**Quality Prediction in Injection Molding Production**

This project aims to predict the quality of products manufactured in an injection molding production setting. The primary focus is on developing a data-driven approach to identify factors that affect product quality and to implement predictive models to enhance production consistency.

**Project Description**

Injection molding is a manufacturing process for producing parts by injecting molten material into a mold. Quality control is critical in injection molding to ensure that the produced parts meet required standards. This project utilizes a dataset from an injection molding production environment to predict product quality, helping manufacturers detect defects early and make necessary adjustments to the process parameters.

The workflow includes the following steps:

1. **Data Collection**: The dataset is uploaded using Google Colab, allowing easy access and analysis in a cloud environment.
2. **Data Exploration and Cleaning**:
   * Display the first few rows of the dataset to understand its structure.
   * Check for missing values and analyse columns with missing data to identify data quality issues.
3. **Data Preprocessing**:
   * Handling missing values and outliers.
   * Feature engineering to create new variables or transform existing ones to better represent the factors affecting product quality.
4. **Exploratory Data Analysis (EDA)**:
   * Visualize important features such as temperature, pressure, and cycle time to understand their influence on product quality.
   * Perform statistical analysis to determine relationships between input variables and quality outcomes.
5. **Predictive Modelling**:
   * Train machine learning models to predict whether a product meets the quality standards or has defects.
   * Models considered include Logistic Regression (Randomized SearchCV), Logistic Regression (GridSearch CV), Logistic Regression Cross Validation (5 folds), and Random Forest Cross Validation (3 folds).
6. **Model Evaluation**:
   * Assess model performance using metrics such as accuracy, precision, recall, and F1-score.
   * Perform hyperparameter tuning to optimize model performance.
7. **Deployment** (Optional):
   * Provide an outline for deploying the predictive model in a real-time production environment for continuous quality monitoring.

**Technologies Used**

* **Python**: For data analysis and modelling.
* **Pandas and NumPy**: For data manipulation and numerical operations.
* **Scikit-learn**: For building machine learning models.
* **Matplotlib/Seaborn**: For data visualization.
* **Google Colab**: Cloud-based platform for interactive Python development.

**Installation and Setup**

To run the code in this repository, follow these steps:

1. Clone the repository:

git clone <https://github.com/yourusername/Quality_Prediction_Injection_Molding.git>

1. Navigate to the project directory:

cd Quality\_Prediction\_Injection\_Molding

1. Install the required dependencies:

pip install -r requirements.txt

1. Open the notebook in Jupyter or Google Colab:

jupyter notebook Quality\_Prediction\_in\_Injection\_Molding.ipynb

**Usage**

* **Data Upload**: Upload the injection molding dataset (Injection\_Molding\_Dataset.csv).
* **Run the Notebook**: Execute the cells in sequence to clean, explore, and model the data.
* **Model Training and Evaluation**: The notebook guides you through the steps to train predictive models and evaluate their performance.

**Dataset**

The dataset used in this project has a shape of **(1000, 13)**, containing 1000 rows and 13 columns. Each row represents an instance of a production cycle, and each column represents a feature that is relevant to the quality prediction.

The features include:

* **ZUx (Cycle Time)**: The time taken to complete each molding cycle (in seconds).
* **SKs (Clamping Force Peak Value)**: The peak value of the clamping force during the cycle (in kN).
* **CRn (Screw Position at End of Hold Pressure)**: The position of the screw at the end of the hold pressure phase (in mm).
* **APSs (Specific Back Pressure Peak Value)**: The specific back pressure peak value (in bar).
* **APVs (Specific Injection Pressure Peak Value)**: The specific injection pressure peak value (in bar).
* **SVx (Shot Volume)**: The volume of the material shot into the mold (in cm³).
* **SKx (Closing Force)**: The closing force applied during the process (in kN).
* **Melt Temperature**: Temperature of the molten material during the molding process (in degrees Celsius).
* **ZDx (Plasticizing Time)**: Time taken for plasticizing the material (in seconds).
* **Ms (Torque Peak Value Current Cycle)**: The peak torque value during the current cycle (in Nm).
* **Time\_to\_fill**: The time taken to fill the mold cavity (in seconds).
* **Mold Temperature**: Temperature of the mold during the production process (in degrees Celsius).
* **Mn (Torque Mean Value Current Cycle)**: The mean torque value during the current cycle (in Nm).

The target variable is the **Quality Indicator**, which has two classes:

* **1**: Represents good quality.
* **2**: Represents defective quality.

**Results**

The predictive models help in identifying the key parameters that significantly influence product quality. The model can provide early warnings in case of suboptimal parameter settings, thus allowing timely adjustments in the production process.

**Model Results**

The following table summarizes the results of different models used for quality prediction:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision (Class 1)** | **Recall (Class 1)** | **F1-Score (Class 1)** | **Precision (Class 2)** | **Recall (Class 2)** | **F1-Score (Class 2)** |
| Logistic Regression (Randomized CV) | 0.72 | 0.73 | 0.74 | 0.73 | 0.72 | 0.71 | 0.71 |
| Logistic Regression (GridSearch CV) | 0.79 | 0.78 | 0.82 | 0.80 | 0.79 | 0.76 | 0.77 |
| Logistic Regression (5-Fold CV) | 0.73 | 0.73 | 0.74 | 0.74 | 0.72 | 0.71 | 0.72 |
| Random Forest (3-Fold CV) | 0.89 | 0.88 | 0.93 | 0.90 | 0.92 | 0.87 | 0.89 |

**Contributing**

Contributions are welcome! Please create an issue or submit a pull request for any improvements or suggestions.

**License**

This project is licensed under the MIT License - see the LICENSE file for details.

**Contact**

For any inquiries, please contact:

* **Name**: Tewogbade Shakir Adeyemi
* **Email**: [pingcommercial@gmail.com](mailto:pingcommercial@gmail.com)