OMDENA ALGERIA GREEN MINI PROJECT – TEAM 1

WATER LEVEL FORECASTING

Group members

1. *Freny Reji*
2. *Mada Sai Surya*
3. *Djazila Souhila Korti*
4. *Boukerma Fedoua*
5. *Krishna Basak*

Problem Statement

Objective

Venezia is an Italian city built on a lagoon, a body of water separated from the sea by a strip of land. However, this city is frequently subject to flooding during autumn and spring due to the "acqua alta", a tide that raises the water level to the point that the sea invades the city. This phenomenon causes great complications in the city. The forecast of the water level is therefore a fundamental task for the safeguard of the city of Venezia.

Dataset & Data Description

The dataset contains hourly measurements of the water level in Venezia from 1983-01-01 to 2016-01-01.

To load the dataset, use the following link: https://www.kaggle.com/datasets/lbronchal/venezia

Data Description:

Attribute Information:

- datetime: date and hour of the measurement in utc time
- level: level of the water in cm from the reference point

Proposed Work:

1. Load data
2. EDA implementation
3. Data Processing & Transformation
4. Preparation of Train, Valid and Test datasets
5. Build, Train & Evaluate the model

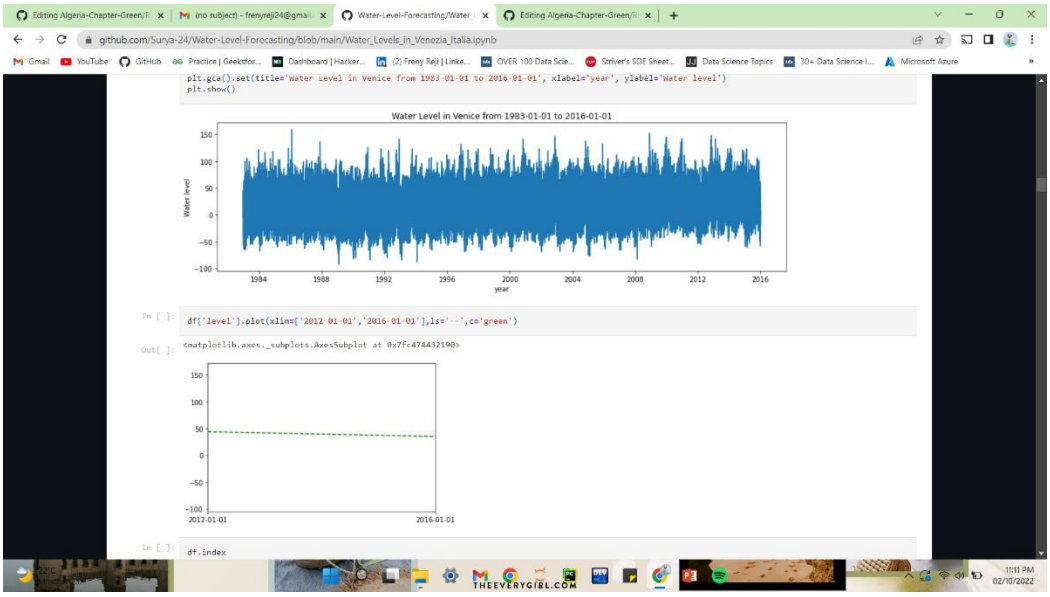
Individual Tasks:

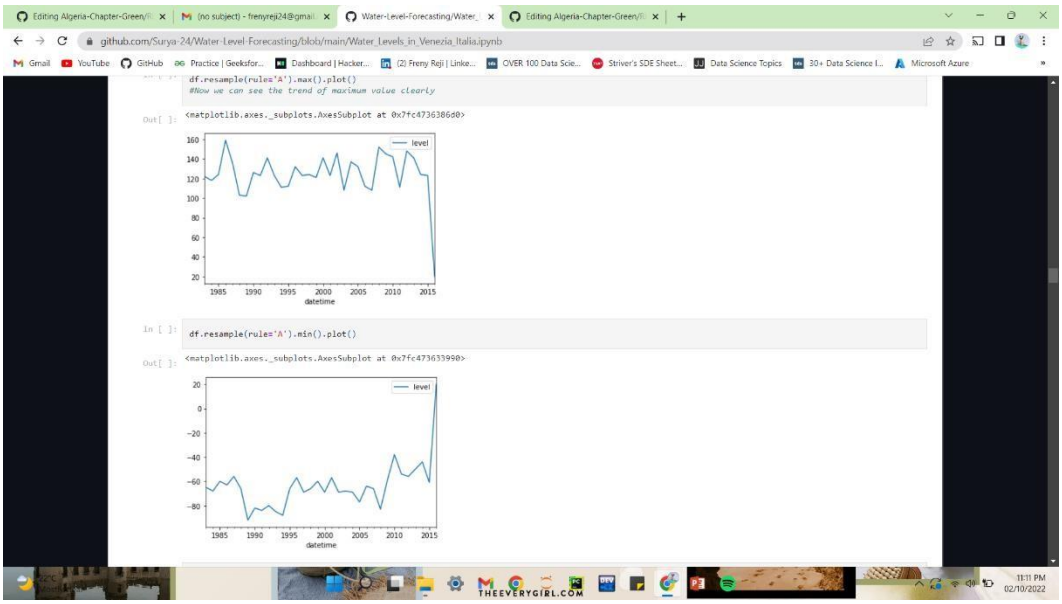
1. Djazila Souhila Korti – **Data Collection, Data Preprocessing, Data Modelling, Model selection , Model training, validation and evaluation, Made final ppt presentation.**

2. Mada Sai Surya(Team Leader) – **Exploratory Data Analysis , Data Modelling , Model Training, video editing.**

3. Freny Reji – **Exploratory Data Analysis ,added /made contents in github , video editing**.

Results:


github.com/Surya-24/Water-Level-Forecasting/blob/main/Water_Levels_in_Venezia_Italia.ipynb
plt.gca().set(title='Water Level in Venice from 1983-01-01 to 2016-01-01', xlabel='year', ylabel='Water level')
plt.show()
Water Level in Venice from 1983-01-01 to 2016-01-01
Water level
year
df['level'].plot(xlim=['2012-01-01','2016-01-01'],ls='--',c='green')
<matplotlib.axes._subplots.AxesSubplot at 0x7fc474432190>
df.index

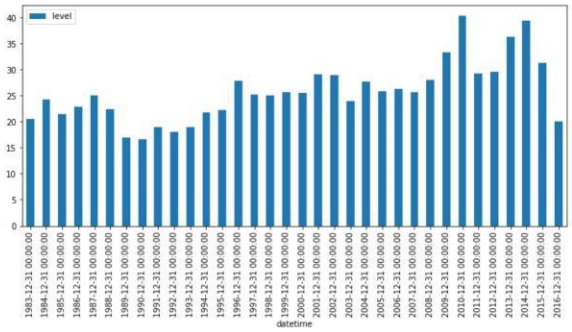

github.com/Surya-24/Water-Level-Forecasting/blob/main/Water_Levels_in_Venezia_Italia.ipynb
df.resample(rule='A').max().plot()
#Now we can see the trend of maximum value clearly
<matplotlib.axes._subplots.AxesSubplot at 0x7fc4736386d0>
level
datetime
df.resample(rule='A').min().plot()
<matplotlib.axes._subplots.AxesSubplot at 0x7fc473633990>
level
datetime

Editing Algeria-Chapter-Green/R | (no subject) - frenyreji24@gmail | Water-Level-Forecasting/Water_ | Editing Algeria-Chapter-Green/R

github.com/Surya-24/Water-Level-Forecasting/blob/main/Water_Levels_in_Venezia_Italia.ipynb

Gmail YouTube GitHub Practice | Geeksfor... Dashboard | Hacker... (2) Freny Reji | Linke... OVER 100 Data Scie... Striver's SDE Sheet... Data Science Topics 30+ Data Science I... Microsoft Azure

Out[ ]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fc474d2a690>
```



Data Processing & Transformation

In [ ]:
```
#Extracting features from date and time this way creating  new features.
data['Month'] = [i.month for i in data['date']]
data['Year'] = [i.year for i in data['date']]
data['day'] = [i.day for i in data['date']]
data['Hour'] = [i.hour for i in data['time']]
data['Minute'] = [i.minute for i in data['time']]
data['Second'] = [i.second for i in data['time']]
#drop unnecessary  columns and re-arrange
data.drop('Minute',axis=1, inplace=True)
data.drop('Second',axis=1, inplace=True)
```

THEEVERYGIRL.COM 11:11 PM 02/10/2022

Editing Algeria-Chapter-Green/R | (no subject) - frenyreji24@gmail | Water-Level-Forecasting/Water_ | Editing Algeria-Chapter-Green/R

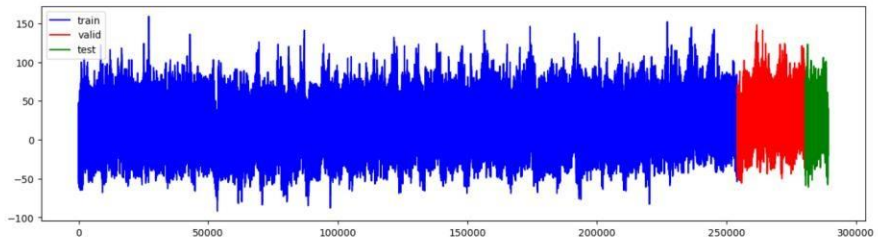
github.com/Surya-24/Water-Level-Forecasting/blob/main/Water_Levels_in_Venezia_Italia.ipynb

Gmail YouTube GitHub Practice | Geeksfor... Dashboard | Hacker... (2) Freny Reji | Linke... OVER 100 Data Scie... Striver's SDE Sheet... Data Science Topics 30+ Data Science I... Microsoft Azure

```
plt.plot(x1,train_y,'b',label='train')
plt.plot(x2, valid_y, 'r', label='valid')
plt.plot(x3,test_y,'g',label='test')

plt.legend(loc='best')
```

Out[ ]:
```
<matplotlib.legend.Legend at 0x7fc48ae83ad0>
```



Model Building

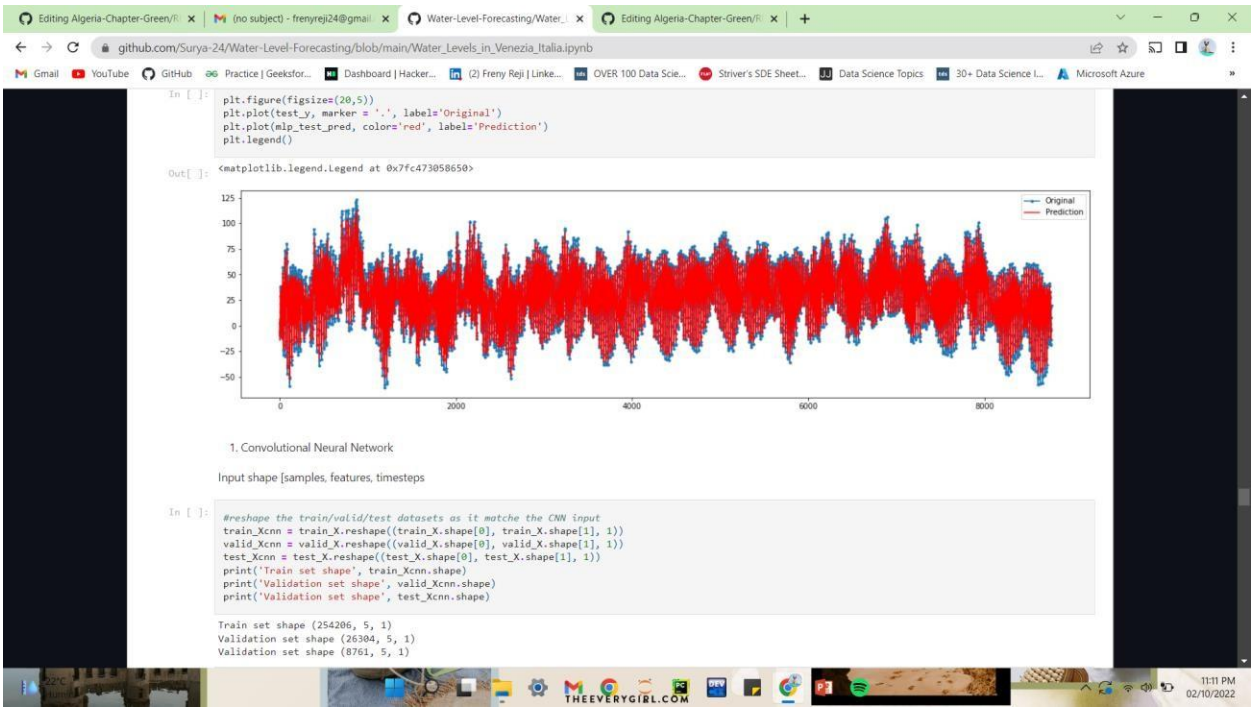
1. Multi Layer Perceptron

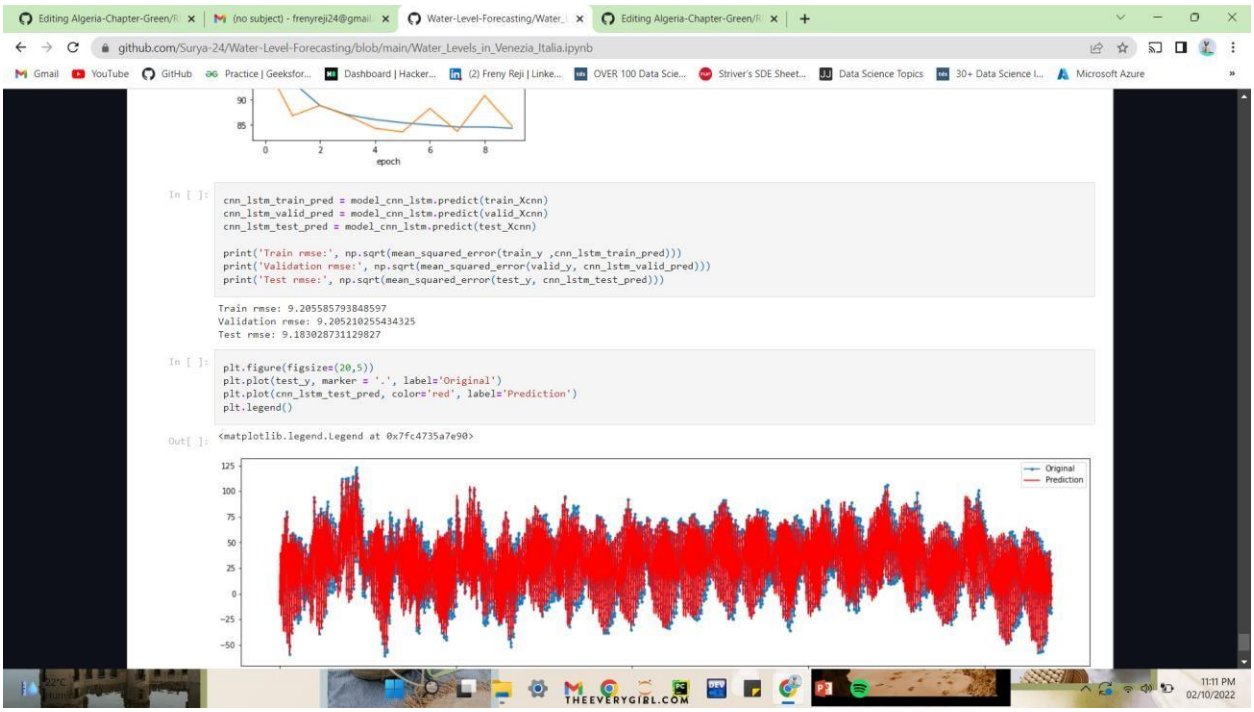
Input shape [samples, timesteps].

In [ ]:
```
model_mlp = Sequential()
model_mlp.add(Dense(100,activation='relu', input_dim=train_X.shape[1]))
model_mlp.add(Dense(50, activation='relu'))
model_mlp.add(Dense(25, activation='relu'))

model_mlp.add(Dense(1))
```

THEEVERYGIRL.COM 11:11 PM 02/10/2022


Editing Algeria-Chapter-Green/R
(no subject) - frenyreji24@gmail
Water-Level-Forecasting/Water_
Editing Algeria-Chapter-Green/R
github.com/Surya-24/Water-Level-Forecasting/blob/main/Water_Levels_in_Venezia_Italia.ipynb
Gmail
YouTube
GitHub
Practice | Geeksfor...
Dashboard | Hacker...
(2) Freny Reji | Linke...
OVER 100 Data Scie...
Striver's SDE Sheet...
Data Science Topics
30+ Data Science I...
Microsoft Azure
In [ ]:
plt.figure(figsize=(20,5))
plt.plot(test_y, marker = '.', label='Original')
plt.plot(mlp_test_pred, color='red', label='Prediction')
plt.legend()
Out[ ]:
<matplotlib.legend.Legend at 0x7fc473058650>
Original
Prediction
1. Convolutional Neural Network
Input shape [samples, features, timesteps
In [ ]:
#reshape the train/valid/test datasets as it matche the CNN input
train_Xcnn = train_X.reshape((train_X.shape[0], train_X.shape[1], 1))
valid_Xcnn = valid_X.reshape((valid_X.shape[0], valid_X.shape[1], 1))
test_Xcnn = test_X.reshape((test_X.shape[0], test_X.shape[1], 1))
print('Train set shape', train_Xcnn.shape)
print('Validation set shape', valid_Xcnn.shape)
print('Validation set shape', test_Xcnn.shape)
Train set shape (254206, 5, 1)
Validation set shape (26304, 5, 1)
Validation set shape (8761, 5, 1)
THEEVERYGIRL.COM
11:11 PM
02/10/2022


Editing Algeria-Chapter-Green/R
(no subject) - frenyreji24@gmail
Water-Level-Forecasting/Water_
Editing Algeria-Chapter-Green/R
github.com/Surya-24/Water-Level-Forecasting/blob/main/Water_Levels_in_Venezia_Italia.ipynb
Gmail
YouTube
GitHub
Practice | Geeksfor...
Dashboard | Hacker...
(2) Freny Reji | Linke...
OVER 100 Data Scie...
Striver's SDE Sheet...
Data Science Topics
30+ Data Science I...
Microsoft Azure
epoch
In [ ]:
cnn_lstm_train_pred = model_cnn_lstm.predict(train_Xcnn)
cnn_lstm_valid_pred = model_cnn_lstm.predict(valid_Xcnn)
cnn_lstm_test_pred = model_cnn_lstm.predict(test_Xcnn)
print('Train rmse:', np.sqrt(mean_squared_error(train_y ,cnn_lstm_train_pred)))
print('Validation rmse:', np.sqrt(mean_squared_error(valid_y, cnn_lstm_valid_pred)))
print('Test rmse:', np.sqrt(mean_squared_error(test_y, cnn_lstm_test_pred)))
Train rmse: 9.205585793848597
Validation rmse: 9.205210255434325
Test rmse: 9.183028731129827
In [ ]:
plt.figure(figsize=(20,5))
plt.plot(test_y, marker = '.', label='Original')
plt.plot(cnn_lstm_test_pred, color='red', label='Prediction')
plt.legend()
Out[ ]:
<matplotlib.legend.Legend at 0x7fc4735a7e90>
Original
Prediction
THEEVERYGIRL.COM
11:11 PM
02/10/2022

CONCLUSION:

The capability of different machine learning models such as the Multi Layer Perceptron (MLP), Long Short Term Memory (LSTM), Convolution Neural Network (CNN), CNN LSTM was investigated for the prediction of water level at a river in Venezia , Italy. Four different input scenarios were investigated, considering correlation analysis. Generally, the comparison of several data-driven regression methods indicated that the exponential GPR model offered better accuracy in predicting daily water levels with respect to different assessment criteria. The findings of this study show the success of the GPR model in capturing the changes in the water level of a river; thus paving the way for which the model can be used in works to mitigate potential risk that may occur in the future from natural events.