 **Predict early signs of battery failure** using historical sensor and operational data.

 **Reduce maintenance costs** by flagging potential issues before complete failure.

 **Evaluate performance** of classification and anomaly detection models.

 **Visualize insights** to aid decision-making using dashboards (optional: Power BI/Tableau).

**🔧 Key Tasks / Works to be Done:**

**1. Data Collection & Preprocessing**

* Source or simulate battery data (features: temperature, voltage, cycle count, internal resistance, SoC, SoH, etc.).
* Handle missing values, outliers, and normalize/standardize features.
* Create binary target variable: 1 = failure, 0 = normal.

**2. Exploratory Data Analysis (EDA)**

* Correlation between features and failure.
* Trend analysis across charge/discharge cycles.
* Distribution plots, heatmaps, and time series visualization.

**✅ 1. Baseline Models (Start Simple)**

* **Logistic Regression**: Easy to interpret, good baseline.
* **Decision Tree**: Simple yet visual and interpretable.
* **Random Forest**: Handles non-linearity, good feature importance insights.

**✅ 2. Advanced Tree-Based Models**

* **XGBoost / LightGBM / CatBoost**:
  + Faster and more accurate than Random Forest in many cases.
  + Handles missing values and imbalanced data well.
  + Use **SHAP values** to explain predictions.

**✅ 3. Anomaly Detection Models (for unsupervised or rare failure cases)**

* **Isolation Forest**: Efficient for large data.
* **One-Class SVM**: Detects outliers in high-dimensional space.
* **Autoencoders (Neural Networks)**:
  + Trained to reconstruct normal behavior; large reconstruction error signals anomaly.
  + Can be used for time-series sensor data.

**✅ 4. Time-Series Models (if data is time-dependent)**

* **LSTM (Long Short-Term Memory)**:
  + Ideal for battery data with temporal sequence (cycle-by-cycle analysis).
  + Can capture gradual degradation patterns.
* **Prophet (for forecasting SoH or failure probability)**:
  + Facebook’s tool for time-series forecasting.

**✅ 5. Ensemble or Hybrid Approach**

* Combine predictions from multiple models:
  + **Stacking**: Use a meta-model to combine base models.
  + **Voting Classifier**: Majority voting across multiple algorithms.

**✅ 6. Imbalanced Data Handling**

* Battery failures are rare; use:
  + **SMOTE** (Synthetic Minority Over-sampling Technique)
  + **Class weighting**
  + **Anomaly-focused metrics**: Precision-Recall, F1-score, ROC-AUC

**4. Hyperparameter Tuning**

* Use GridSearchCV or RandomizedSearchCV to optimize models.

**5. Deployment (Optional but impressive)**

* Create an API using Flask or FastAPI for model predictions.
* Dockerize the solution or deploy to a cloud platform (AWS/GCP/Azure).

**6. Visualization Dashboard (Optional)**

* Power BI:
  + Failure prediction heatmap by location/time.
  + Distribution of risk score per battery.
  + Cycle life and failure pattern visualizations.