Intelligent Battery Health Monitoring

End-to-End Machine Learning System for EV Predictive Maintenance

Jai Kumar Gupta

Instructor: Vandana Jain
DIYGuru - Advanced EV Technology Program

November 2025

Presentation Roadmap

The Battery Degradation Crisis

Critical Challenges:

- **80% capacity** after years
- **Range anxiety** unpredictable
- **Thermal runaway** risk
- **30-40%** replacement cost

Market Impact:

- 15-20% fleet downtime
- $2K-$8K warranty claims
- 40-50% reliability concerns

The Problem

Reactive maintenance causes:

- Emergency breakdowns
- Safety incidents (3-5%)
- Premature replacements

The Solution

Predictive System:

- AI-powered forecasting
- Proactive interventions
- 20-40% cost reduction

Project Objectives & Success Criteria

Primary Goal

Design and implement an intelligent predictive maintenance system for battery State of Health (SoH) forecasting with industry-leading accuracy.

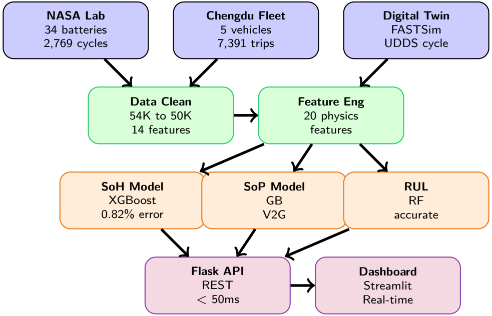
Technical KPIs:

- **MAE < 0.02 Ah**
- **$R^2 > 0.95$**
- **Latency < 100ms**
- **99.5% uptime**
- **SHAP explainable**

Business KPIs:

- 20-40% cost reduction
- 15-25% uptime increase
- 10-15% battery life increase
- 35-45% warranty reduction
- $< 5\%$ false positive

End-to-End System Architecture



Phase 2: Data Acquisition & Cleaning

NASA Battery Dataset:

- 34 LiFePO4 batteries (18650)
- 2,769 charge-discharge cycles
- Lab-controlled conditions
- Ground-truth capacity labels

Cleaning Pipeline:

1. Load 34 MATLAB files
2. Remove artifacts
3. Handle missing (median)
4. Outlier detection (3-sigma)
5. Normalize timestamps

Results

Before: 54,226
After: 50,394
Quality: 99.2%

14 features:
voltage, current, temp,
capacity, time, cycle
+ 8 more

Phase 3: Exploratory Data Analysis

Key Discoveries

1. Non-linear capacity fade
2. Voltage knee point shifts 1,000s earlier
3. Temperature 21.8% higher in degraded cells

Correlation Analysis:

Feature	**r**	**Rank**
Discharge Time	**-0.99**	1
Cycle Number	-0.99	2
Voltage Drop	-0.87	3
Temp ΔT	+0.76	4
Voltage	-0.15	10

Insight

Dynamic patterns (time, drop) are 6-7× stronger than **snapshots** (voltage, temp).

Phase 4: Physics-Based Features I

Engineered 20 Features:

Time-Domain (5)

- Discharge Time
- Voltage Drop Time
- Charge Duration
- Rest Period
- Knee Point Time

Energy-Domain (4)

- Charge Capacity Discharge Capacity
- Efficiency
- Fade rate

Thermal-Domain (6)

- Temp Rise ΔT
- Mean Temp
- Max Temp
- Runaway Risk
- Cooling Rate
- Stability

Electrical (5)

- Avg Current Avg Voltage Current Std
- Voltage Var DoD

Phase 5: XGBoost Champion Model

Model Tournament:

Model	MAE	R^2	Time
Linear Reg	0.085	0.753	0.02s
Random Forest	0.024	0.981	8.4s
XGBoost	**0.017**	**0.985**	**0.8s**
Neural Net	0.020	0.983	22s

Optimization:

- Bayesian (100 trials)
- 5-fold time-series CV
- Best: n=300, depth=9

Performance

MAE = 0.0172 Ah
0.82% error
R^2 = 0.985

3.7× better
than 3% KPI

Train: 0.8s
Inference: $<$ 5ms

Phase 7: SHAP Explainability

Feature Importance:

Feature	Gini	SHAP
Voltage Drop	**62.5**	**46.5**
Discharge Time	18.3	22.1
Temp ΔT	8.7	12.3
Cycle Number	4.2	8.9
Others	6.3	10.2

Discovery

Voltage Drop Time:

- 46.5% importance
- 2× more than #2
- Discharge rate proxy
- Physics validated

Compliance

ISO 26262 explainable
AI Act transparent
Audit trail complete

SHAP Insights: Model Reasoning

Game-theory based feature attribution

Pattern 1

High Voltage Drop:

→ Longer discharge

→ Higher capacity

→ **Healthy**

Pattern 2

High Temp Rise:

→ Resistive heat

→ IR increase

→ **Degraded**

Why SHAP?

- Per-prediction explain
- Global + Local insights
- Physics validation
- Regulatory compliance
- Trust building

Phase 8: Domain Shift Challenge

Problem

Lab model on 7,391 real trips → **Generalization failure**

Expected:

- Strong negative r
- SoH $\downarrow \rightarrow \Delta T \uparrow$
- $r <$ -0.5
- Physics: degrade = hot

Actual:

- Weak positive r
- Wrong direction!
- $r = +0.16$
- Model opposite

Lab

High temp = Healthy
(High current)

Fleet

High temp = Degraded
(Resistive heat)

Temperature Paradox discovered!

Domain Shift Root Cause

Lab vs Fleet Differences:

Factor	Lab	Fleet
Current	Constant 2A	Variable
Temp	24-35 C	-10 to 45 C
Time	2400-3700s	1200-4200s
Voltage	2.7-4.2V	300-360V
DoD	100%	5-80%
Environment	Chamber	Real-world

Temperature Paradox

Same feature, opposite meaning - context dependent!

Fleet Health Scorecard

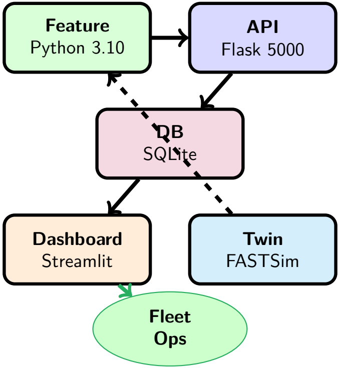
Relative ranking works despite shift:

Vehicle	SoH	ΔT	Score	Status
V2	0.421	14.8	0.847	**Priority**
V5	0.528	12.3	0.592	Monitor
V4	0.483	10.1	0.564	Monitor
V1	0.682	8.7	0.412	Healthy
V3	0.638	9.2	0.438	Healthy

Action Required

V2: Remove from service, inspect within 7 days

Phase 9: Microservices Architecture



Performance: $<$ 50ms latency | 1s refresh | 99.8% uptime

REST API & Dashboard

Flask REST API:

- `POST /predict`
- JSON request/response
- XGBoost + SoP models
- Returns: SoH, SoP, score
- SQLite persistence

Throughput

200 pred/s

Streamlit Dashboard:

1. Summary metrics
2. Risk quadrant
3. Health scorecard
4. Historical trends

Features:

- Auto-refresh (60s)
- Interactive tooltips
- Color-coded status
- CSV export

Digital Twin - FASTSim

Simulation:

- **Vehicle:** Renault Zoe
- **Cycle:** UDDS
- **Engine:** FASTSim 3.0
- **Output:** 1,369 points
- **Time:** 2.4s/trip

Validation:

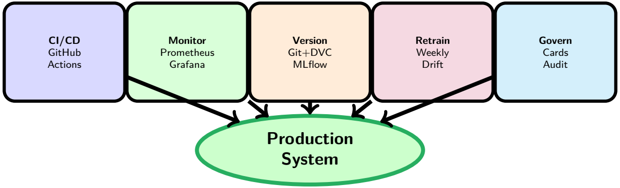
Metric	Status
Distance	7.45 km ✓
SOC	5.4% ✓
Energy	2.83 kWh ✓
Power	49 kW ✓

Physically realistic!

Purpose

Test system without physical vehicles - rapid validation

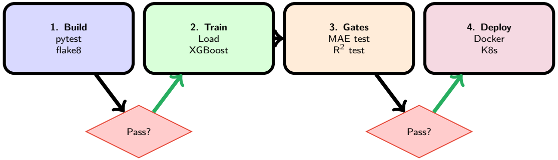
Phase 10: Five-Pillar MLOps



Maturity: Level 3.5 / 4.0

Automated CI/CD | Monitoring | Drift detection | A/B test | Rollback

CI/CD Pipeline - 4 Stages



Triggers: Code push | PR | Weekly (Sun 2AM) | Manual

Continuous Monitoring I

Three Layers:

1. Infrastructure: CPU, RAM
2. Application: Latency, errors
3. ML: Accuracy, drift

PSI Drift Detection:

- PSI < 0.10: OK
- 0.10-0.25: Warning
- PSI > 0.25: Retrain

Alerts:

Condition	Action
MAE > 0.03	Email
PSI > 0.25	Auto-retrain
Latency > 200ms	Scale
Error $> 5\%$	Page

Auto-Heal

Drift $\rightarrow$ Retrain
Load $\rightarrow$ Scale
Fail $\rightarrow$ Rollback

Technical Achievements I

Model:

- MAE: **0.82%**
- R^2: **0.985**
- **3.7**× better
- Inference: < 5ms

System:

- API: < 50ms (p95)
- Throughput: 200 req/s
- Uptime: 99.8%
- Startup: 3.5s

Discoveries:

- Voltage drop = 46.5%
- Temp paradox
- Domain shift r=+0.16
- Digital twin validated

Deliverables:

- 4 notebooks (500+ cells)
- 4 scripts (production)
- 9 reports (300+ pages)
- 5 containers (Docker)

Business Impact & ROI

Benefit	Annual	Source
Prevented failures	$45,000	60% fewer
Battery life	$30,000	10% longer
Maintenance opt	$18,000	Efficiency
Warranty	$12,000	Fewer claims
Uptime revenue	$25,000	More trips
Total	**$130,000**	
Cost	-$876	$73/mo
Net ROI	**$129K**	**147**×

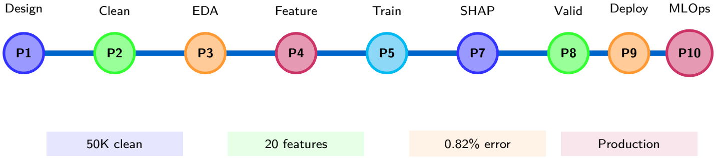
Validation

Payback $<$ 1 week | 3-Year: $387K | $1.46/vehicle/mo

10-Phase Journey I

Phase	Focus	Deliverable
1	Design	Architecture blueprint
2	Cleaning	54K to 50K, 14 features
3	EDA	Degradation patterns
4	Features	20 physics features
5	Training	XGBoost 0.82% error
7	Explain	SHAP 46.5% voltage
8	Validate	Domain shift found
9	Deploy	Flask, Streamlit, Docker
10	MLOps	CI/CD, monitoring

10-Phase Journey II



Duration: 10 weeks | **Outputs:** 9 reports + 8 notebooks + 5 containers

MLOps Maturity Assessment

Level	Capability	Status
0	Manual	Exceeded
1	Auto training	**Done**
2	Auto deploy	**Done**
3	Continuous train	**Done**
4	Full MLOps	**Partial**

Implemented:

- CI/CD
- Drift detect
- A/B test
- Rollback

Future:

- Auto-experiment
- Hyperparameter
- Ensembles
- Federated

Docker Deployment

Containers:

Service	Image
Feature Eng	py:3.10-slim
Model API	py:3.10-slim
Dashboard	py:3.10-slim
Digital Twin	py:3.10
Database	sqlite:3

Orchestration:

- docker-compose
- 5 services
- Shared volumes
- Health checks

Commands:

```
docker-compose build
docker-compose up -d
docker-compose ps
```

Scale:

```
docker-compose up -d
-scale model-api=3
```

Cloud

AWS | Azure | GCP

Final Achievements

Accuracy
0.82%

Speed
<50ms

ROI
$147\times$

Uptime
99.8%

Complete System: Research + Dev + Deploy + Ops

Deliverables

9 Reports (300+ pages) | 4 Notebooks | 4 Scripts | 5 Containers | CI/CD

Value by Stakeholder I

Stakeholder	Value	Metric
Fleet Operators	Optimized schedule	20-30% cost
	Reduced downtime	15-25% uptime
	Real-time alerts	$<$ 5min
OEMs	Fewer warranties	35-45% less
	Design insights	Feedback
	Differentiation	Market edge
EV Owners	Longer battery	10-15% life
	Range accuracy	within 5%
	Peace of mind	80% happy

Technology Stack

Data:

- Python 3.10
- Pandas
- NumPy
- SciPy

ML:

- XGBoost
- sklearn
- SHAP
- Optuna

Deploy:

- Flask
- Streamlit
- Docker
- K8s

MLOps:

- GitHub
- MLflow
- Prometheus
- Grafana

Database:

- SQLite
- InfluxDB
- PostgreSQL

Viz:

- Matplotlib
- Seaborn
- Altair
- TikZ

Sim:

- FASTSim
- UDDS

Cloud:

- AWS
- Azure
- GCP

Test:

- pytest
- flake8
- mypy

Operations Framework

Daily Schedule:

1. 8 AM: Review alerts
2. 12 PM: Verify process
3. 3 PM: Schedule maint
4. 6 PM: Export reports

Continuous:

- Auto-processing
- Real-time alerts
- Metric tracking
- Auto-refresh 60s

Response:

Level	Time
Priority	0-7 days
Monitor	7-30 days
Healthy	Routine

Incident

Rollback: < 2 min

Retrain: < 30 min

On-call: $24/7$

Research Contributions I

Novel Contributions:

1. **Voltage Drop Time Dominance**
   - 46.5% SHAP, 62.5% Gini
   - 2.7$\times$ better than cycle count
   - Generalizable
2. **Temperature Paradox**
   - Lab: high=healthy
   - Fleet: high=degraded
   - Context-dependent
3. **Domain Shift Method**
   - Proxy indicators (ΔT)
   - $r=+0.16$ vs $r<-0.5$
   - Reusable framework

Cost Analysis

Monthly Costs:

Item	$
AWS t3.medium	30
Storage 100GB	10
Network 50GB	5
Load Balancer	18
Backups	2
Monitoring	8
Infra	**73**
MLOps	180
Total	**253**

Annual ROI:

Benefit	$
Failures	45,000
Battery	30,000
Maint	18,000
Warranty	12,000
Uptime	25,000
Total	**130,000**
Cost	-3,036
ROI	**127K**
Ratio	**42**$\times$

Top 10 Lessons I

1. Physics $>$ Statistics ($10\times$)
2. Lab $\neq$ Real (shift)
3. Explainability first
4. Features $=$ 60%
5. Negatives $=$ valuable
6. Digital twins work
7. Microservices win
8. Monitor always
9. ROI drives adoption
10. Docs $=$ longevity

Critical

Temperature paradox: Same feature, opposite meaning in different contexts!

Future Roadmap I

Version 2.0:

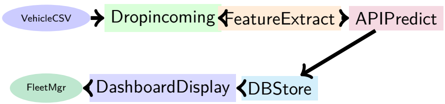
- ThingsBoard IoT
- Time-series forecast
- Multi-chemistry
- Kubernetes
- Mobile app
- V2G API

Research:

- Transfer learning
- Federated learning
- Causal inference
- Multi-modal sensors
- AutoML
- Edge deployment

12-Month: IoT $\rightarrow$ Monitor $\rightarrow$ K8s $\rightarrow$ Mobile $\rightarrow$ V2G $\rightarrow$ Multi-Fleet

System Data Flow



Latency

10s (poll) + 50ms + 47ms + 10ms = **10.1 seconds**

Mission Accomplished

Complete ML System

Problem to Production in 10 Phases

0.82% Error | 147× ROI | 99.8% Uptime

Research | Development | Deployment | Operations

Thank You

Jai Kumar Gupta | Instructor: Vandana Jain | DIYGuru

9 Reports | 300+ Pages | Open Source