# **Data Science Capstone Project: Manufacturing Equipment Output Prediction with Linear Regression**

## **Problem Statement**

You are working as a data analyst for a manufacturing company that operates injection molding machines to produce plastic components. The company wants to optimize production efficiency by predicting the hourly output (number of parts produced per hour) based on various machine operating parameters. Your task is to build a linear regression model that can predict machine output using factors like temperature, pressure, cycle time, and material properties. This will help the production team optimize machine settings, plan production schedules, and identify when machines are underperforming.

## **Dataset**

**Dataset Source**: Synthetic Manufacturing Dataset (we'll create a realistic dataset based on real manufacturing principles)

* **Alternative**: Industrial IoT datasets from UCI ML Repository or manufacturing datasets from Kaggle
* **Size**: 5,000+ records of hourly machine performance data
* **Target**: Parts produced per hour

**Features included**:

* **Injection\_Temperature**: Molten plastic temperature (°C) - Range: 180-250°C
* **Injection\_Pressure**: Hydraulic pressure (bar) - Range: 80-150 bar
* **Cycle\_Time**: Time per part cycle (seconds) - Range: 15-45 seconds
* **Cooling\_Time**: Part cooling duration (seconds) - Range: 8-20 seconds
* **Material\_Viscosity**: Plastic material flow resistance - Range: 100-400 Pa·s
* **Ambient\_Temperature**: Factory floor temperature (°C) - Range: 18-28°C
* **Machine\_Age**: Equipment age in years - Range: 1-15 years
* **Operator\_Experience**: Operator experience level (months) - Range: 1-120 months
* **Maintenance\_Hours**: Hours since last maintenance - Range: 0-200 hours

## **Project Steps and Goals**

### **Step 1: Data Generation and Loading**

**Goal**: Create a realistic manufacturing dataset with known relationships

**What to do**:

* Generate synthetic manufacturing data with realistic relationships
* Add appropriate noise and correlations between variables
* Save dataset as CSV for reproducibility
* Load and display basic dataset information

### **Step 2: Data Exploration and Understanding**

**Goal**: Understand the manufacturing process and data characteristics

**What to do**:

* Display dataset shape, data types, and summary statistics
* Check for missing values and outliers
* Understand the business meaning of each variable
* Identify the target variable distribution

### **Step 3: Exploratory Data Analysis (EDA)**

**Goal**: Discover patterns and relationships in manufacturing data

**What to do**:

* Create histograms for all variables to understand distributions
* Generate correlation matrix to identify relationships
* Create scatter plots between key operational parameters and output
* Analyze how different machine settings affect production
* Identify optimal operating ranges

### **Step 4: Data Preprocessing and Feature Engineering**

**Goal**: Prepare data for optimal model performance

**What to do**:

* Handle any outliers using IQR method
* Create derived features that might be meaningful (e.g., efficiency ratios)
* Scale features if necessary
* Split data into training and testing sets

### **Step 5: Model Building and Training**

**Goal**: Build and train the linear regression model

**What to do**:

* Create linear regression model
* Train on training data
* Make predictions on test data
* Ensure reproducibility with random states

### **Step 6: Model Evaluation and Performance Analysis**

**Goal**: Comprehensively evaluate model performance

**What to do**:

* Calculate R², MSE, RMSE, and MAE for both training and test sets
* Create residual plots to check model assumptions
* Plot predicted vs actual values
* Analyze model performance across different operating conditions

### **Step 7: Manufacturing Insights and Feature Interpretation**

**Goal**: Extract actionable insights for manufacturing optimization

**What to do**:

* Analyze which parameters most strongly affect output
* Identify optimal operating ranges for key parameters
* Understand the business implications of each coefficient
* Provide recommendations for production optimization

### **Step 8: Production Optimization Recommendations**

**Goal**: Translate model insights into actionable business recommendations

**What to do**:

* Identify the most impactful parameters for optimization
* Suggest machine setting adjustments
* Recommend maintenance and training strategies
* Develop a production monitoring framework

### **Step 9: Model Validation and Business Impact Assessment**

**Goal**: Validate model assumptions and estimate business value

**What to do**:

* Check linear regression assumptions (linearity, independence, homoscedasticity, normality)
* Estimate potential cost savings from optimization
* Identify model limitations and improvement opportunities
* Plan for model deployment and monitoring

## **Expected Outcomes**

By completing this project, students will:

1. **Manufacturing Domain Knowledge**: Understand how operational parameters affect production output
2. **Regression Modeling**: Build and evaluate linear regression models for continuous prediction
3. **Business Analytics**: Translate statistical results into actionable business recommendations
4. **Data Science Pipeline**: Execute complete workflow from data generation to deployment planning
5. **Industrial Application**: Apply data science to real-world manufacturing optimization

## **Success Criteria**

* **Model Performance**: Achieve R² > 0.75 on test data
* **Business Relevance**: Provide clear, actionable optimization recommendations
* **Statistical Rigor**: Properly validate model assumptions and interpret results
* **Documentation**: Clear explanation of methodology and business implications
* **Code Quality**: Clean, well-commented, reproducible analysis

# **Additional Steps: Model Deployment Pipeline**

Here are the additional steps to transform your machine learning model into a complete production-ready application:

## **Step 10: Model Serialization and Persistence**

**Goal**: Save the trained model and preprocessing components for deployment

**What to do**:

* Save the trained model using pickle
* Save any preprocessing objects (scalers, encoders)
* Create a model metadata file with performance metrics
* Implement model loading functions

## **Step 11: FastAPI Backend Development**

**Goal**: Create a REST API to serve model predictions

**What to do**:

* Create FastAPI application with prediction endpoints
* Implement input validation using Pydantic models
* Add health checks and model information endpoints
* Handle errors gracefully

## **Step 12: Docker Configuration**

**Goal**: Containerize the FastAPI application for easy deployment

**What to do**:

* Create Dockerfile for the FastAPI application
* Set up requirements.txt with all dependencies
* Create docker-compose.yml for multi-service deployment
* Add .dockerignore for optimization