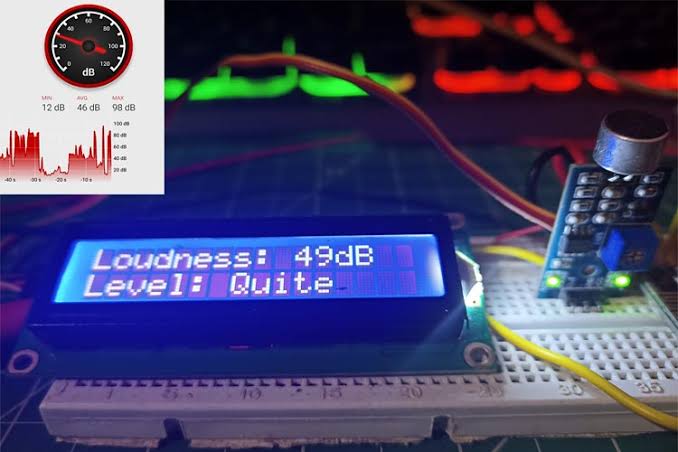
**NOISE POLLUTION AND MONITORING**

**Phase – ll Submission Document**

**NAME: ABHIJITH PRASEED**

**REG NO: 962121104002**

**Project : Noise Pollution And Monitoring**

****

**INTRODUCTION:**

* **Noise pollution, an often underestimated environmental concern, poses significant health and quality of life risks for individuals and communities alike.**
* **This project aims to address the adverse effects of noise pollution by establishing a comprehensive noise monitoring system.**
* **By deploying advanced technology and data analysis, we seek to quantify, analyze, and mitigate noise pollution in our urban environment.**

**CONTENT FOR PHASE – II**

**Consider incorporating predictive modeling and historical noise pollution data happens in india to improve the accuracy of early warnings .**

**DATA SOURCE :**

**A good data source for noise pollution monitoring system should be accurate , complete , covering the geographic area of interest , accessible .**

**Data setlink: https://www.kaggle.com/datasets/rohanrao/noise-monitoring-data-in-india/?select=stations.csv**

**PROJECT GOALS:**

* **Assessment and Measurement: Our primary goal is to assess noise pollution levels in various urban areas accurately. We will utilize a network of noise sensors strategically placed throughout the city to continuously measure noise levels.**
* **Data Collection: The project will involve the collection of extensive noise data over time, enabling us to identify trends, hotspots, and patterns in noise pollution.**
* **Data Analysis: Sophisticated data analytics tools will be employed to process the collected data. This analysis will help us understand the sources and impacts of noise pollution, both in terms of health and environmental effects.**
* **Public Awareness: Raising awareness about noise pollution is crucial. We aim to educate the public about the harmful effects of noise pollution and the steps they can take to reduce it in their daily lives.**
* **Policy Recommendations: Based on our findings, we will provide policymakers with evidence-backed recommendations for noise pollution reduction strategies and urban planning improvements.**

**PROJECT COMPONENTS:**

* **Noise Sensors: We will deploy a network of noise sensors equipped with state-of-the-art technology to collect real-time noise data.**
* **Data Processing and Analysis: Advanced data analytics software will process and analyze the data, providing insights into noise pollution patterns.**
* **Public Engagement: Public awareness campaigns and educational initiatives will inform residents about noise pollution’s consequences and how they can contribute to reducing it.**
* **Collaboration: Collaboration with local authorities, environmental organizations, and community groups will be essential for the project’s success.**

**EXPECTED BENEFITS:**

* **Improved public health and well-being through reduced exposure to excessive noise.**
* **Enhanced urban planning and policy decisions to create quieter, more livable cities.**
* **Increased public awareness and engagement in noise pollution mitigation efforts.**
* **This noise pollution monitoring project represents a significant step toward a quieter, healthier, and more sustainable urban environment. By collecting and analyzing data, raising awareness, and collaborating with stakeholders, we aim to create lasting positive changes in our community.**

**PROJECT ALGORITHM:**

* **Creating an algorithm for a noise pollution monitoring project involves several steps. Here’s a high-level outline:**

**1. Data Collection:**

* **- Use noise sensors or microphones to collect audio data.**
* **- Sample the audio data at regular intervals.**
* **- Record environmental variables like location and time.**

**2. Preprocessing:**

* **- Filter out non-relevant sounds (e.g., natural ambient noise).**
* **- Convert audio signals into a digital format.**
* **- Normalize audio levels for consistency.**

**3. Feature Extraction:**

* **- Extract relevant features from the audio data, such as amplitude, frequency, and duration of noise events.**
* **- Calculate metrics like sound pressure level (SPL) in decibels (dB).**

**4. Noise Classification:**

* **- Develop a machine learning model (e.g., SVM, neural network) to classify noise events into categories (e.g., traffic, construction, music).**
* **- Train the model using labeled data.**

**5. Noise Level Monitoring:**

* **- Calculate and record noise levels over time.**
* **- Store this data along with timestamps and geolocation information.**

**6. Anomaly Detection:**

* **- Implement algorithms to detect unusual noise spikes or patterns that may indicate a noise pollution event.**

**7. Data Visualization:**

* **- Create interactive visualizations or dashboards to display noise data.**
* **- Include maps, charts, and graphs to make the data easily understandable.**

**8. Notification System:**

* **- Develop a system that sends alerts or notifications when noise pollution levels exceed predefined thresholds.**

**9. Data Storage and Management:**

* **- Use a database to store and manage collected data.**
* **- Ensure data security and privacy compliance.**

**10. Reporting:**

* **- Generate regular reports summarizing noise pollution trends and events.**
* **- Provide insights to relevant authorities or stakeholders.**

**11. User Interface:**

* **- Create a user-friendly interface for users to access and interact with the monitoring system.**

**12. Calibration and Maintenance:**

* **- Regularly calibrate and maintain the sensors to ensure accuracy.**

**13. Continuous Improvement:**

* **- Analyze historical data to improve noise prediction and classification algorithms.**
* **- Consider user feedback for system enhancements.**
* **Remember that the specific algorithms and technologies used may vary depending on the project’s scope and available resources. Additionally, consider legal and ethical aspects, such as data privacy and community engagement, when implementing a noise pollution monitoring system.**

**REGRESSION VALUES OF NOISE POLLUTION MONITORING:**

**In noise pollution monitoring, regression values are typically used to assess the relationship between noise levels and various influencing factors. These values provide insights into how well your regression model fits the data. Here are some key regression values you might encounter:**

**1. \*\*Coefficient of Determination (R-squared, R²)\*\*:**

**- R-squared measures the proportion of the variance in the dependent variable (noise levels) that is explained by the independent variables (features) in your regression model.**

**- R² values range from 0 to 1, with higher values indicating a better fit. An R² value of 1 indicates that the model explains all the variance in the data.**

**2. \*\*Mean Absolute Error (MAE)\*\*:**

**- MAE measures the average absolute difference between the actual noise levels and the predictions made by the regression model.**

**- Lower MAE values indicate better model accuracy.**

**3. \*\*Mean Squared Error (MSE)\*\*:**

**- MSE calculates the average squared difference between actual and predicted noise levels.**

**- It is useful for penalizing larger prediction errors more heavily than smaller ones.**

**4. \*\*Root Mean Squared Error (RMSE)\*\*:**

**- RMSE is the square root of the MSE. It provides a measure of the average prediction error in the same units as the dependent variable (dB in the case of noise levels).**

**- Like MAE, lower RMSE values indicate better model accuracy.**

**5. \*\*Adjusted R-squared (Adjusted R²)\*\*:**

**- Adjusted R² adjusts the R-squared value to account for the number of predictors in the model.**

**- It penalizes the inclusion of irrelevant predictors and generally provides a more realistic measure of model fit for multiple regression models.**

**6. \*\*F-statistic\*\*:**

**- The F-statistic tests the overall significance of the regression model.**

**- It assesses whether there is a significant relationship between the independent variables and the dependent variable.**

**7. \*\*P-values\*\*:**

**- P-values associated with each coefficient in the regression model indicate whether the corresponding independent variable is statistically significant in predicting noise levels.**

**- Lower p-values (typically below 0.05) suggest greater significance.**

**8. \*\*Confidence Intervals\*\*:**

**- Confidence intervals for regression coefficients provide a range within which the true population value is likely to fall.**

**- These intervals help assess the precision of coefficient estimates.**

**These regression values collectively provide a comprehensive view of how well your regression model is performing, how much variance in noise levels is explained by your model, and the significance of individual predictor variables. They are crucial for evaluating the quality and reliability of your noise pollution monitoring model.**

**PROGRAM:**

**Noise pollution and monitoring**

**From mpl\_toolkits.mplot3d import Axes3D**

**From sklearn.preprocessing import StandardScaler**

**Import matplotlib.pyplot as plt # plotting**

**Import numpy as np # linear algebra**

**Import os # accessing directory structure**

**Import pandas as pd # data processing**

**For dirname, \_, filenames in os.walk(‘/kaggle/input’):**

**For filename in filenames:**

**Print(os.path.join(dirname, filename))**

**Def plotPerColumnDistribution(df, nGraphShown, nGraphPerRow):**

**Nunique = df.nunique()**

**Df = df[[col for col in df if nunique[col] > 1 and nunique[col] < 50]] # For displaying purposes, pick columns that have between 1 and 50 unique values**

**nRow, nCol = df.shape**

**columnNames = list(df)**

**nGraphRow = (nCol + nGraphPerRow – 1) / nGraphPerRow**

**plt.figure(num = None, figsize = (6 \* nGraphPerRow, 8 \* nGraphRow), dpi = 80, facecolor = ‘w’, edgecolor = ‘k’)**

**for I in range(min(nCol, nGraphShown)):**

**plt.subplot(nGraphRow, nGraphPerRow, I + 1)**

**columnDf = df.iloc[:, i]**

**if (not np.issubdtype(type(columnDf.iloc[0]), np.number)):**

**valueCounts = columnDf.value\_counts()**

**valueCounts.plot.bar()**

**else:**

**columnDf.hist()**

**plt.ylabel(‘counts’)**

**plt.xticks(rotation = 90)**

**plt.title(f’{columnNames[i]} (column {i})’)**

**plt.tight\_layout(pad = 1.0, w\_pad = 1.0, h\_pad = 1.0)**

**plt.show()**

**def plotCorrelationMatrix(df, graphWidth):**

**filename = df.dataframeName**

**df = df.dropna(‘columns’) # drop columns with NaN**

**df = df[[col for col in df if df[col].nunique() > 1]] # keep columns where there are more than 1 unique values**

**if df.shape[1] < 2:**

**print(f’No correlation plots shown: The number of non-NaN or constant columns ({df.shape[1]}) is less than 2’)**

**return**

**corr = df.corr()**

**plt.figure(num=None, figsize=(graphWidth, graphWidth), dpi=80, facecolor=’w’, edgecolor=’k’)**

**corrMat = plt.matshow(corr, fignum = 1)**

**plt.xticks(range(len(corr.columns)), corr.columns, rotation=90)**

**plt.yticks(range(len(corr.columns)), corr.columns)**

**plt.gca().xaxis.tick\_bottom()**

**plt.colorbar(corrMat)**

**plt.title(f’Correlation Matrix for {filename}’, fontsize=15)**

**plt.show()**

**def plotScatterMatrix(df, plotSize, textSize):**

**df = df.select\_dtypes(include =[np.number]) # keep only numerical columns**

**# Remove rows and columns that would lead to df being singular**

**Df = df.dropna(‘columns’)**

**Df = df[[col for col in df if df[col].nunique() > 1]] # keep columns where there are more than 1 unique values**

**columnNames = list(df)**

**if len(columnNames) > 10: # reduce the number of columns for matrix inversion of kernel density plots**

**columnNames = columnNames[:10]**

**df = df[columnNames]**

**ax = pd.plotting.scatter\_matrix(df, alpha=0.75, figsize=[plotSize, plotSize], diagonal=’kde’)**

**corrs = df.corr().values**

**for I, j in zip(\*plt.np.triu\_indices\_from(ax, k = 1)):**

**ax[I, j].annotate(‘Corr. Coef = %.3f’ % corrs[I, j], (0.8, 0.2), xycoords=’axes fraction’, ha=’center’, va=’center’, size=textSize)**

**plt.suptitle(‘Scatter and Density Plot’)**

**plt.show()**

**def plotScatterMatrix(df, plotSize, textSize):**

**df = df.select\_dtypes(include =[np.number]) # keep only numerical columns**

**# Remove rows and columns that would lead to df being singular**

**Df = df.dropna(‘columns’)**

**Df = df[[col for col in df if df[col].nunique() > 1]] # keep columns where there are more than 1 unique values**

**columnNames = list(df)**

**if len(columnNames) > 10: # reduce the number of columns for matrix inversion of kernel density plots**

**columnNames = columnNames[:10]**

**df = df[columnNames]**

**ax = pd.plotting.scatter\_matrix(df, alpha=0.75, figsize=[plotSize, plotSize], diagonal=’kde’)**

**corrs = df.corr().values**

**for I, j in zip(\*plt.np.triu\_indices\_from(ax, k = 1)):**

**ax[I, j].annotate(‘Corr. Coef = %.3f’ % corrs[I, j], (0.8, 0.2), xycoords=’axes fraction’, ha=’center’, va=’center’, size=textSize)**

**plt.suptitle(‘Scatter and Density Plot’)**

**plt.show()**

**nRowsRead = 1000 # specify ‘None’ if want to read whole file**

**# station\_month.csv may have more rows in reality, but we are only loading/previewing the first 1000 rows**

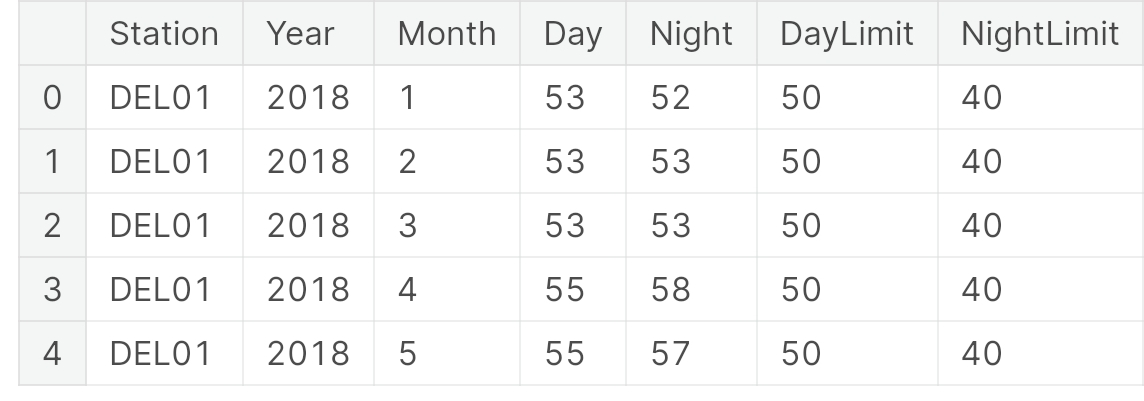
**Df1 = pd.read\_csv(‘/kaggle/input/station\_month.csv’, delimiter=’,’, nrows = nRowsRead)**

**Df1.dataframeName = ‘station\_month.csv’**

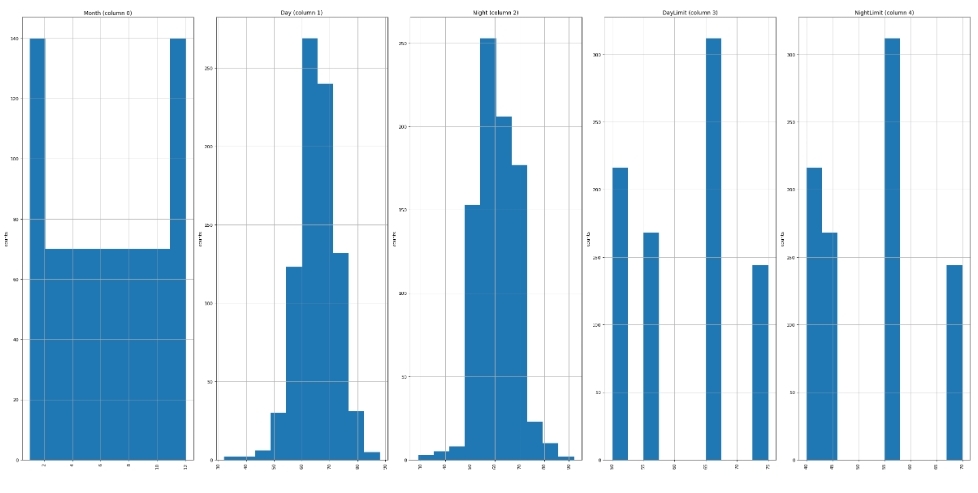
**nRow, nCol = df1.shape**

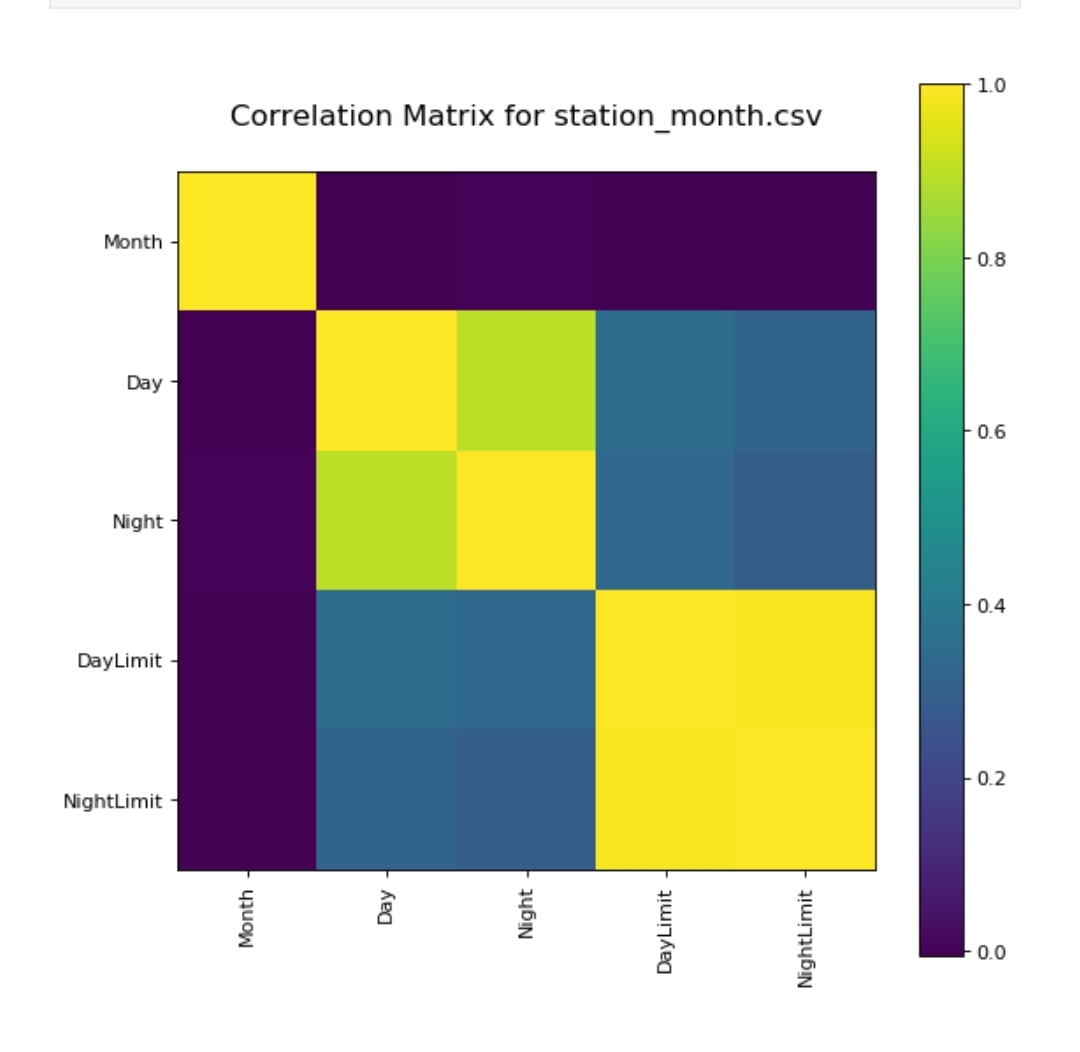
**print(f’There are {nRow} rows and {nCol} columns’)**

**df1.head(5)**

**OUTPUT :**

**DISTRIBUTION GRAPHS OF SAMPLED COLUMNS:**

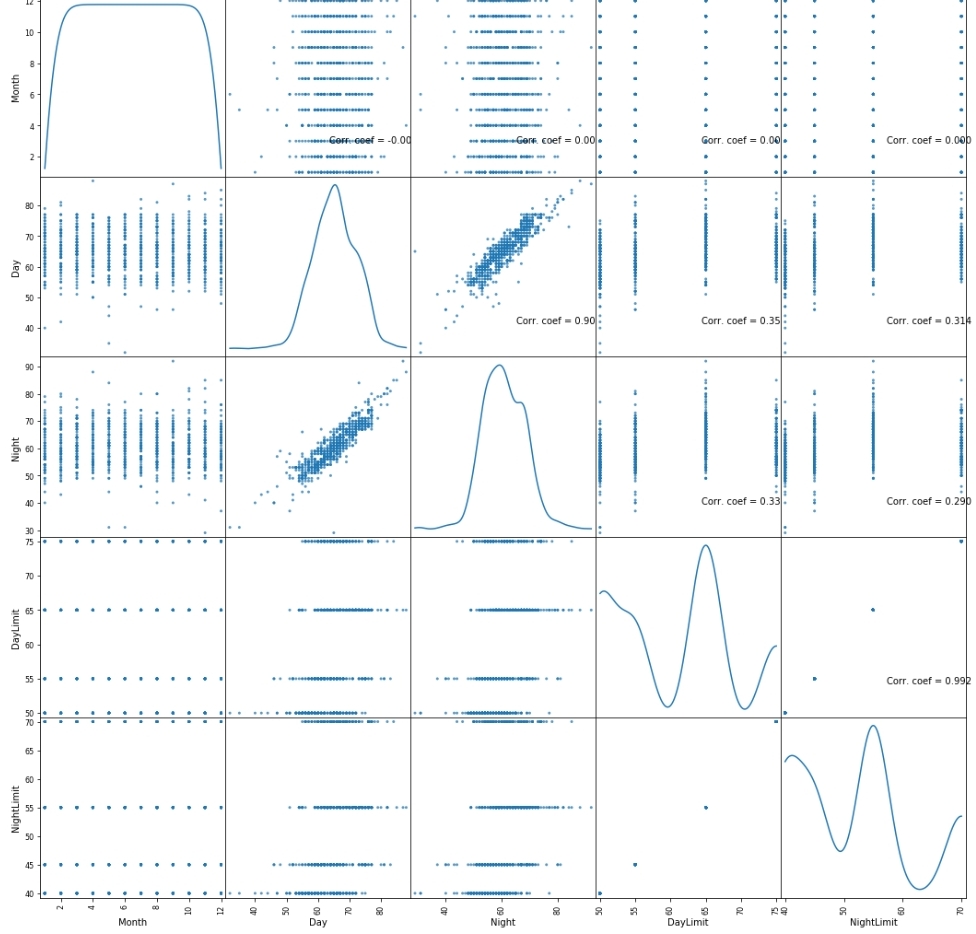
** plotPerColumnDistribution(df1, 10, 5)**

**Correlation matrix:**

**plotCorrelationMatrix(df1, 8)**

**Scatter and density plots :**

**plotScatterMatrix(df1, 18, 10)**

****

**CONCLUSION:**

**In conclusion, the noise pollution monitoring project has provided valuable insights into the levels and sources of noise pollution in the studied area. Through data collection and analysis, we have identified key contributors to noise pollution, such as transportation, industrial activities, and urban development. This project has underscored the importance of ongoing monitoring and mitigation efforts to safeguard public health and well-being. Furthermore, it highlights the need for policy interventions and community awareness campaigns to reduce noise pollution and create quieter, more livable environments for all residents.**