Garment Worker Productivity Prediction Project | HubbleMind

Project Overview:

This project aims to build a machine learning model that predicts productivity levels of garment workers based on various operational factors within a manufacturing setting. By analyzing features like work-in-progress, overtime, incentives, and team dynamics, interns will develop models to predict continuous productivity values, helping businesses improve their workflow efficiency and resource management.

Dataset : Download | Data Source : CC BY 4.0 | UCI Archive

Dataset:

The dataset used for this project is the **Garment Worker Productivity Dataset**. It includes 1,197 records and 14 features, covering different attributes related to the garment production process, such as targeted productivity, overtime, and the actual productivity achieved by each team. Each row represents a record for a team on a particular day, and the target variable (actual productivity) indicates the team's performance.

Key Features:

- **date**: Date of the record.
- **quarter**: The quarter of the year (e.g., Q1, Q2).
- **department**: Department of workers (e.g., sewing, finishing).
- **team**: The number representing the team.
- **targeted_productivity**: The target productivity (between 0 and 1).
- **smv**: Standard Minute Value (time required to complete the task).
- **wip**: Work In Progress (missing values present).
- **over_time**: Overtime in minutes.
- **incentive**: Bonus paid to workers.
- **idle_time**: Time during which no work was done.
- **idle_men**: Number of idle workers.
- **no_of_style_change**: Number of style changes in production.
- **no_of_workers**: Number of workers in the team.
- **actual_productivity**: Target variable representing the productivity achieved (between 0 and 1).

Week 1: Data Understanding and Preprocessing

1. **Dataset Exploration:**
    - Import the dataset, explore its structure, and differentiate between categorical and numerical columns.
    - Understand the distribution and data types of key columns.
2. **Data Cleaning:**
    - Handle missing values, such as imputing the wip column using the median.
    - Detect and manage outliers in columns like idle_time, incentive, and actual_productivity.
3. **Feature Engineering:**
    - Perform one-hot encoding for categorical features (e.g., quarter, department).
    - Extract useful date features such as month and day_of_week.
4. **Feature Scaling:**
    - Scale numerical features (e.g., smv, over_time) using StandardScaler for consistency in model training.

Week 2: Exploratory Data Analysis (EDA)

1. **Target Variable Analysis:**
    - Analyze the distribution of actual_productivity to understand its spread.
2. **Feature Relationships:**
    - Use scatter plots and box plots to visualize the relationships between actual_productivity and features like over_time, incentive, and smv.
3. **Correlation Analysis:**
    - Generate a correlation heatmap to see relationships between numerical features and the target variable.
4. **EDA Summary:**
    - Summarize insights from visualizations to guide model selection and inform feature importance.

Week 3: Machine Learning Model Selection and Evaluation

1. **Data Splitting:**
    - Split the data into training and test sets (80% training, 20% testing).
2. **Model Selection and Training:**
    - Train multiple regression models to predict productivity:
        - **Linear Regression:** A basic regression model as a benchmark.
        - **Ridge and Lasso Regression:** Regularized linear models to handle multicollinearity and overfitting.
        - **Random Forest Regressor:** An ensemble model using decision trees for robustness.

        - **Gradient Boosting Regressor:** A boosting model for iterative accuracy improvement.
        - **XGBoost Regressor:** An advanced boosting algorithm with optimizations for speed and performance.
        - **Support Vector Regressor (SVR):** A kernel-based model to capture complex relationships.
3. **Model Evaluation and Visualization:**
    - Evaluate models using regression metrics:
        - **Mean Absolute Error (MAE):** Measures average absolute error.
        - **Mean Squared Error (MSE):** Measures squared average error.
        - **R-squared (R^2):** Proportion of variance explained by the model.
    - **Visualization of Model Performance:**
        - Create a bar chart or box plot comparing the models' MAE, MSE, and R^2 to visualize their effectiveness.
        - Use residual plots to visualize how well each model captures the actual productivity values, revealing any patterns of under- or over-estimation.
4. **Insights from Model Comparison:**
    - Summarize findings from the comparison and visualize key takeaways from the performance plots.

Week 4: Hyperparameter Tuning and Final Report

1. **Hyperparameter Tuning:**
    - Use GridSearchCV or RandomizedSearchCV to tune hyperparameters of the top models (e.g., Random Forest, Gradient Boosting, and XGBoost).
    - Focus on parameters such as n_estimators, learning_rate, max_depth, and alpha (for regularization in Ridge/Lasso).
2. **Final Evaluation and Visualization:**
    - Evaluate the tuned models on the test data and plot the final results.
    - Use a line chart to show performance before and after tuning for the top models.
    - Create a final residual plot for the best-performing model to ensure accurate prediction across the data range.
3. **Predictive System Development:**
    - Develop a predictive system where users can input new data and receive productivity predictions.
4. **Project Reporting:**
    - Prepare a comprehensive report summarizing each stage:
        - **Data Overview:** Summarize the dataset and key features.
        - **Data Cleaning and EDA:** Highlight cleaning, visualization, and findings.
        - **Model Comparison and Tuning:** Include tables, charts, and insights on model performance.
        - **Predictive System:** Describe the final predictive system and its application.

Project Submission:

By the end of the internship, you are expected to submit:

1. **Cleaned dataset** used for modeling.
2. **Python code** for data cleaning, exploration, model building, and tuning.
3. **Final report** summarizing the project, with insights, visualizations, and performance evaluation.
4. **Submit your work:** Submit the document via the provided **Google Form**.