DS-600 FINAL PROJECT REPORT

Project Report:

Introduction:

This is study was to use machine learning techniques to estimate air quality. We made use of a dataset that included daily pollution and Air Quality Index (AQI) data for a few different cities.

Briefly Explaining about the Dataset:

The dataset, "city_day.csv," was retrieved from an accessible data source. The dataset includes columns pertaining to several pollutants, including the AQI and AQI bucket for each day, as well as PM2.5, PM10, NO, NO2, CO, SO2, O3, Benzene, Toluene, and Xylene.

The Algorithms that are Used in the air quality prediction:

To estimate the AQI based on pollution levels and other environmental parameters, we used a Random Forest Regressor. This approach can be found in Apache Spark's MLlib and is well-suited for regression problems.

The following stages were engaged in our data processing pipeline:

1. Data loading and examination.
2. To improve clarity, rename columns.
3. Using column means to fill in any missing values.
4. Choosing the target variable (AQI) and pertinent feature columns.
5. Using Vector Assembler to create feature vectors.
6. Dividing the data into sets for testing and training.
7. Developing a model for a Random Forest Regressor.
8. Applying RMSE to the model evaluation**.**

Models:

```
8 hours ago (6s)                                                    12                    Python
# Drop rows with null or NaN values in the target column
data = train_data.dropna(subset=[target_column])

# Train RandomForestRegressor model
rf = RandomForestRegressor(featuresCol="features", labelCol=target_column)
model = rf.fit(data)
datat = test_data.dropna(subset=[target_column])
predictions_var = model.transform(datat)
▸ (8) Spark Jobs
▸ data: pyspark.sql.dataframe.DataFrame
▸ datat: pyspark.sql.dataframe.DataFrame
▸ predictions_var: pyspark.sql.dataframe.DataFrame
```

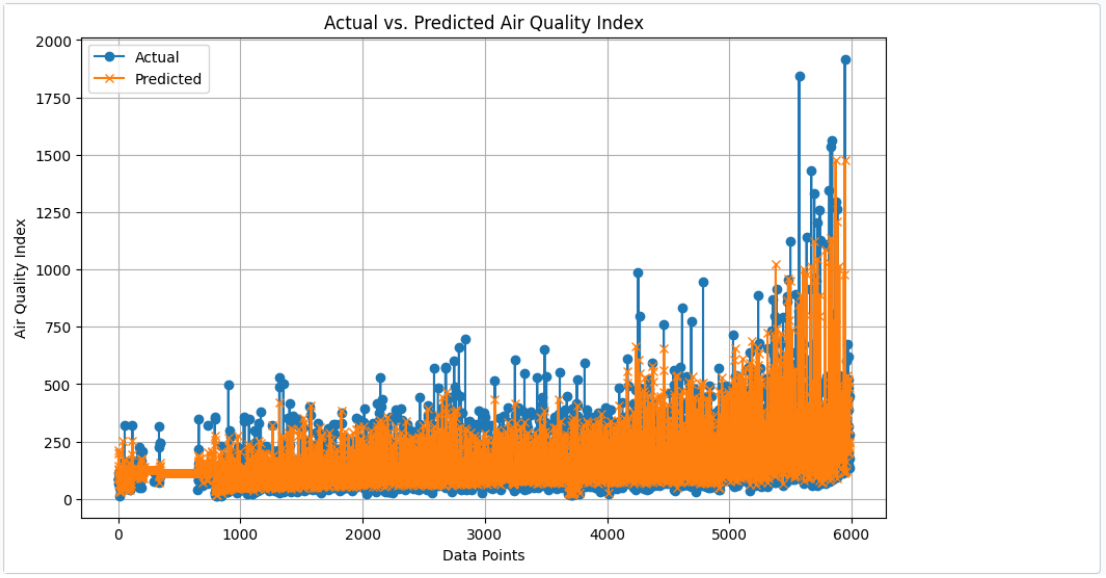
```
2 minutes ago (1s)                                                  21                    Python
# Use the correct label column name
evaluator = RegressionEvaluator(labelCol="Air_Quality_Index", predictionCol="prediction", metricName="r2")

# Calculate the R-squared score
r2_score = evaluator.evaluate(predictions_var)
print("R2 score:", r2_score)
▸ (1) Spark Jobs
R2 score: 0.6549477070014464
```

By trying this out:

After experimenting with several setups, we found that the model's RMSE for air quality prediction was around 82.2957 and R square value 0.65. As we changed the Spark cluster's partition count, we evaluated the model's performance as well.

Visual for actual vs predicted values:



The scatter plot illustrates the general trend of increasing Air Quality Index (AQI) values by comparing actual and expected values. At lower AQI levels, predictions closely match actual values; however, at higher AQI levels, predictions underpredict, suggesting that the model is limited when dealing with abrupt or severe AQI fluctuations.

Final Thoughts and Upcoming Works:

To sum up, we used Apache Spark to successfully create a prediction model for air quality. Our research offers insightful information on how environmental variables and air quality are related. By utilizing more sophisticated methods and data sources, we want to improve the model's accuracy and investigate real-time air quality prediction in the future. Furthermore, our goal is to make this model available to authorities and pertinent parties for real-world application to enhance urban air quality management and monitoring.