Capstone Project Report: Smart Pressure Control Prediction

Project Title:

Smart Pressure Control Prediction: Enhancing Industrial Efficiency through Real-Time Data and Predictive Algorithms

Executive Summary

The Smart Pressure Control Prediction project aims to improve industrial pressure management by automating and optimizing pressure regulation through the application of machine learning models. Traditionally, pressure regulation required manual adjustments, leading to inefficiencies and higher energy consumption. This system uses predictive algorithms to optimize pressure control in real-time, significantly reducing operational costs and energy wastage. The project follows a structured data science approach, including data preprocessing, exploratory data analysis (EDA), model training, evaluation, hyperparameter tuning, and deployment using **Streamlit** for real-time predictions.

Table of Contents

1. Project Overview

The aim of this project is to develop a machine learning system that optimizes pressure regulation in industrial settings. By automating the process, it enhances operational efficiency and sustainability. The primary objective is to predict the target variable **mmH2O** (a pressure unit) using various input features like temperature, flow rate, and humidity.

Dataset:

- **Source**: Kaggle Dataset for Smart Pressure Control.
- **Features**: The dataset contains multiple variables related to pressure regulation, including temperature, pressure levels, flow rate, and humidity.
- **Target Variable**: **mmH2O** (Pressure in millimeters of water).

2. Data Preprocessing

Preprocessing is crucial to clean the dataset and prepare it for analysis. The following steps were taken:

Data Cleaning:

- **Handling Missing Values**: Checked for missing values in the dataset and verified that no significant imputation was required.
- **Duplicate Removal**: Duplicates were found and removed to avoid skewing model performance.
- **Outlier Detection & Removal**: Outliers were detected using the Interquartile Range (IQR) method. The extreme values were removed to prevent them from affecting model predictions.

Data Integration:

- Merged the **train** and **test** datasets for exploratory data analysis and model training.

Data Summary:

- **Shape**: The final dataset after preprocessing had 4,320 rows and multiple features.
- **Missing Values**: No missing values were detected after preprocessing.

3. Exploratory Data Analysis (EDA)

EDA was performed to understand the relationships between features and their impact on the target variable. Key insights were gained using visualizations and statistical measures:

Univariate Analysis:

- **Histograms & Boxplots**: Visualized the distribution of each numerical feature to understand central tendencies and identify potential outliers.

Skewness and Kurtosis:

- These metrics were calculated to check if the data distribution was symmetric or skewed, guiding further transformations.

Correlation Analysis:

- **Correlation Matrix**: A correlation matrix was created to determine the relationship between features and the target variable (mmH2O). Features like **temperature** and **flow rate** showed a strong correlation with pressure, making them important predictors.

---

4. Feature Selection

Feature selection was based on correlation analysis and feature importance from model training. Features with an absolute correlation of greater than 0.2 with the target variable were selected:

Selected Features:

- **Temperature**
- **Pressure Levels**
- **Flow Rate**
- **Humidity**

These features were used in the final model to ensure predictive accuracy while avoiding overfitting.

---

5. Model Building and Evaluation

Multiple machine learning models were trained and evaluated using the preprocessed dataset. The models were evaluated on their ability to predict the target variable (mmH2O).

Models Trained:

1. **Linear Regression**
2. **Ridge Regression**
3. **Lasso Regression**
4. **Random Forest Regressor**
5. **Gradient Boosting Regressor**
6. **Support Vector Regressor**
7. **Neural Networks (MLP Regressor)**
8. **Extra Trees Regressor**

Evaluation Metrics:

- **R² Score**: Indicates the proportion of variance explained by the model.
- **Mean Absolute Error (MAE)**: A key metric used to assess model performance by calculating the average absolute error in predictions.

6. Model Comparison

A comparison of the models based on **Mean Absolute Error (MAE)** is as follows:

Model	MAE
Linear Regression	23.14
Ridge Regression	22.9
Lasso Regression	23.0
Random Forest Regressor	**12.2**
Gradient Boosting Regressor	14.1
Support Vector Regressor	18.2
Neural Networks (MLP)	16.7
Extra Trees Regressor	12.5

Best Model:

- **Random Forest Regressor** achieved the lowest MAE (12.2), making it the best-performing model.
- **Feature Importance**: Random Forest helped rank feature importance, identifying **Temperature** and **Flow Rate** as the most critical factors influencing pressure regulation.

7. Hyperparameter Tuning

The Random Forest model was fine-tuned using **GridSearchCV** to optimize its hyperparameters. The parameters that were tuned include:

- **n_estimators**: Number of trees in the forest.
- **max_depth**: Maximum depth of the tree.

- **min_samples_split**: Minimum number of samples required to split a node.
- **max_features**: Number of features to consider for the best split.

Best Parameters Found:

- **n_estimators**: 300
- **max_depth**: 20
- **min_samples_split**: 5
- **max_features**: 'sqrt'

The tuned Random Forest model showed improved performance with even lower errors.

---

8. Model Deployment with Streamlit

The final **Random Forest Regressor** model was deployed using **Streamlit**, a Python framework that simplifies web app creation for machine learning models.

Features of the Streamlit App:

- **User Input Interface**: Users can input values for features like temperature, flow rate, and pressure levels to get real-time predictions.
- **Real-Time Predictions**: The app uses the trained Random Forest model to predict the pressure values (mmH2O) based on user inputs.

9. Conclusion

The Smart Pressure Control Prediction project successfully demonstrates how machine learning can optimize industrial pressure regulation. Through data preprocessing, feature selection, model training, and hyperparameter tuning, the **Random Forest Regressor** was found to be the most effective model, achieving a **MAE of 12.2**.

The project was deployed using **Streamlit**, allowing for real-time predictions and user interaction. This system is poised to make significant improvements in industrial efficiency by automating pressure control and reducing energy waste.

---

10. Future Work

Future improvements to this project may include:

- **Integration with IoT Devices**: For real-time data collection and prediction in industrial environments.
- **Model Enhancements**: Explore deep learning models for further accuracy improvements.
- **Scalability**: Extend the model to work in larger industrial setups with multiple zones.
- **Real-Time Monitoring Dashboard**: Add real-time monitoring and alert systems for pressure anomalies.