data.
- Identify whether the current available water resources will be able to meet the water needs till the subsequent monsoon.
- The severity of the Chennai water crisis of 2019.

Through this project, we aim to analyse the historical data about water resources available for Chennai city. With the help of visualizations and statistical modelling techniques we will be able to predict whether the currently available water resources will be able to meet the water requirements of the residents of Chennai and surrounding areas.

PROBLEM ANALYSIS

OBJECTIVES

The objective of the project is to visualize and analyse available historical data about water resource and water management in Chennai, Tamil Nadu.

The main objectives of this project are as follows:

1. Analyse individual water resources for Chennai.
2. Analyse water requirements of Chennai city through analysis of historical data about water resources for Chennai.
3. Perform data visualization over data to uncover trends and patterns in rainfall and water usage.
   For example, charts such as line charts will be plotted to visualize and analyse time-based data.
4. Visualize the water requirements of the city over the years.
   Various charts, such as bar charts and line charts, will be used to visualize the water usage of the city over the years.
5. Analyse the water crisis faced by Chennai city in 2019.
   Based on the various charts plotted using available data, we will try to find out important patterns and trends that led to the crisis of 2019, by comparing it with previous years.
6. Perform statistical modelling and forecasting to predict water requirements in the future.
   We will use an ARIMA model to analyse time-based data. This will be important for forecasting the water requirements of the city in the near future.
7. Use time-series analysis algorithms to perform analysis.
   Time series analysis will be used for forecasting purposes. Time-series analysis models are widely used for non-stationary data, such as stock prices, weather conditions as well as economic parameters such as unemployment or GDP.

OUTCOMES

The outcomes of this project include:

1. Analysis of historical data available about water resources for Chennai.
   The data will be analysed using graphs and charts to find patterns in the data. Using this pattern, various management techniques will be suggested.

2. Time series analysis of data for forecasting.
   Time series analysis based on the data will be helpful to forecast future water requirements. By forecasting the water need in advance, we can prepare in advance to tackle any problems in water availability with proper water resource management.
3. Developing statistical model.
   We will develop a statistical model specific to water resources data for analysing it effectively.
4. Suggest better management techniques.
   By uncovering patterns in the water resources dataset by performing in depth analysis, we will be able to suggest certain measures that can be adopted by the governing bodies to better manage available water resources.

Only by performing such an analysis will we be able to highlight the important patterns and trends in the dataset. This will be very crucial in formulating better water management policies so that tragedies such as the 2019 water crisis can be avoided in the future.

CONSTRAINTS

There are a few constraints that are needed to be overcome in order to successfully implement this project. These are:

1. Adding time series analysis and forecasting
   In addition to the existing data analysis methods, time series analysis and forecasting should also be included to perform better analysis of the data.
   Since the data available for water resources is time-based data about rainfall and water usage, time series analysis will provide more information about the data.

2. Hardware requirements for performing time-series analysis
   In order to successfully perform time-series analysis on the datasets available, hardware with exceptional computing capability is required as the datasets available contains nearly 6000 unique records for time-based data. High computing capability is required to model such data effectively.

RELEVANCE

The city of Chennai is the sixth-most populous city and fourth-most populous urban agglomeration in India. The city together with the adjoining regions constitutes the Chennai Metropolitan Area, which is the 36th-largest urban area by population in the world.

It has a population of more than 1 crore residents (as per 2019 census) that occupy nearly 30 lakh households and attracts 45 percent of health tourists visiting India, and 30 to 40 percent of domestic health tourists. As such, it is termed "India's health capital".

Being such an important metropolitan city, it is important that Chennai city manages its water resources in a sustainable manner to accommodate for the needs of its residents.

The city is entirely dependent on ground water resources to meet its water requirements. The major sources of fresh water for the city consist of 4 reservoirs in the city.

These four reservoirs are:

1. **Red Hills**: Also known as Puzhal Lake is one of the 2 rain-fed reservoirs from where water is drawn for supply to Chennai city.
2. **Cholavaram Aeri**: Located in Ponneri taluk of Thiruvallur district, Cholavaram Lake is one of the main sources of fresh water for Chennai city.
3. **Poondi Lake**: Poondi Lake is the reservoir across Kotralai river in Thiruvallur District.
4. **Chembarambakkam Lake**: It is a lake located in Chennai, Tamil Nadu, India, about 25 km from Chennai. It is the second rain-fed reservoir along with Puzhal Lake from where water is drawn for supply to Chennai city.

These four reservoirs have a combined water capacity of 11,057 million cubic feet or nearly 313100 million litres.

Apart from the 4 reservoirs, other sources of water are:

- Desalination plants at Nemelli and Minjur
- Aquifers in Neyveli, Minjur and Panchetty
- Water from Cauvery river in Veeranam Lake

In spite of having abundant water resources Chennai faced a huge water crisis in 2019. Two years of deficient monsoon rainfall, particularly in late 2017 and throughout much of 2018 had led to this crisis.

As we already know, Chennai faced a huge water crisis last year which affected a large number of households. Keeping such an event in mind, it is important that water resources be managed properly to handle extreme situations of crisis effectively in the future.

EXISTING SYSTEM

SAMPLE APPLICATION

Current methodologies for analysis of water resources dataset apply standard analysis methods such as

- Using graphs for discovering trends
- Analysing historical data

Such methods are good for analysing and visualizing patterns from available dataset and are widely used for analysis.

However, they do not provide a method to perform predictions and forecasting based on trends and patterns in historical data. Hence, they do not provide much significance in real-life.

ISSUES

There are a few points where the existing methodologies lack and have scope for improvement. These are:

1. Adding time series analysis and forecasting.
   In addition to the existing data analysis methods, time series analysis and forecasting can also be included to perform better analysis of the data.
   Since the data available for water resources is time-based data about rainfall and water usage, time series analysis will provide more information about the data.
2. Performing statistical modelling.
   Statistical modelling will be used to develop model for forecasting purposes. This methodology is not adopted in existing analysis techniques.

CHALLENGES

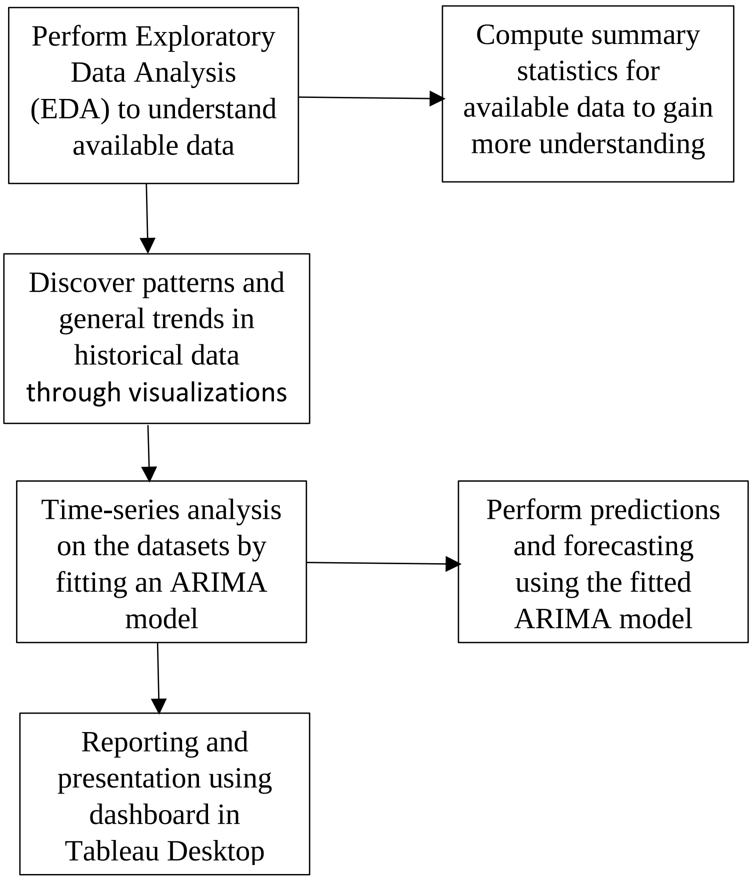
Performing time-series analysis requires significant computing capabilities as the available historical data contains a lot of records with time-based data. Utilising such computing capabilities is a major challenge in analysing available historical data through time-series analysis and forecasting.

PROPOSED SYSTEM

In this project, we will be working with historical data about the Chennai water reservoir levels. We will make use of algorithms to predict whether the available water resources will be able to meet the water need of the Chennai and nearby people till next monsoon or not, starting with simple algorithms like time series forecasting.

CONCEPTUAL MODEL

The conceptual model of our project can be described appropriately with the help of the following flowchart:



Flowchart: Conceptual model for project

DATA ANALYSIS

The dataset used for this project is available at the link given below:

https://www.kaggle.com/sudalairajkumar/chennai-water-management

This dataset has details about the water availability in the four main reservoirs over the last 15 years

- Poondi Lake
- Cholavaram Aeri
- Red Hills
- Chembarambakkam Lake

The data about water availability is available on a daily basis.

The dataset also includes data about the rainfall levels for Chennai over the last 15 years. This data will be used to perform required forecasting and analysis.

Water availability in the four main reservoirs:

About this file

Water level in different reservoir

Date	# POONDI	# CHOLAVARAM	# REDHILLS	# CHEMBARAMBAK...
Date of observation	Rainfall in Poondi Reservoir	Rainfall in Cholavaram Reservoir	Rainfall in Redhills Reservoir	Rainfall in Chembarambakkam Reservoir
5836 unique values				
01-01-2004	3.90	0.00	268.00	0.00
02-01-2004	3.90	0.00	268.00	0.00
03-01-2004	3.90	0.00	267.00	0.00
04-01-2004	3.90	0.00	267.00	0.00
05-01-2004	3.80	0.00	267.00	0.00
06-01-2004	3.80	0.00	266.00	0.00
07-01-2004	3.80	0.00	266.00	0.00
08-01-2004	3.70	0.00	265.00	0.00
09-01-2004	3.70	0.00	264.00	0.00

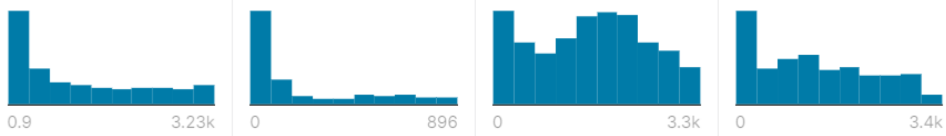


Table: chennai_reservoir_level.csv

As we can see, this dataset about the volume of water available in the four main reservoirs near Chennai consists of daily data about water levels for all 4 reservoirs for the past 15 years (2004 - 2019).

The dataset consists of 5836 unique data records.

Rainfall level for different reservoirs:

About this file

Rainfall level in different reservoirs

Date	POONDI	CHOLAVARAM	REDHILLS	CHEMBARAMBAK...
Date of observation	Rainfall in Poondi Reservoir	Rainfall in Cholavaram Reservoir	Rainfall in Redhills Reservoir	Rainfall in Chembarambakkam Reservoir
5836 unique values	0 – 300	0 – 293	0 – 320	0 – 475
01-01-2004	0	0	0	0
02-01-2004	0	0	0	0
03-01-2004	0	0	0	0
04-01-2004	0	0	0	0
05-01-2004	0	0	0	0
06-01-2004	0	0	0	0
07-01-2004	0	0	0	0
08-01-2004	0	0	0	0

Table: chennai_reservoir_rainfall.csv

The second dataset about the daily rainfall levels for the four major reservoirs is described above. It contains data about the daily amount rainfall over the 4 reservoirs that are used to obtain water for Chennai city.

This dataset also has 5836 records of time-based data.

Both the datasets above will be used to perform analysis of water resource management and requirement for Chennai city. Time series forecasting can be applied since the data available to us is time-based.

ALGORITHM

In this project, we have implemented time-series analysis of reservoir water level and rainfall data using an ARIMA model.

Time series analysis and forecasting is the use of a model to predict future values based on previously observed values. It comprises of methods for analysing time series data in order to extract meaningful statistics and other characteristics of the data.

In time series forecasting, data is stored and recorded at a specific interval of time and then that data is analysed to forecast future. It models the dataset based on the following components of time-based data:

1. Trend
2. Seasonality
3. Cyclicity
4. Irregularity

In order to perform statistical modelling on the datasets to perform time-series analysis, the algorithm that we have implemented can be described as follows:

- Add column ‘Total’ to reservoir water level dataset for total daily water level in all four reservoirs.
- Remove columns containing daily water level for individual reservoirs.
- Sort the records of the dataset by the ‘Date’ column.
- After sorting all records, set the ‘Date’ column as index for the dataframe.
- Fit an ARIMA model for the reservoir water level dataset;
    - Select parameters for the ARIMA model as:
        - Seasonality (p) = 1
        - Trend (d) = 0
        - Noise (q) = 0
    - The required ARIMA model is available in the Python package statsmodels.api
    - Initialize ARIMA model with appropriate parameters.
    - Fit ARIMA model for the reservoir water level dataset.
- Plot diagnostics for the fitted ARIMA model to examine the accuracy.

Hence, we have fitted an ARIMA model to analyse the data based on time-series analysis. We have also performed forecasting using the fitted ARIMA model to forecast the total reservoir water level and total rainfall level.

IMPACT ANALYSIS

Our project has various real-life impacts. Analysis of water resource data will be very useful in order to formulate better water management policies. By discovering interesting patterns in the data through data visualizations, various suggestions will be made to conserve and manage water resources effectively.

Along with the data visualizations and analysis, time series forecasting will also help to forecast future water requirements and prepare to manage water resources in advance. This will be very helpful to combat situations of water crisis in the future.

BENEFITS

By adopting such data analysis methods, we will be able to answer important questions about water resource usage and management. This will be useful in the management of water in the future and avoiding crisis.

This method will also help many other states and countries for better water management.

HARDWARE REQUIREMENTS

The hardware requirements for this project are:

1. Minimum of 4GB RAM will be required to efficiently perform analysis over the dataset and perform statistical modelling.
2. An i5 or higher processor is required to perform time-series analysis on the datasets.
3. Enough storage space (minimum of 1GB) is required to store datasets, visualizations and reporting dashboards for the project.

SOFTWARE REQUIREMENTS

PLATFORM AND TOOLS

For the purpose of this project, we will be requiring the following software and tools:

1. **Anaconda**: The Anaconda navigator contains Spyder, which is an integrated development environment (IDE) for Python. We will be using this to perform data analysis and visualization using Python.
2. **Python 3.7.4**: Python 3.7.4 is a stable release of Python which we will be using for our analysis. It consists of all the required libraries that we will need for analysis.
3. **Matplotlib**: Matplotlib is a data visualization library for Python. We will use matplotlib to plot the various charts and graph required for analysis.
4. **Tableau Desktop**: We will be using Tableau Desktop, a data visualization and business intelligence tool, to make a data visualization dashboard for review presentations.
5. **Tableau Prep**: Tableau Prep will be used along with Tableau Desktop to prepare the dataset for data visualization.

Image: Software and Tools used

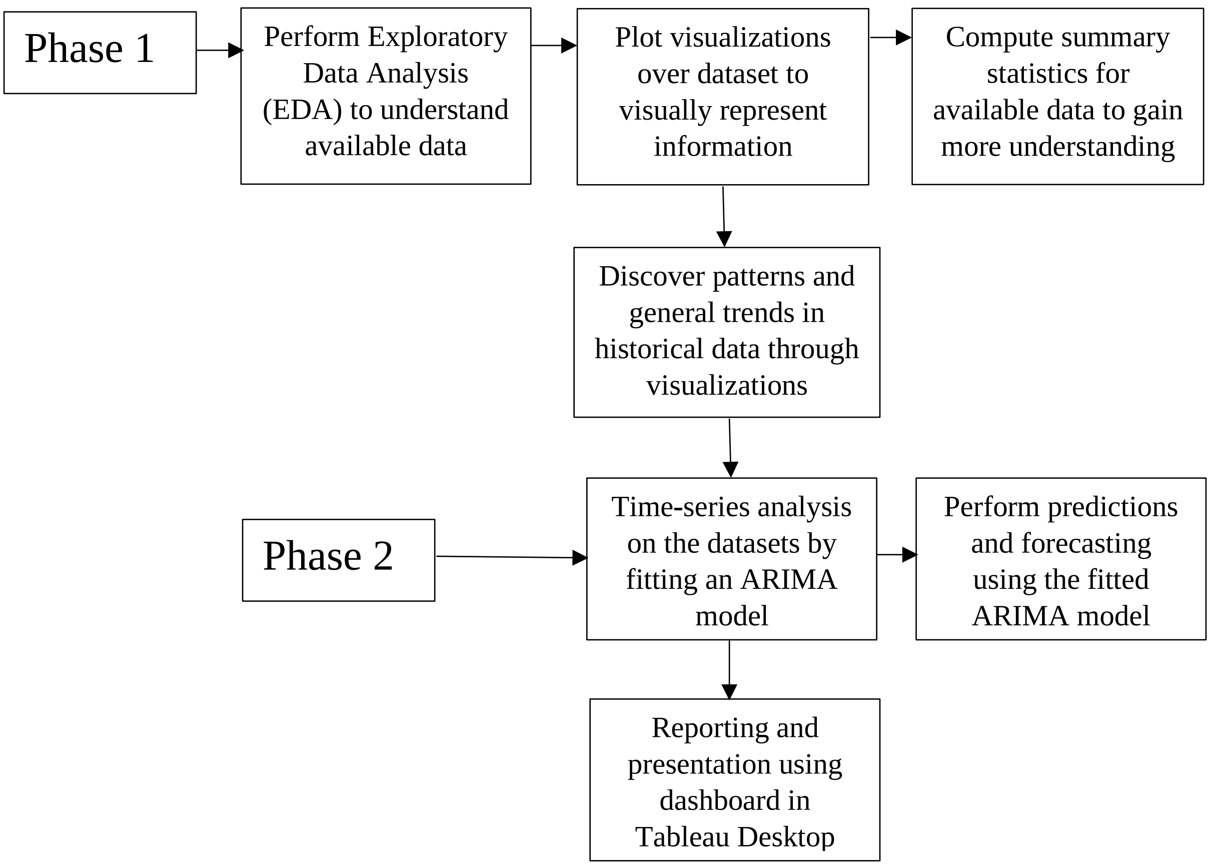
IMPLEMENTATION

MODULE DIAGRAM

The implementation of this project has been divided into the following 2 phases:

1. Exploratory Data Analysis (EDA): In this phase of the project we have performed preliminary exploration of the datasets used in the project. Through this preliminary exploration, we gained a better understanding of the available data.
   We performed exploratory data analysis of the project with the help of 2 main techniques – summary statistics and visualizations. After gaining a better understanding of the available datasets, we started the next phase of the project which is described next.
2. Time-series Analysis: during this phase, we performed time-series analysis and forecasting for reservoir water level and rainfall datasets. We fitted and ARIMA model for the time-based data and performed forecasting with the fitted ARIMA model.

Finally, we have consolidated all results of the project into a dashboard designed using Tableau Desktop for final presentation and reporting.



ORIGINAL CODE

Phase 1: Exploratory Data Analysis (EDA)

In [1]:
```
# Importing required libraries

import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
import warnings
warnings.filterwarnings("ignore")
```

Code Snippet: Importing required libraries

In [2]:
```
# Loading reservoir levels and rainfall datasets

lvl = pd.read_csv("chennai_reservoir_levels.csv")
rain = pd.read_csv("chennai_reservoir_rainfall.csv")
```

Code Snippet: Loading reservoir water level and rainfall datasets

In [3]:
```
# Displaying a few records

lvl.head()
```

Out[3]:

	Date	POONDI	CHOLAVARAM	REDHILLS	CHEMBARAMBAKKAM
0	01-01-2004	3.9	0.0	268.0	0.0
1	02-01-2004	3.9	0.0	268.0	0.0
2	03-01-2004	3.9	0.0	267.0	0.0
3	04-01-2004	3.9	0.0	267.0	0.0
4	05-01-2004	3.8	0.0	267.0	0.0

Code Snippet: Displaying few records for reservoir water level dataset

In [4]:
```
# Displaying some records

rain.head()
```

Out[4]:

	Date	POONDI	CHOLAVARAM	REDHILLS	CHEMBARAMBAKKAM
0	01-01-2004	0.0	0.0	0.0	0.0
1	02-01-2004	0.0	0.0	0.0	0.0
2	03-01-2004	0.0	0.0	0.0	0.0
3	04-01-2004	0.0	0.0	0.0	0.0
4	05-01-2004	0.0	0.0	0.0	0.0

Code Snippet: Displaying few records for rainfall dataset

In [5]:
```
# Displaying summary statistics for water level dataset

lvl.describe()
```

Out[5]:

	POONDI	CHOLAVARAM	REDHILLS	CHEMBARAMBAKKAM
count	5836.000000	5836.000000	5836.000000	5836.000000
mean	1116.271270	234.265965	1533.680166	1285.674289
std	1020.316805	272.289990	906.251292	964.355749
min	0.900000	0.000000	0.000000	0.000000
25%	200.225000	18.375000	791.000000	408.000000
50%	775.500000	91.000000	1596.000000	1190.000000
75%	1975.000000	445.000000	2212.250000	2047.000000
max	3231.000000	896.000000	3300.000000	3396.000000

Code Snippet: Summary statistics for reservoir water level dataset

In [6]:
```
# Displaying summary statistics for rainfall dataset

rain.describe()
```

Out[6]:

	POONDI	CHOLAVARAM	REDHILLS	CHEMBARAMBAKKAM
count	5836.000000	5836.000000	5836.000000	5836.000000
mean	3.464402	3.758773	3.828992	4.012688
std	13.065897	14.267783	15.011809	16.251079
min	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.000000	0.000000	0.000000
50%	0.000000	0.000000	0.000000	0.000000
75%	0.000000	0.000000	0.000000	0.000000
max	300.000000	293.000000	320.000000	475.000000

Code Snippet: Summary statistics for rainfall dataset

In [7]:
```
# Plot for water level in Poondi reservoir

fig = px.line(lvl, x='Date', y='POONDI', title='Water level for Poondi reservoir')
fig.show();
```

Code Snippet: Plot for Poondi reservoir water level

In [8]:
```
# Plot for rainfall level in Poondi reservoir

fig = px.line(rain, x='Date', y='POONDI', title='Rainfall level for Poondi reservoir')
fig.show();
```

Code Snippet: Plot for rainfall level in Poondi reservoir

In [9]:
```
# Plot for water level in Cholavaram reservoir

fig = px.line(lvl, x='Date', y='CHOLAVARAM', title='Water level for Cholavaram reservoir')
fig.show();
```

Code Snippet: Plot for Cholavaram reservoir water level

```
In [10]:
# Plot for rainfall level in Cholavaram reservoir

fig = px.line(rain, x='Date', y='CHOLAVARAM', title='Rainfall level for Cholavaram reservoir')
fig.show();
```

Code Snippet: Plot for rainfall level in Cholavaram reservoir

```
In [11]:
# Plot for water level in Red Hills reservoir

fig = px.line(lvl, x='Date', y='REDHILLS', title='Water level for Red Hills reservoir')
fig.show();
```

Code Snippet: Plot for Red Hills reservoir water level

```
In [12]:
# Plot for rainfall level in Red Hills reservoir

fig = px.line(rain, x='Date', y='REDHILLS', title='Rainfall level for Red Hills reservoir')
fig.show();
```

Code Snippet: Plot for rainfall level in Red Hills reservoir

```
In [13]:
# Plot for water level in Chembaranbakkam reservoir

fig = px.line(lvl, x='Date', y='CHEMBARAMBAKKAM', title='Water level for Chembaranbakkam reservoir')
fig.show();
```

Code Snippet: Plot for Chembarambakkam reservoir water level

```
In [14]:
# Plot for Rainfall level in Chembaranbakkam reservoir

fig = px.line(rain, x='Date', y='CHEMBARAMBAKKAM', title='Rainfall level for Chembaranbakkam reservoir')
fig.show();
```

Code Snippet: Plot for rainfall level in Chembarambakkam reservoir

```
In [15]:
rain.Date = pd.to_datetime(rain.Date)
rain.set_index('Date', inplace=True)
```

```
In [16]:
rain.total = rain.POONDI + rain.CHOLAVARAM + rain.REDHILLS + rain.CHEMBARAMBAKKAM
rain.total.plot(figsize=(20,10), linewidth=3, fontsize=15)
plt.xlabel('Year', fontsize=15)
plt.ylabel('Rainfall', fontsize=15);
```

Code Snippet: Total rainfall level for all 4 reservoirs

```
In [17]:
# Correlation between reservoir water levels

corr = lvl.corr()
corr.style.background_gradient(cmap='PuBu')
```

Code Snippet: Correlation matrix between reservoir water levels

```
In [18]:
# Correlation between rainfall levels

corr = rain.corr()
corr.style.background_gradient(cmap='PuBu')
```

Code Snippet: Correlation matrix between reservoir rainfall levels

Phase 2: Time-series Analysis

```
In [1]:
# Import Required Libraries

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
plt.style.use('fivethirtyeight')
import itertools
import statsmodels.api as sm

import matplotlib
matplotlib.rcParams['axes.labelsize'] = 14
matplotlib.rcParams['xtick.labelsize'] = 12
matplotlib.rcParams['ytick.labelsize'] = 12
matplotlib.rcParams['text.color'] = 'k'

import warnings
warnings.filterwarnings("ignore")
```

Code Snippet: Importing required libraries

```
In [2]:
# Loading reservoir levels and rainfall dataset

lvl = pd.read_csv("chennai_reservoir_levels.csv")
rain = pd.read_csv("chennai_reservoir_rainfall.csv")
```

Code Snippet: Loading reservoir water level and rainfall datasets

```
In [3]:
# Creating new column 'Total' for total water level and total rainfall

lvl['Total'] = lvl.POONDI + lvl.CHOLAVARAM + lvl.REDHILLS + lvl.CHEMBARAMBAKKAM
rain['Total'] = rain.POONDI + rain.CHOLAVARAM + rain.REDHILLS + rain.CHEMBARAMBAKKAM
```

```
In [4]:
# Removing other columns

cols = ['POONDI', 'CHOLAVARAM', 'REDHILLS', 'CHEMBARAMBAKKAM']

lvl.drop(cols, axis=1, inplace=True)
rain.drop(cols, axis=1, inplace=True)
```

```
In [5]:
# Sorting records by date and setting 'Date' as index

lvl = lvl.sort_values('Date')
lvl = lvl.set_index('Date')

rain = rain.sort_values('Date')
rain = rain.set_index('Date')
```

Code Snippet: Data pre-processing

In [6]:

```python
# ARIMA models are denoted with the notation ARIMA(p, d, q)
# p, d and q account for seasonality, trend and noise in data

p = d = q = range(0, 2)
pdq = list(itertools.product(p, d, q))
seasonal_pdq = [(x[0], x[1], x[2], 12) for x in list(itertools.product(p, d, q))]

print('Examples of parameter combinations for Seasonal ARIMA...')
print('SARIMAX: {} x {}'.format(pdq[1], seasonal_pdq[1]))
print('SARIMAX: {} x {}'.format(pdq[1], seasonal_pdq[2]))
print('SARIMAX: {} x {}'.format(pdq[2], seasonal_pdq[3]))
print('SARIMAX: {} x {}'.format(pdq[2], seasonal_pdq[4]))
```

Code Snippet: Time-series modelling with ARIMA model

In [7]:

```python
# Fitting ARIMA model and displaying summary

lvl_model = sm.tsa.statespace.SARIMAX(lvl,
                                      order=(1, 0, 0),
                                      seasonal_order=(1, 1, 1, 12),
                                      enforce_stationarity=False,
                                      enforce_invertibility=False)
lvl_results = lvl_model.fit()
print(lvl_results.summary().tables[1])
```

```
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.5630      0.011     49.430      0.000       0.541       0.585
ar.S.L12       0.0505      0.015      3.391      0.001       0.021       0.080
ma.S.L12      -0.9908      0.003   -395.685      0.000      -0.996      -0.986
sigma2      5.899e+06   6.86e-10   8.61e+15      0.000     5.9e+06     5.9e+06
==============================================================================
```

In [8]:

```python
# Plotting diagnostics for ARIMA model for water level data

lvl_results.plot_diagnostics(figsize=(16, 8))
plt.show()
```

Code Snippet: Fitting ARIMA model for water level data and plotting diagnostics

In [9]:

```python
# Fitting ARIMA model and displaying summary

rain_model = sm.tsa.statespace.SARIMAX(rain,
                                       order=(1, 0, 1),
                                       seasonal_order=(0, 1, 1, 12),
                                       enforce_stationarity=False,
                                       enforce_invertibility=False)
rain_results = rain_model.fit()
print(rain_results.summary().tables[1])
```

```
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.9460      0.010     92.949      0.000       0.926       0.966
ma.L1         -0.8887      0.013    -66.965      0.000      -0.915      -0.863
ma.S.L12      -0.9927      0.002   -509.038      0.000      -0.997      -0.989
sigma2      2727.5178      8.693    313.751      0.000    2710.479    2744.556
==============================================================================
```

In [10]:

```python
# Plotting diagnostics

rain_results.plot_diagnostics(figsize=(16, 8))
plt.show()
```

Code Snippet: Fitting ARIMA model for reservoir rainfall level data and plotting diagnostics

RESULT AND ANALYSIS

OUTPUT

Phase 1: Exploratory Data Analysis

Reservoir water and rainfall level for Poondi reservoir

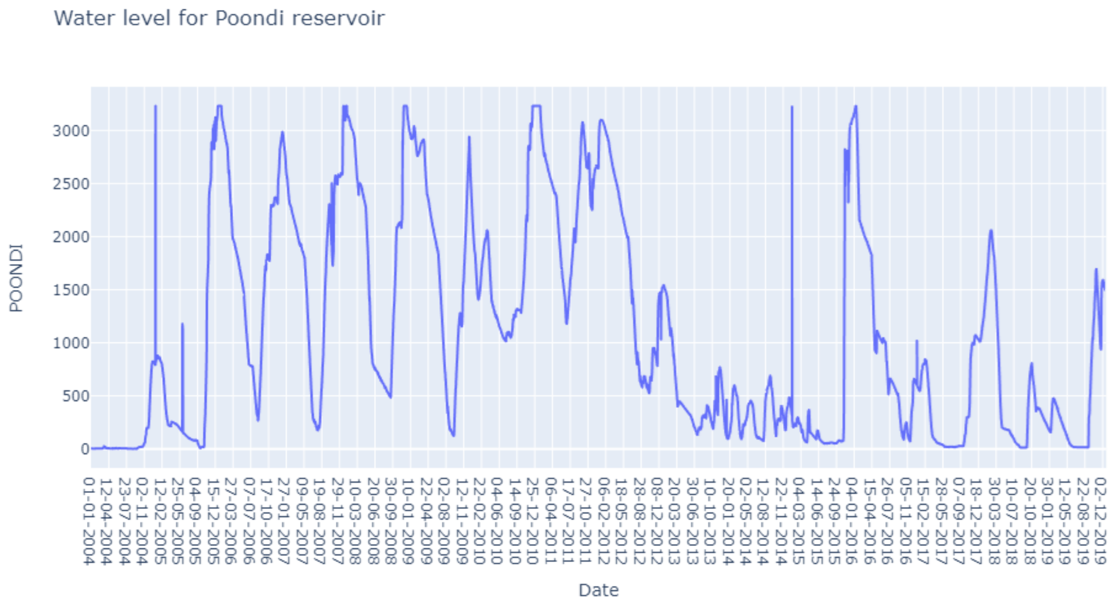


Figure: Reservoir Water Level for Poondi reservoir, 2004-2019

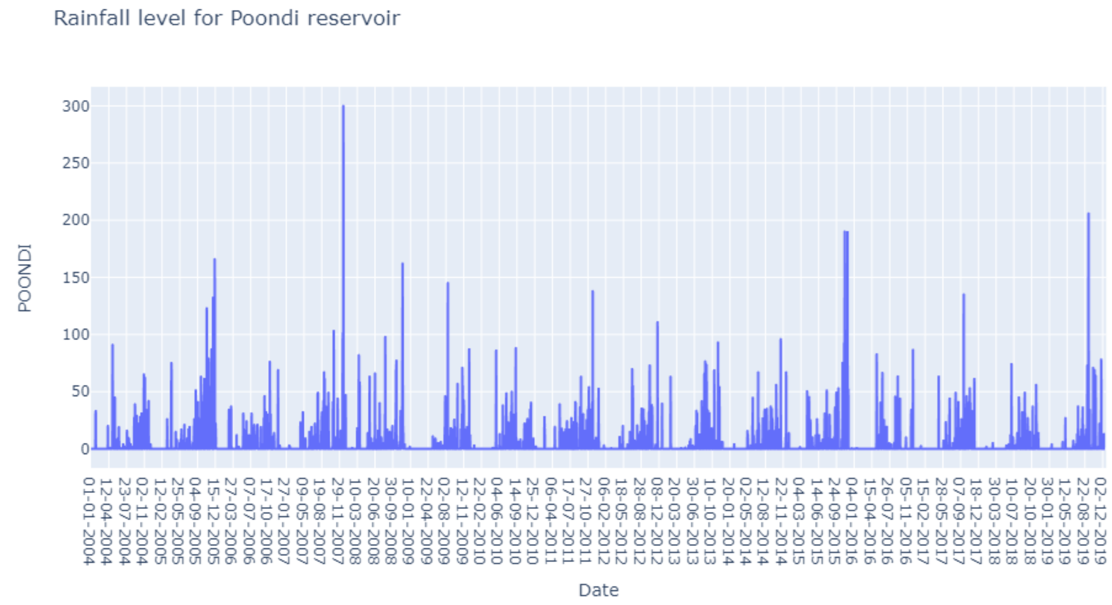


Figure: Rainfall Level for Poondi reservoir, 2004-2019

Reservoir water and rainfall level for Cholavaram reservoir

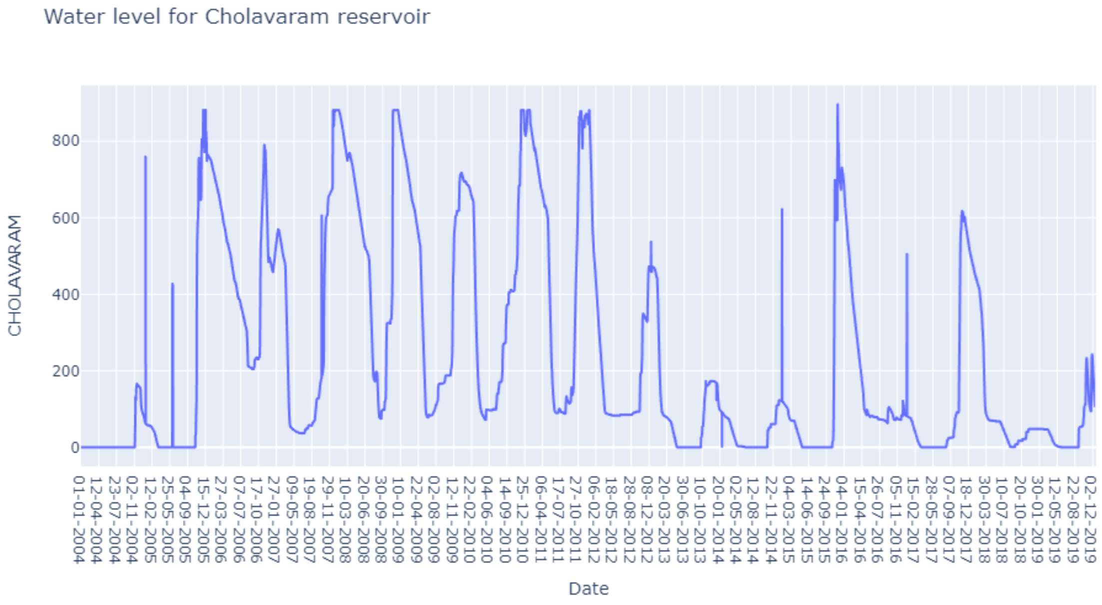


Figure: Reservoir Water Level for Cholavaram reservoir, 2004-2019

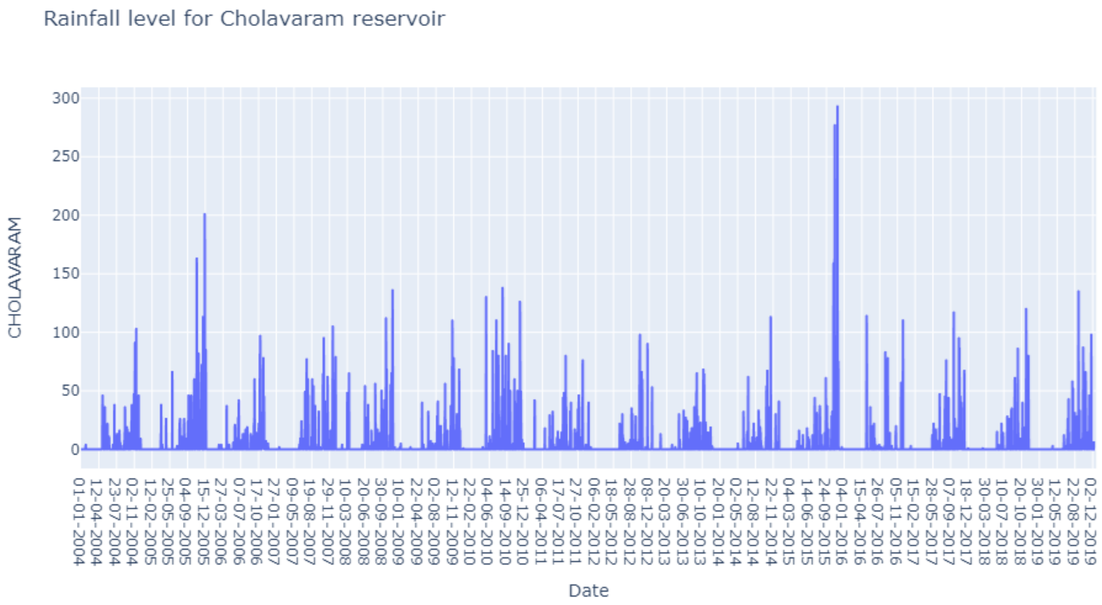


Figure: Rainfall Level for Cholavaram reservoir, 2004-2019

Reservoir water and rainfall level for Red Hills reservoir

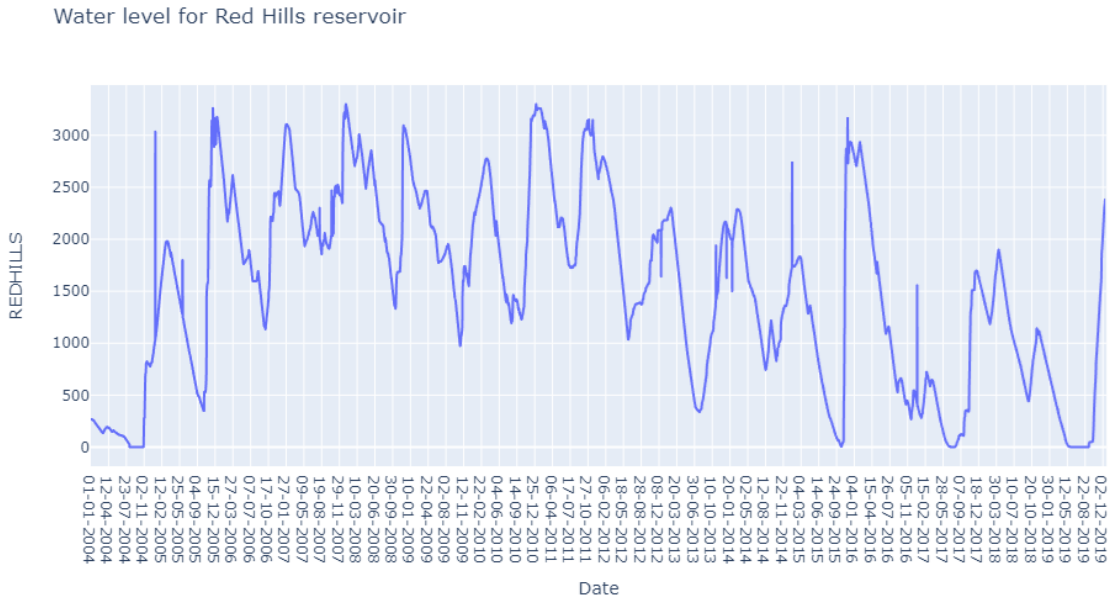


Figure: Reservoir Water Level for Red Hills reservoir, 2004-2019

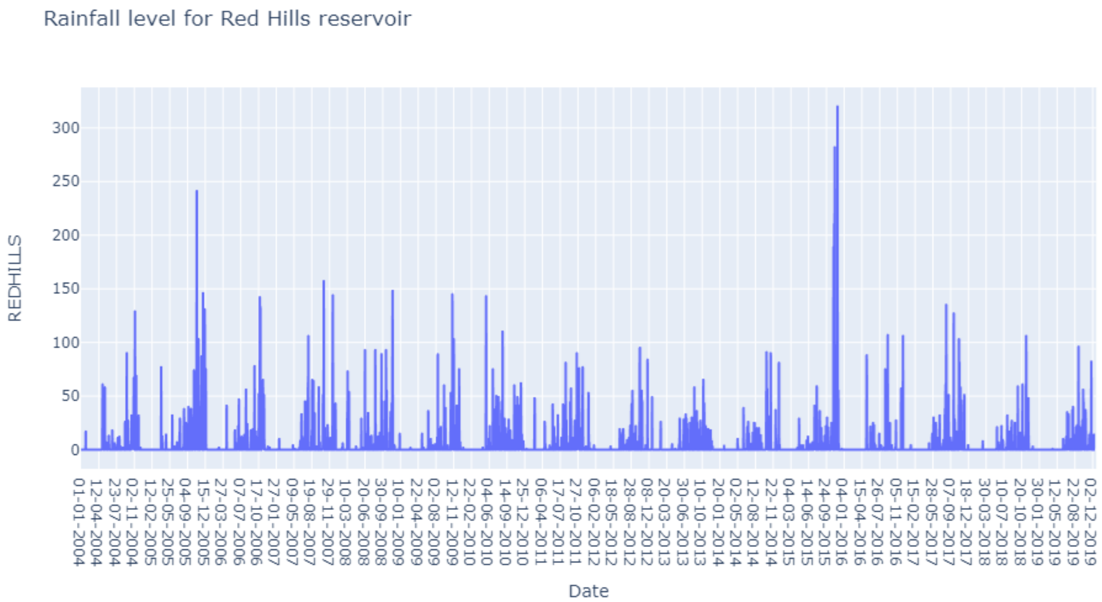


Figure: Rainfall Level for Red Hills reservoir, 2004-2019

Reservoir water and rainfall level for Chembarambakkam reservoir

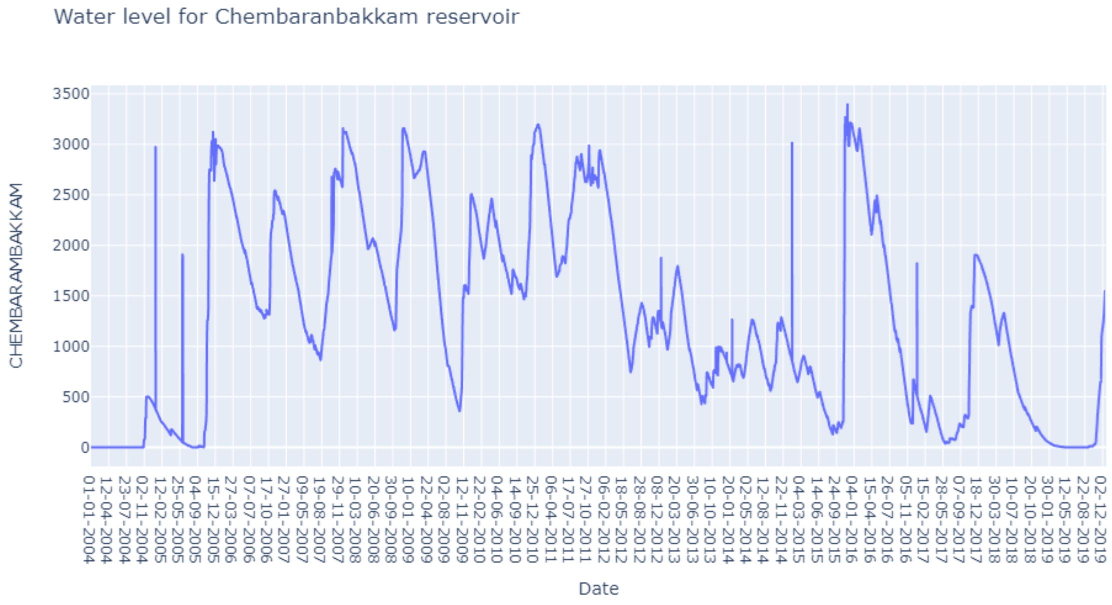


Figure: Reservoir Water Level for Chembarambakkam reservoir, 2004-2019

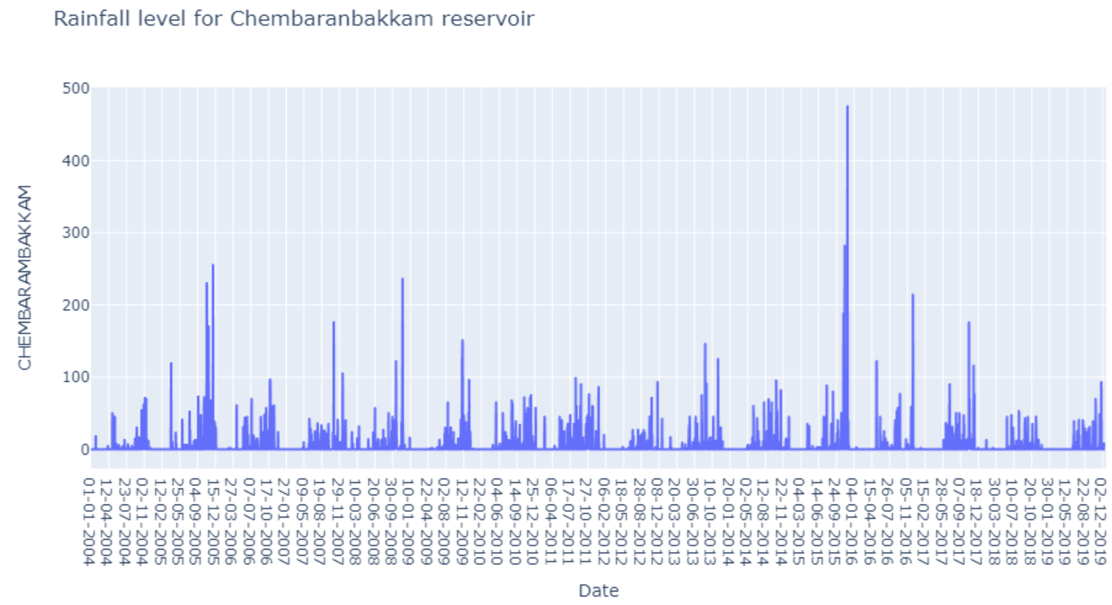


Figure: Rainfall Level for Chembarambakkam reservoir, 2004-2019

Total rainfall level for all 4 resrevoirs

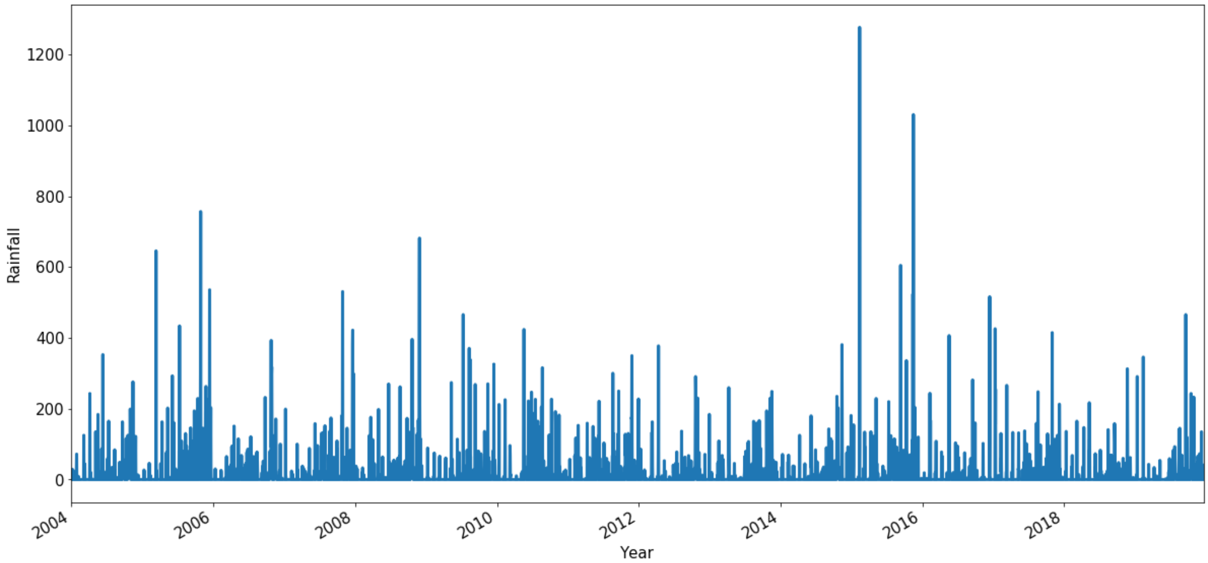


Figure: Total Rainfall Level, 2004-2019

Correlation matrix for reservoir water levels

	POONDI	CHOLAVARAM	REDHILLS	CHEMBARAMBAKKAM
POONDI	1	0.833647	0.804117	0.879381
CHOLAVARAM	0.833647	1	0.757776	0.812085
REDHILLS	0.804117	0.757776	1	0.867192
CHEMBARAMBAKKAM	0.879381	0.812085	0.867192	1

Figure: Correlation Heatmap for Reservoir Water Levels

Correlation matrix for reservoir rainfall levels

	POONDI	CHOLAVARAM	REDHILLS	CHEMBARAMBAKKAM
POONDI	1	0.716779	0.708773	0.673569
CHOLAVARAM	0.716779	1	0.901169	0.741132
REDHILLS	0.708773	0.901169	1	0.780106
CHEMBARAMBAKKAM	0.673569	0.741132	0.780106	1

Figure: Correlation Heatmap for Reservoir Rainfall Levels

Phase 2: Time-series Analysis with ARIMA Model

Diagnostics for ARIMA model for reservoir water level data

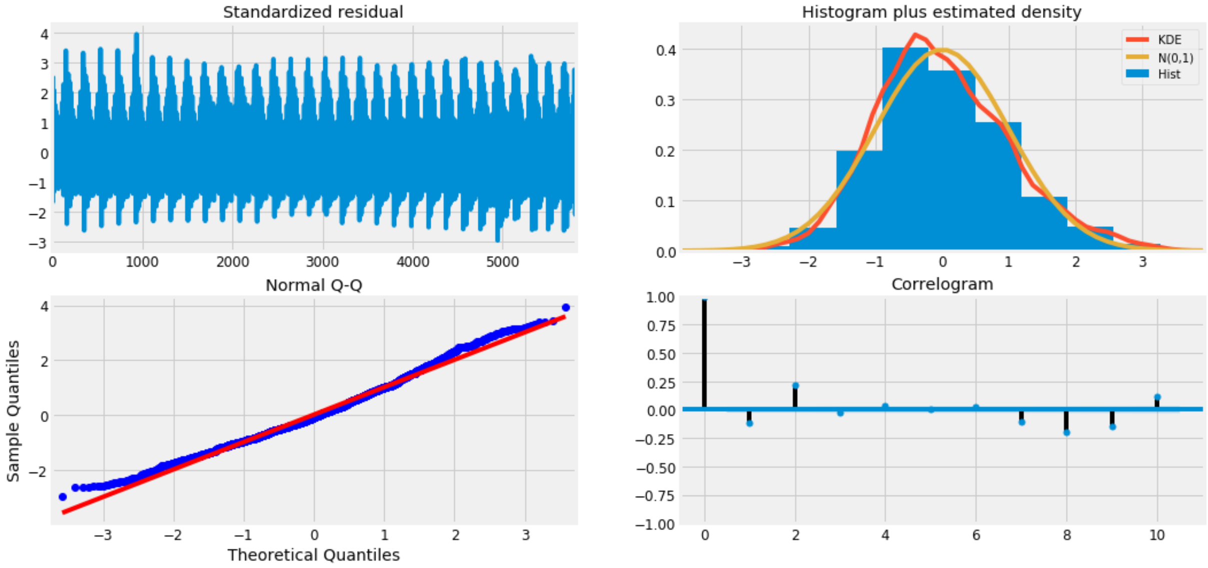


Figure: Diagnostics for reservoir water level ARIMA model

Diagnostics for ARIMA model for reservoir rainfall level data

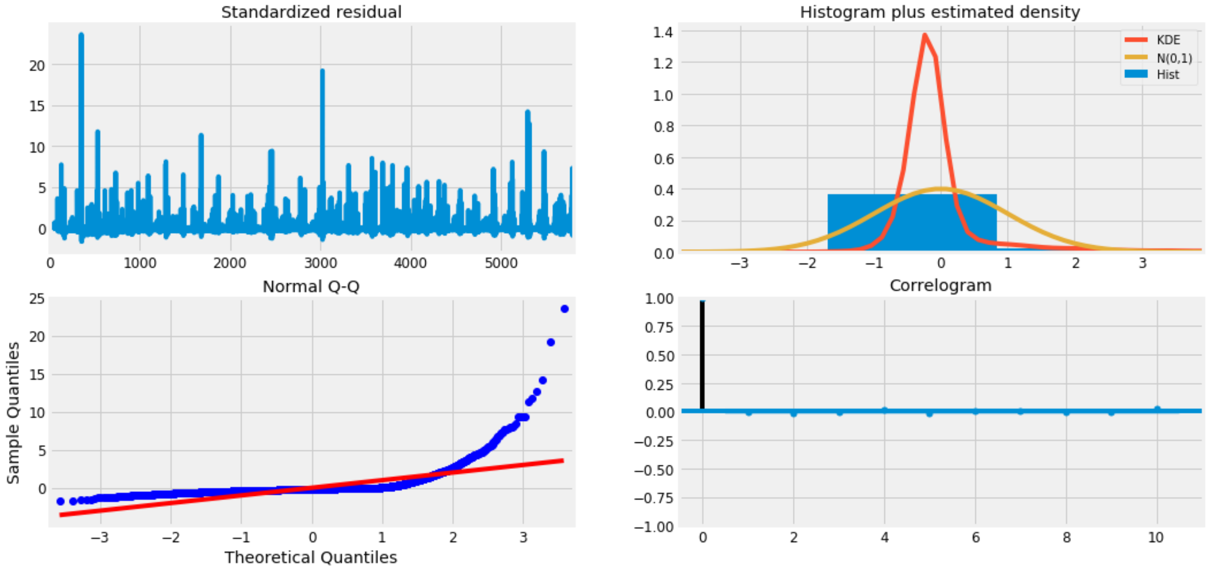


Figure: Diagnostics for reservoir rainfall level ARIMA model

RESULTS

Using the visualizations above, we can conclude the following about the reservoir water level and rainfall level datasets:

- Water level data has a seasonality that can be seen in the plots.
- Although this trend has been decreasing, in 2019 only Poondi reservoir had some water left. All other 3 reservoirs were running dry in 2019, during the water crisis in Chennai.
- During the month of December 2014, we can observe a sudden growth in water level. This is unnatural and the water must have been supplied to Chennai city from some other region.
- Downfall in water level is significant during the months from February to July in 2019.
- Evidence of water shortage can be noticed in all 4 reservoirs during the years 2017 as well as 2018.
- The dry period generally starts in the month of October and lasts until the month of December.
- Total water availability is very low in 2019. Periodic cycle is disturbed in this phase.
- A significant downfall in total water level has been observed since 2015 and keeps decreasing till 2019.
- Signs of heavy correlation is observed in the correlation matrices for reservoir water level and rainfall level. Seasonality can be explained using this evidence.
- The forecasted values for reservoir rainfall level indicate that during the next few years the rainfall is expected to be average.
- Also, forecasted values for reservoir water level show that the total water level across all reservoirs is expected to be lower than the historical average.
- These observations indicate the likelihood of a water crisis occurring in Chennai. Therefore, it may be required that water is conserved to deal with a likely crisis situation.

Hence, through our exploratory data analysis and time-series analysis, we can conclude that there is a seasonality and general trend in water and rainfall level for reservoirs. With the help of forecasts, we can say that water availability may be reduced due to average amount of rainfall and below average reservoir water levels in the coming few years. Therefore, authorities and citizens should prepare adequately to deal with a probable water crisis in the near future.

TABLEAU DASHBOARD

Dashboard created with Tableau Desktop

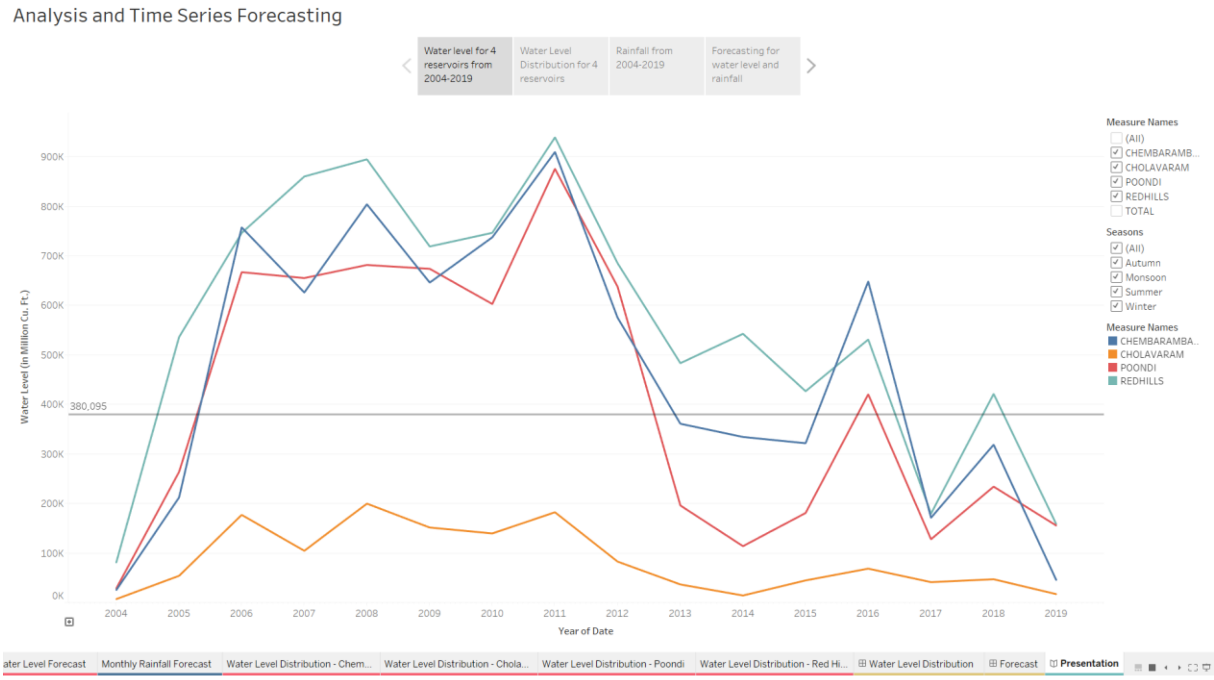


Figure: Reservoir Water Levels, 2004-2019

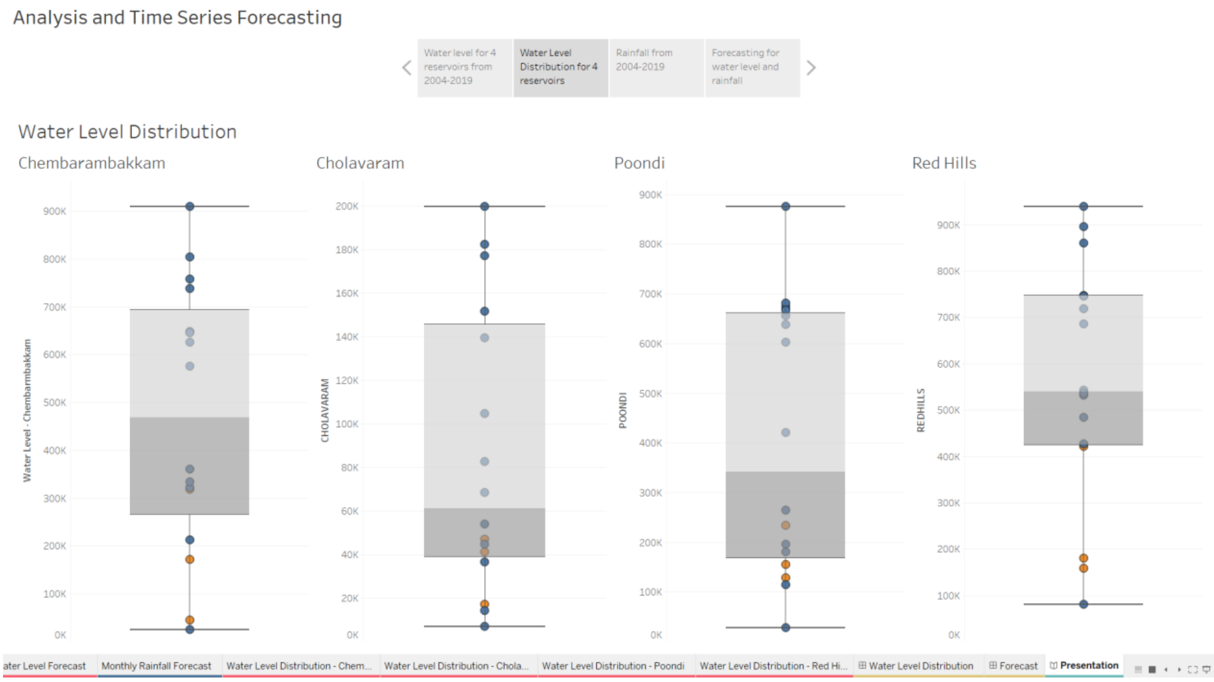


Figure: Distribution of Water Level for 4 Reservoirs, 2004-2019

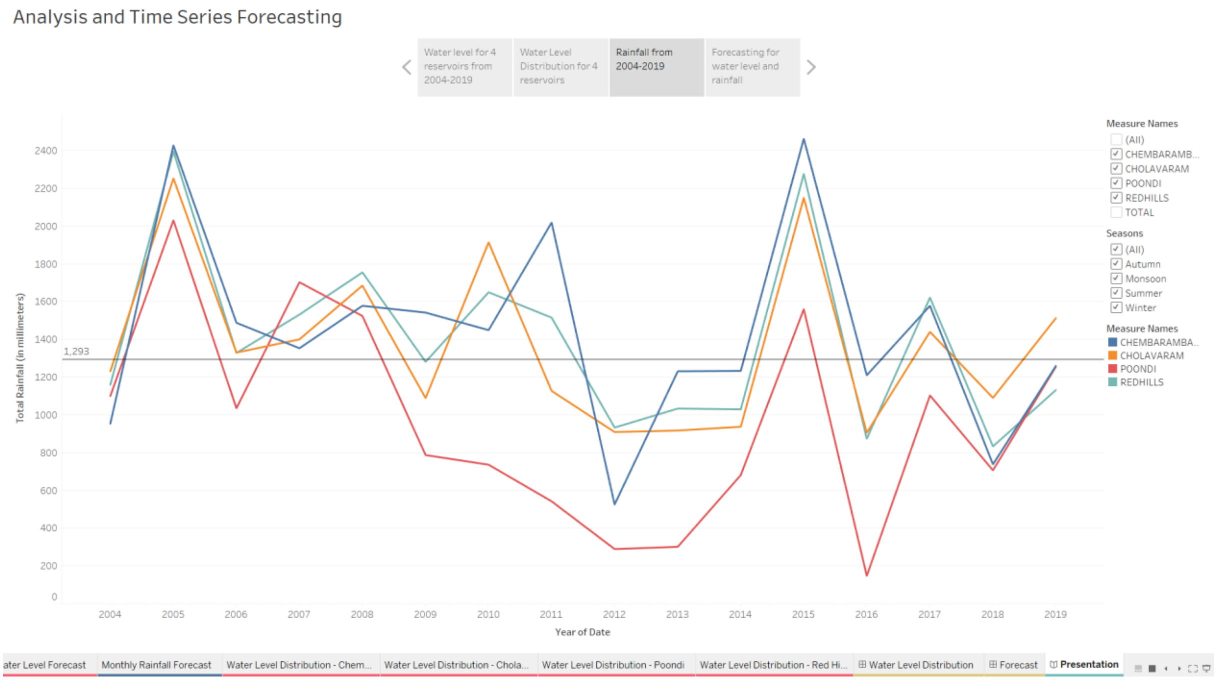


Figure: Reservoir Rainfall Level, 2004-2019

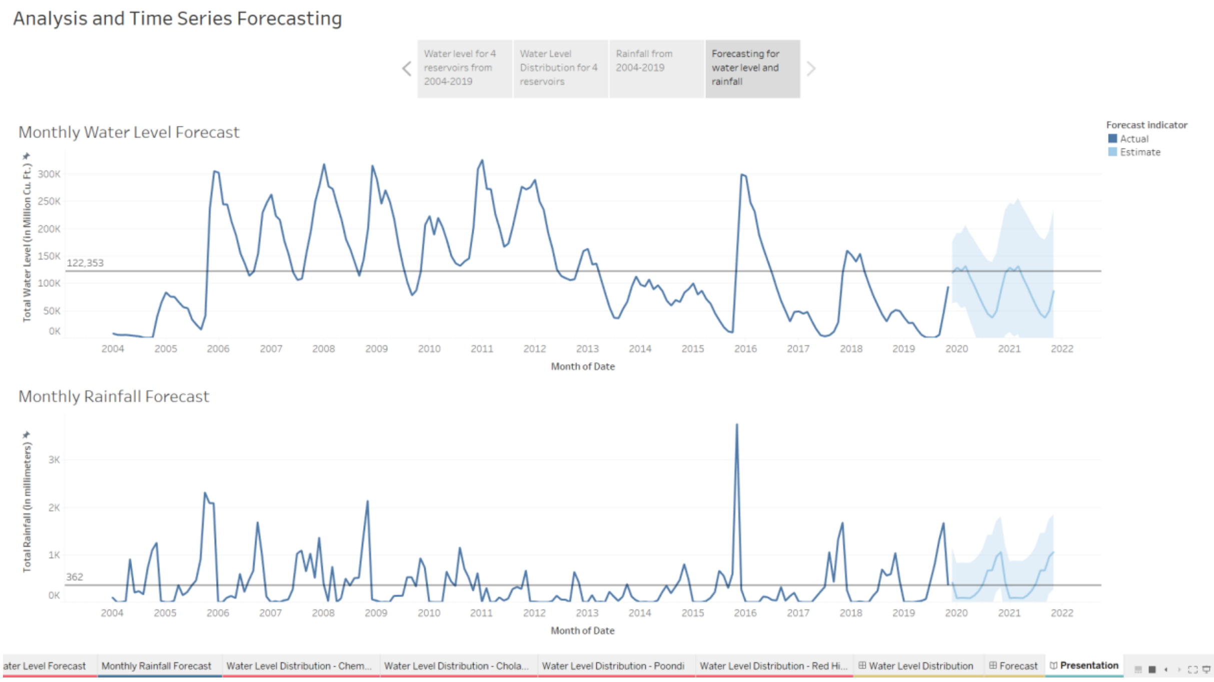


Figure: Forecasting for Reservoir Water and Rainfall Level

CONCLUSION

With this project, we have successfully analysed the historical data for water resources available in Chennai and discovered important patterns and trends in the data. Various statistical modelling techniques, including time series analysis and forecasting, have been used as part of the data analysis. The results of the project and data analysis can prove to be useful in many ways.

REFERENCES

Chennai, Tamil Nadu - https://en.wikipedia.org/wiki/Chennai

Water Management in Chennai - https://en.wikipedia.org/wiki/Water_management_in_Chennai

Puzhal Aeri - https://en.wikipedia.org/wiki/Puzhal_aeri

Chembarambakkam Lake - https://en.wikipedia.org/wiki/Chembarambakkam_Lake

Cholavaram Aeri - https://en.wikipedia.org/wiki/Cholavaram_aeri

2019 Chennai Water Crisis - https://en.wikipedia.org/wiki/2019_Chennai_water_crisis

Chennai Water Management Dataset - https://www.kaggle.com/sudalairajkumar/chennai-water-management?select=chennai_reservoir_rainfall.csv

What is Exploratory Data Analysis? - https://towardsdatascience.com/exploratory-data-analysis-8fc1cb20fd15

Time-series Analysis and Forecasting - https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b

Time-series Modelling - https://www.analyticsvidhya.com/blog/2015/12/complete-tutorial-time-series-modeling/

Exploratory Analysis of Time-series Data - https://www.sciencedirect.com/science/article/pii/S0198971514000179#:~:text=A%20new%20exploratory%20method%20for,groups%20based%20on%20their%20similarities.