SYNOPSIS

TITLE

Bosch Production Failure Detection

1. Title of the Project

Bosch Production Failure Detection is a machine learning system that predicts manufacturing line failures using sensor data from Bosch production processes. The system processes 200 key numeric sensors from 20K+ parts to classify production failures with an XGBoost model achieving AUC 0.646. A Streamlit web app provides live failure probability predictions for maintenance decisions.app.py+1

2. Statement of the Problem

Manufacturing lines experience costly downtime from equipment failures, with Bosch production challenged by imbalanced datasets (1.2% failure rate) and 970+ high-dimensional sensor readings requiring automated detection. Manual quality checks cannot handle real-time prediction across 200 sensors per part. The project addresses this with a predictive classifier enabling proactive maintenance to minimize production losses.Bosch-Production-Failure-Detection_fixed.ipynb

3. Significance of the Project Present State of the Art

Industry 4.0 leverages ML for predictive maintenance, with XGBoost established for sensor-based classification on imbalanced industrial data. Current Bosch operations use reactive maintenance while advanced factories employ real-time sensor analytics. This project advances the field by deploying a production-ready predictor with feature selection (top 200 varying sensors) and interactive dashboard, bridging research to practical manufacturing insights.app.py+1

4. Objectives of the Project

1. Preprocess Bosch production datasets by cleaning and selecting top 200 varying sensors
2. Train XGBoost classifier targeting AUC >0.60 on 200K production parts
3. Validate model performance with 80/20 train/test split and ROC-AUC metrics
4. Export trained model (bosch_model.json) for production deployment
5. Develop Streamlit dashboard for live sensor input and failure predictions
6. Demonstrate high-risk scenarios with sensor readings >150 triggering alertsBosch-Production-Failure-Detection_fixed.ipynb+1

5. Scope of the Project

The project analyzes Bosch's Kaggle production dataset focusing on numeric sensors for binary failure prediction (Response 0/1). Scope includes data cleaning, model training, evaluation, and web deployment using 200K samples × 200 features. Excludes categorical features, real-time streaming, and multi-class failure types for focused implementation with production-ready deliverables.Bosch-Production-Failure-Detection_fixed.ipynb

6. Hardware and Software Specifications

Hardware Requirements

- Processor: Intel i5 or above
- RAM: Minimum 16 GB (200K×200 data processing)
- Storage: 500 GB SSD
- Display: Standard monitor with browser supportBosch-Production-Failure-Detection_fixed.ipynb

Software Requirements

- Operating System: Windows/Linux/MacOS
- Python: 3.12+
- Libraries: Pandas, NumPy, XGBoost, Scikit-learn, Streamlit
- Development Environment: Jupyter Notebook, VS Codeapp.py+1

7. Data Collection and Methodology

Data Collection

Sourced from Kaggle Bosch Production Line Performance dataset containing 200K+ parts with 970 numeric sensors plus Response labels (1.2% positive failures).Bosch-Production-Failure-Detection_fixed.ipynb

Methodology

1. Load train_numeric.csv (200K rows, safe nrows sampling)
2. Fill NaN→0, drop >99% zero variance columns
3. Select top 200 std-dev sensors (L3S29F346, L3S29F347 stations)
4. 80/20 train/test split, fit XGBoost (n_estimators=100)
5. Evaluate AUC 0.646, save bosch_model.json
6. Deploy Streamlit app for 3-sensor live prediction demoapp.py+1

8. Algorithm

1. Start
2. Load train_numeric.csv → 200K rows
3. Clean: drop zero columns → Xclean.csv (200K×200), yclean.csv
4. Train/test split (80/20, random_state=42)
5. Train XGBoost model (n_estimators=100)
6. Evaluate: predict_proba → AUC 0.646

7. Save model → bosch_model.json
8. Deploy Streamlit: live sensor inputs → failure probability
9. Endapp.py+1

9. Limitations Constraints of the Project

- Limited to static Kaggle dataset; no live factory sensor feeds
- Excludes categorical features for numeric focus
- Imbalanced classes (1.2% failures) limit precision/recall
- Fixed 200-sensor selection; full 970+ possible with more compute
- Manual 3-sensor demo; full 200-input scaling needs optimizationapp.py+1

10. Conclusion and Future Scope

Conclusion
The Bosch Production Failure Detection system successfully transforms complex sensor data into actionable maintenance predictions with AUC 0.646. The Streamlit dashboard provides intuitive failure probability display (e.g., >50% triggers MAINTENANCE NOW alerts), demonstrating practical Industry 4.0 predictive maintenance.Bosch-Production-Failure-Detection_fixed.ipynb+1

Future Scope

- Hyperparameter optimization/ensemble models for AUC >0.70
- Include categorical features and full 970-sensor dataset
- Real-time IoT sensor integration via APIs
- Multi-class failure type prediction
- Cloud deployment (AWS/Heroku) with auto-scalingBosch-Production-Failure-Detection_fixed.ipynb

11. References Bibliography

1. Kaggle: Bosch Production Line Performance Dataset (2016)
2. XGBoost: eXtreme Gradient Boosting Documentation
3. Streamlit: Deploy ML Apps in Minutes
4. Scikit-learn: ROC-AUC Metrics Documentation