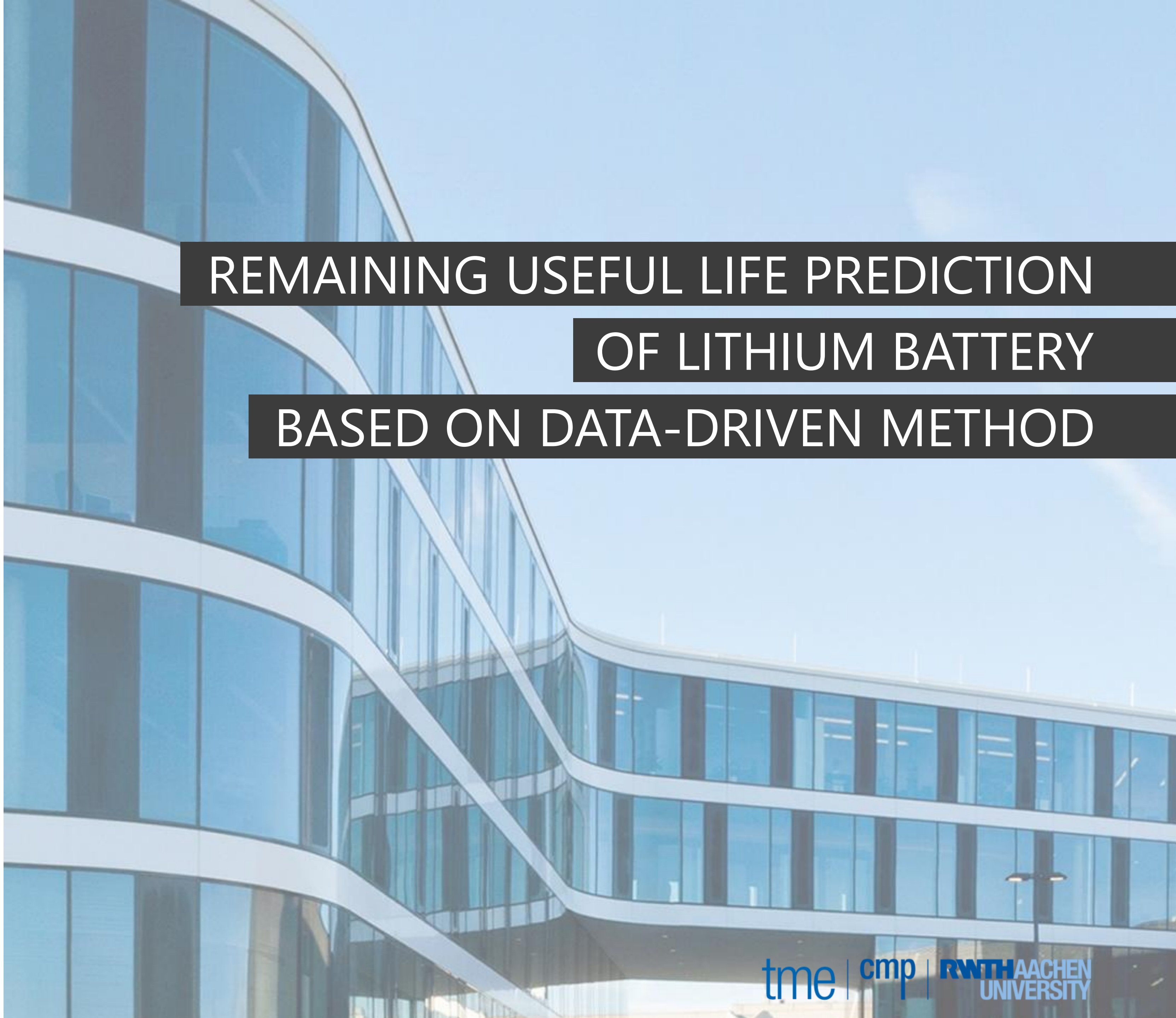
# REMAINING USEFUL LIFE PREDICTION OF LITHIUM BATTERY BASED ON DATA-DRIVEN METHOD

11.09.2025, AACHEN
YUZHI ZHENG

INSTITUTE FOR THERMODYNAMICS OF MOBILE
ENERGY CONVERSION SYSTEMS &
CENTER FOR MOBILE PROPULSION



SUPERVISOR:

PROF. DR.-ING STEFAN PISCHINGER
M.SC WENBIN LI

THE FINAL REPORT OF MASTER THESIS

# AGENDA

CHAPTER 1 INTRODUCTION

CHAPTER 2 THEORY & DATA FOUNDATIONS

CHAPTER 3 METHODOLOGY

CHAPTER 4 EXPERIMENTS & RESULTS

CHAPTER 5 CONCLUSION

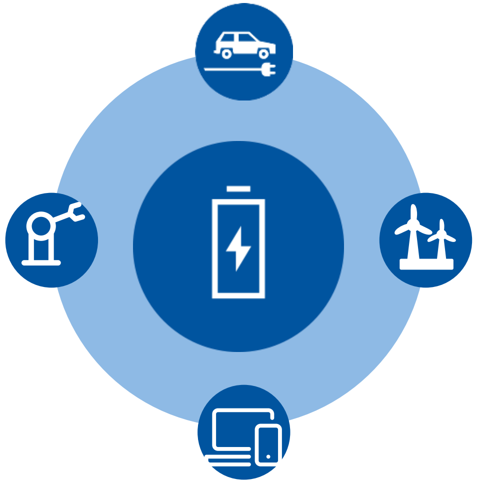
CHAPTER 1

# Introduction

- 1-1 Background & Motivation
- 1-2 Basic Concepts
- 1-3 The Main Methods of RUL Prediction
- 1-4 Our Main Focus
- 1-5 Our Contributions

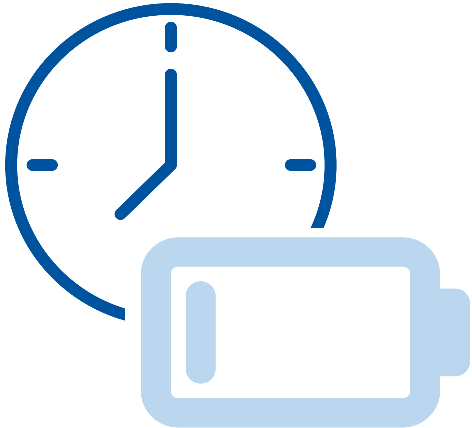
# Chapter 1: Introduction

## 1-1 BACKGROUND & MOTIVATION



*"Battery– The Core Pillar of Industry"*

Powering EVs, New Energy, Electronics, and Robotics



"*RUL*– Empowering Battery Health"

Enabling Early Maintenance and Risk Prevention

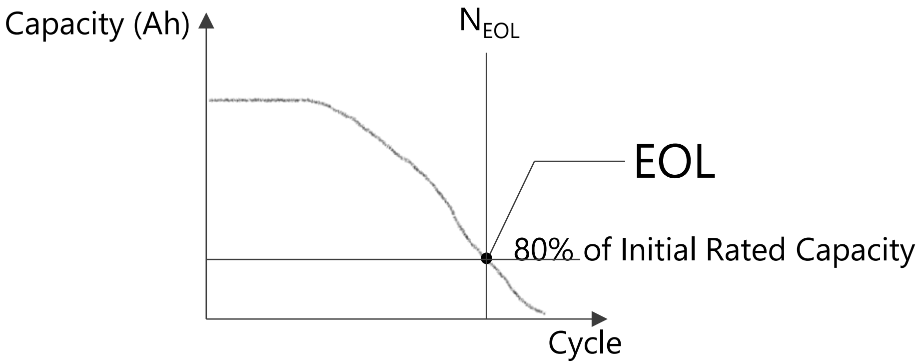
RUL: Remaining Useful Life; EVs: Electric Vehicles

# Chapter 1: Introduction

## 1-2 BASIC CONCEPTS

**EOL**

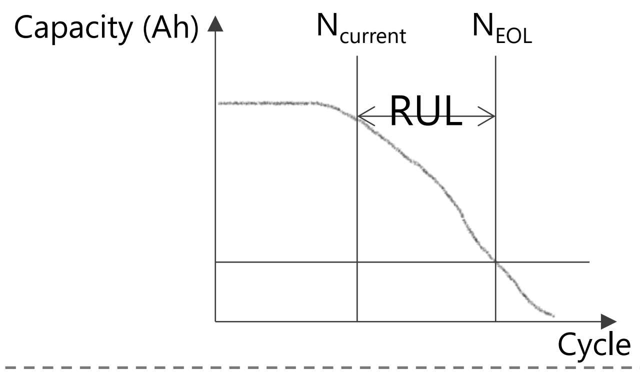
End of life[1)]




$EOL=0.8*Q_{\text{Initial rated capacity}}$

**RUL**

Remaining Useful Life[1)]

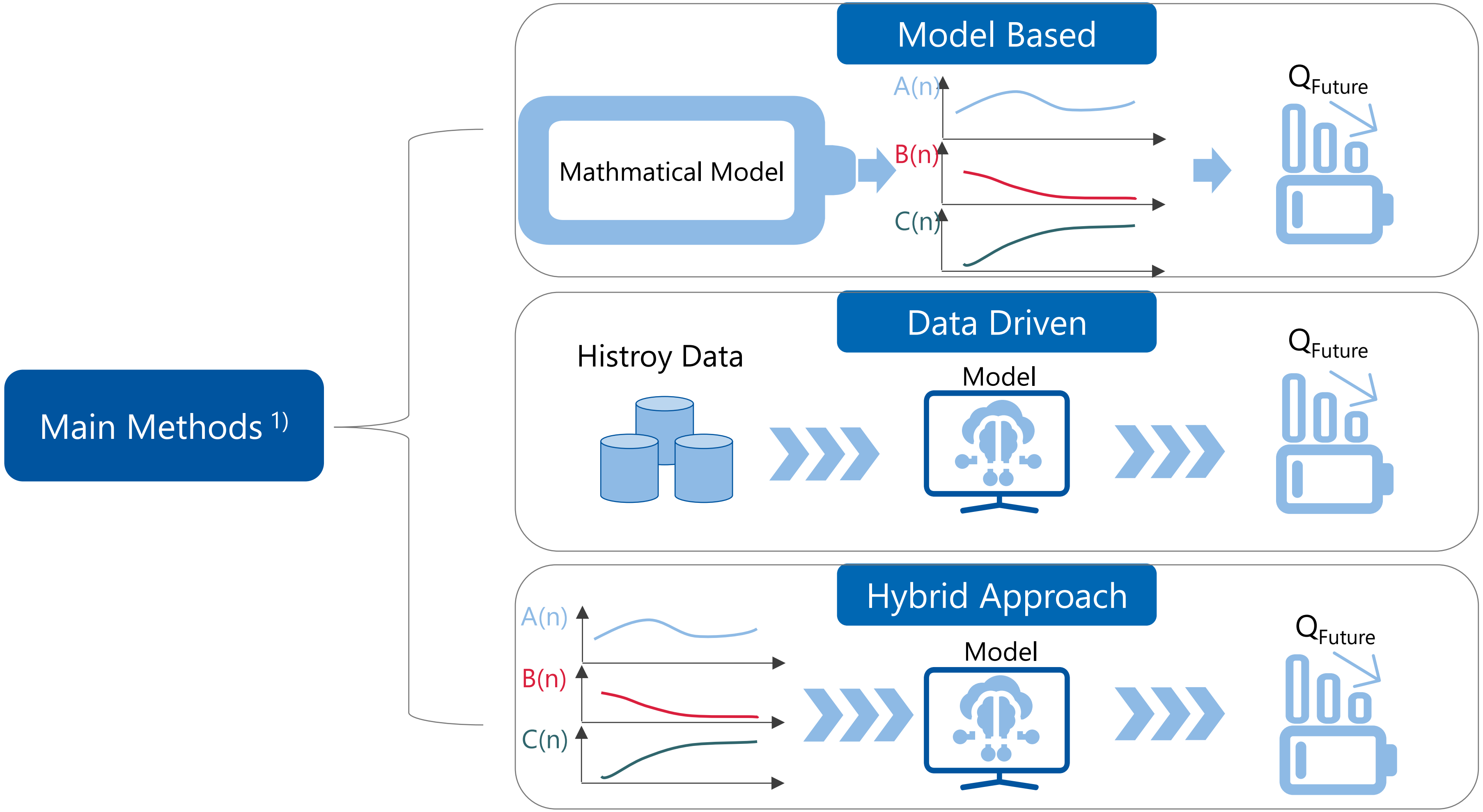



$RUL=N_{Current} - N_{EOL}$

Source: 1) Predicting the Future Capacity and Remaining Useful Life of Lithium-Ion Batteries Based on Deep Transfer Learning

# Chapter 1: Introduction

## 1-3 THE MAIN METHODS OF RUL PREDICTION



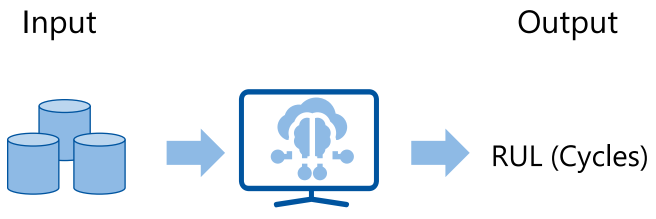

Source: 1) Battery Lifetime Prognostics

# Chapter 1: Introduction

## 1-4 OUR MAIN FOCUS: DATA DRIVEN

### 1 DIERECT RUL PREDICTION[1)]

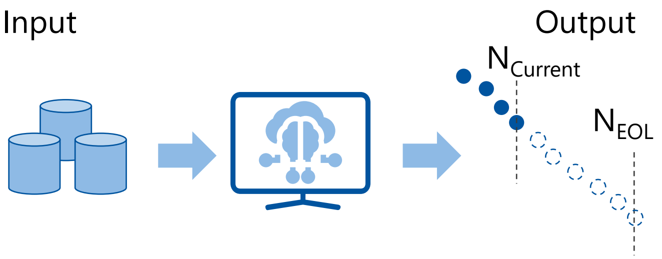
**Direct RUL Model**




- Direct RUL estimation
- No complete trajectory

### 2 ONE-SHOT PREDICITON[2)]

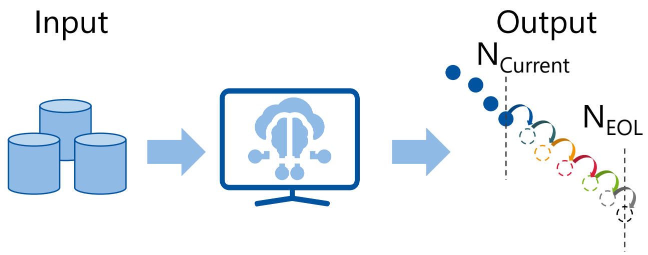
**One-Shot Model**



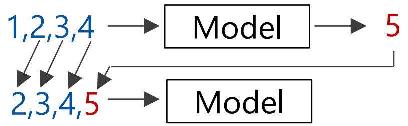

- Trajectory-based prediction
- Limited prediction accuracy

### 3 ITERATIVE PREDICTION[3)]

**Iterative Model**




- Sliding window prediction




- Error accumulation

Error → Error → Error → Error

Source: 1) Data-driven prediction of battery cycle life before capacity degradation; 2) One-shot battery degradation trajectory prediction with deep learning; 3) RUL prediction of Li batteries based on wavelet denoising and transformer neural network

# Chapter 1: Introduction

## 1-5 OUR CONTRIBUTIONS

1. A Novel „Twin" Model Iterative Prediction Framework

Train

1.1 A Sampling Strategy

1.2 Pesudo-Battery Transfer Learning

Training Strategy

2. A Signal-Denosing Pipeline with DWT Technologie

DWT: Discrect Wavelet Transformation

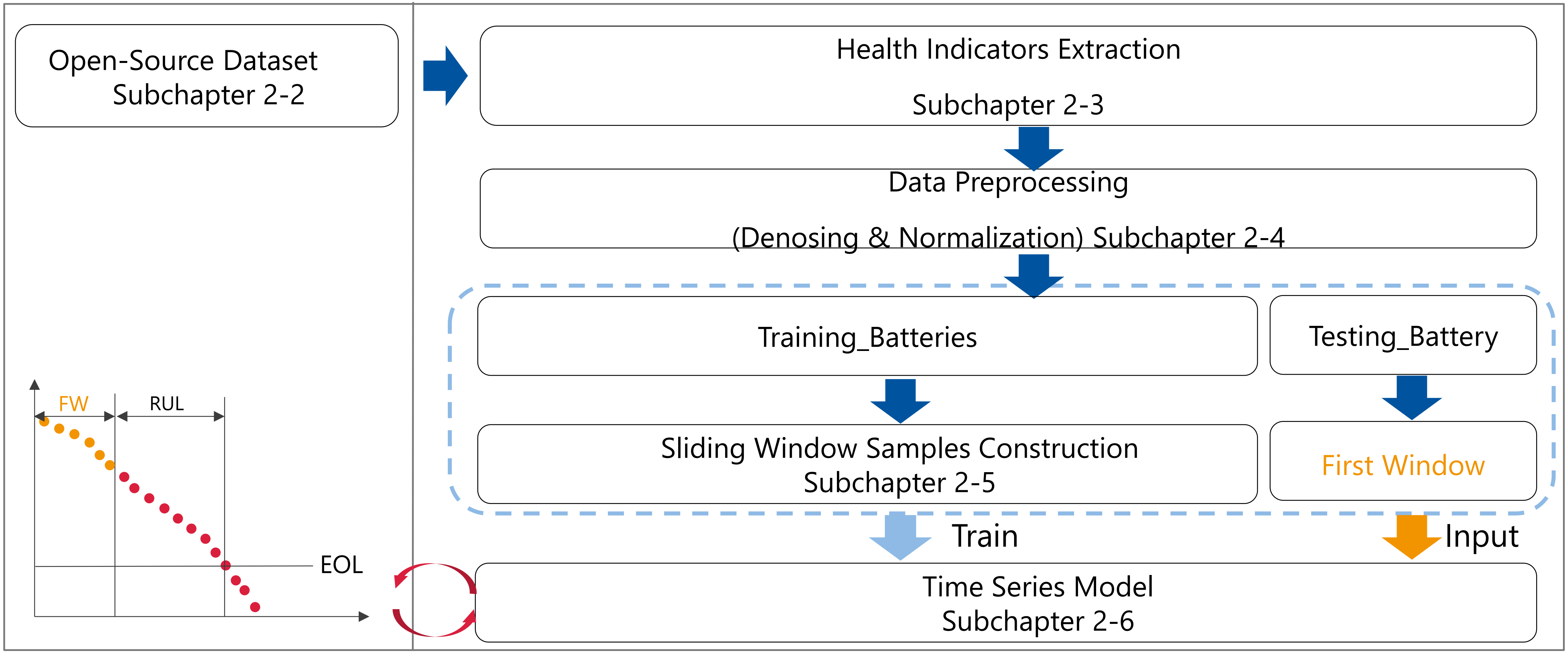
CHAPTER 2

# Theory & Data Foundations

- 2-1 The Workflow Of RUL Predicition
- 2-2 The Battery Dataset & Operating Conditions
- 2-3 Health Indicators
- 2-4 Data Preprocessing
- 2-5 Sliding Window Construction
- 2-6 Time Series Modeling

# Chapter 2: Theory & Data Foundations

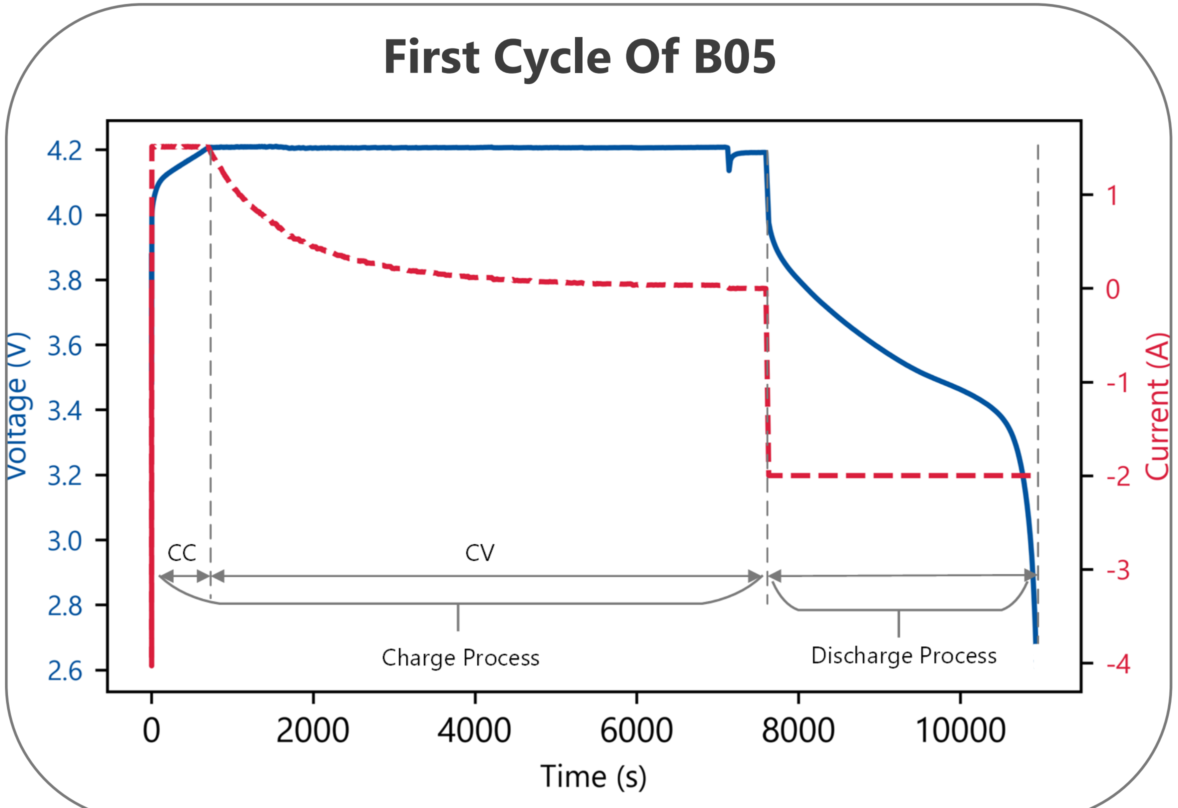
## 2-1 THE WORKFLOW OF RUL PREDICTION




FW: First Window from test battery

# Chapter 2: Theory & Data Foundations

## 2-2 BATTERY DATASET & OPERATING CONDITIONS




### NASA DATASET FY08Q4

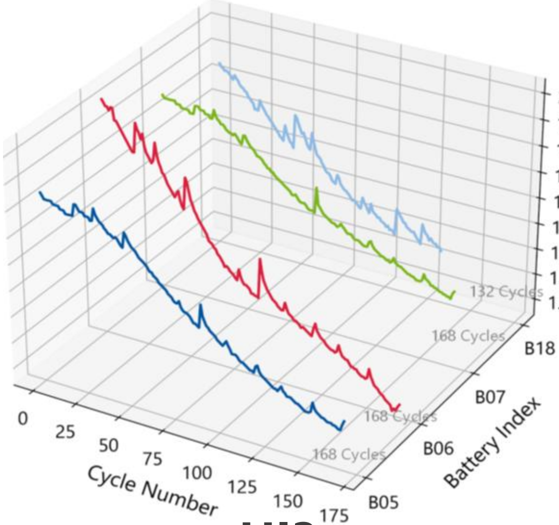
- 4 Cells: B05/06/07/18
- Each cycle: charge & discharge profile
  - Charge profile: CC+CV modul
    - Voltage; Current; Tempreture; Time
  - Discharge profile:
    - Same as charge + Discharge capacity
- Charge operating conditions: Identical
- Discharge Operating Conditions: Different
  - Discharge Cut-Off Voltage
    - B05 (Test battery): 2.7V
    - B06/07/18 (Train batteries): 2.5V, 2.2V, 2.5V

CC: Constant Current; CV: Constant Voltage

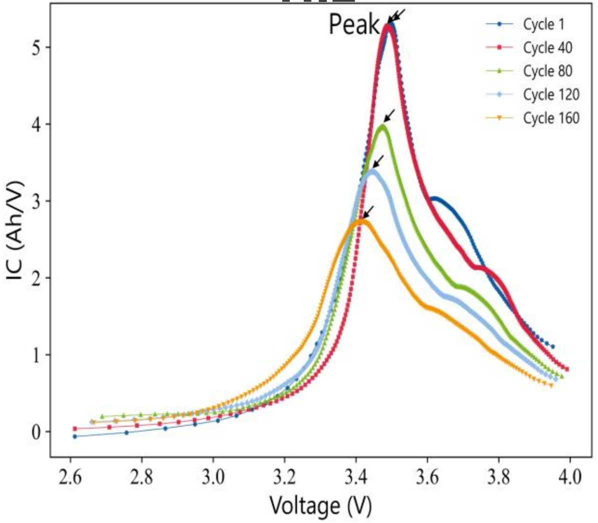
# Chapter 2: Theory & Data Foundations

## 2-3 HEALTH INDICATORS

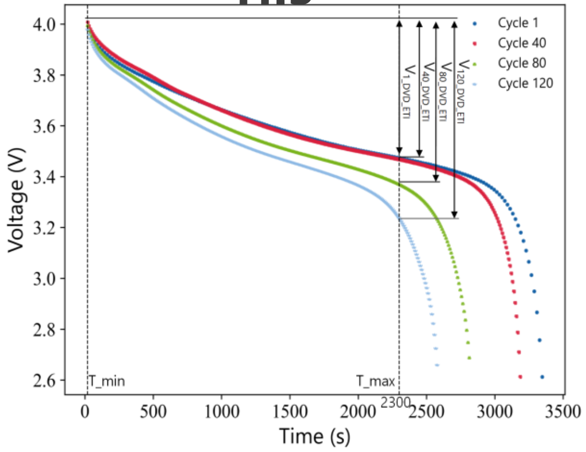
**HI1**



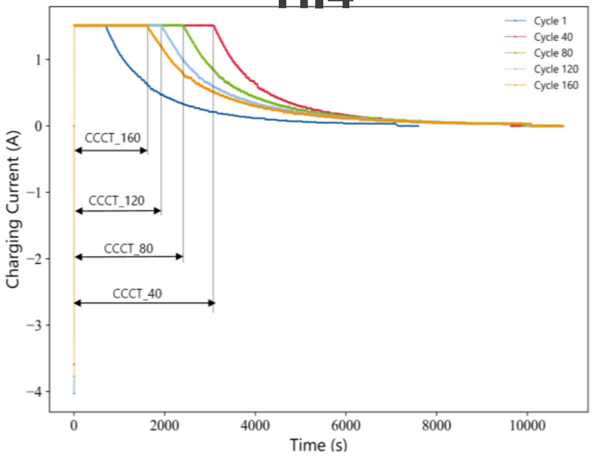

**HI2**




**HI3**




**HI4**




### FOUR HEALTH INDICATORS

- Directly related to aging
  - **HI1**: Capacity
- Indirectly related to aging
  - **HI2**: Peak Point of Incremental Capacity Curve
  - **HI3**: Voltage Difference of Equal Time Interval
  - **HI4**: Constant Current Charging Time

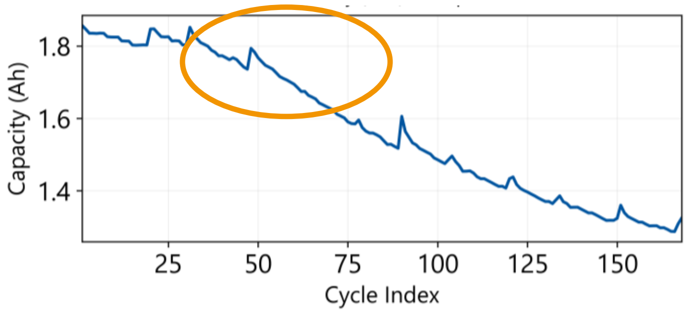
Extracted HI from discharge profile

Extracted HI from charge profile

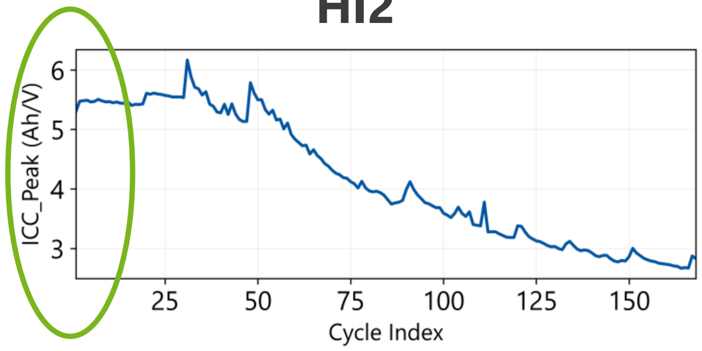
# Chapter 2: Theory & Data Foundations

## 2-3 HEALTH INDICATORS

**HI1**

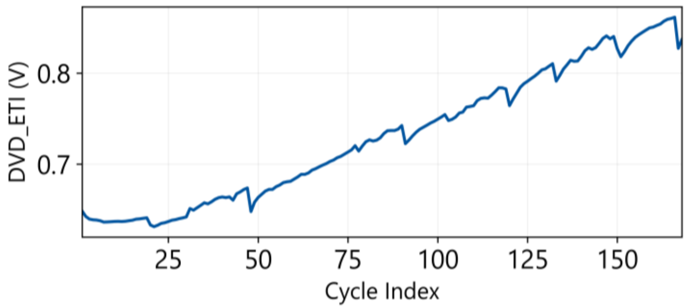



**HI2**



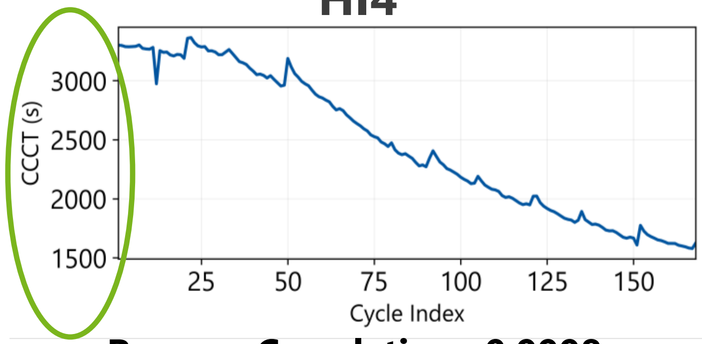

**Pearson Correlation: 0.9960**

**HI3**




**Pearson Correlation: -0.9933**

**HI4**




**Pearson Correlation: 0.9998**

## CORRELATION ANALYSIS

- Pearson correlation
  - Assess representativeness of HI2/3/4 for battery aging
  - All exhibit strong correlation

$$r = \frac{\sum_1^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_1^n (x_i - \bar{x})^2 \sum_1^n (y_i - \bar{y})^2}}$$

## PROBLEMS

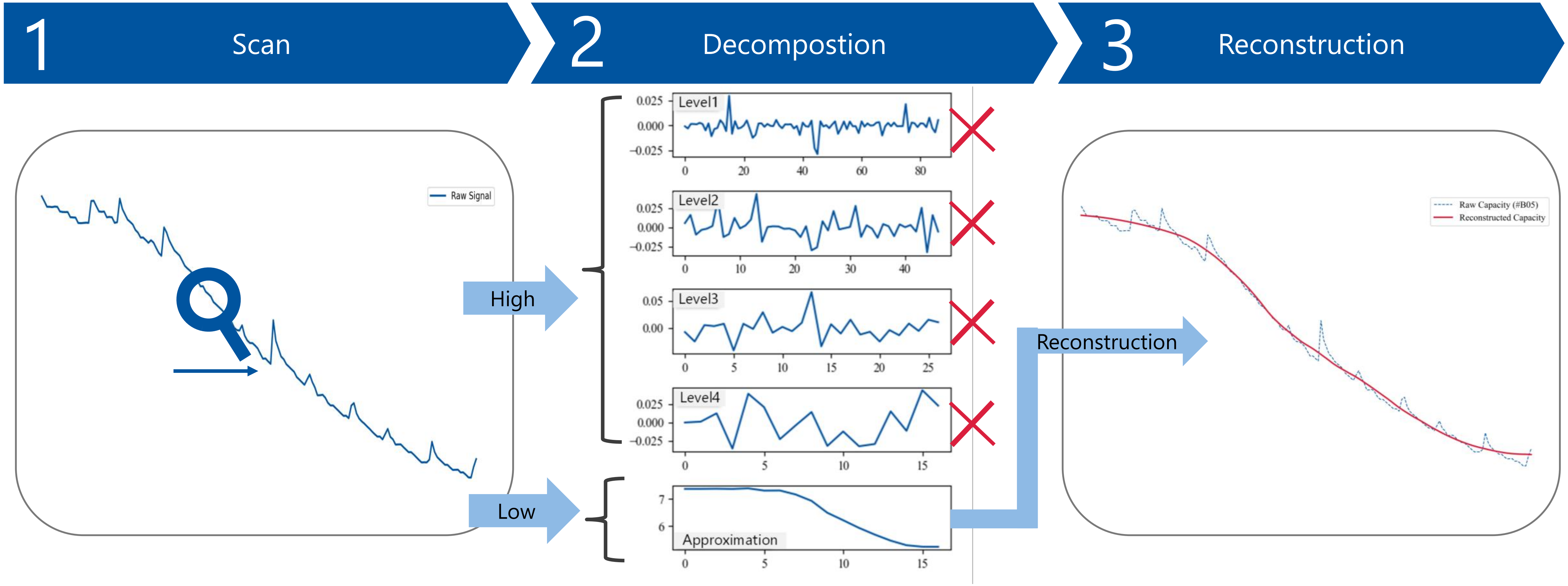
- Problem1: Noise and regeneration
  - Solution: DWT denoise
- Problem 2: Different scales
  - Solution: Min-Max normalization

DWT: Discrete Wavelet Transformation; HI2: Peak Point of Incremental Capacity Curve (ICC_Peak); HI3: Discharge Voltage Difference of Equal Time Interval (DVD_ETI); HI4: Constant Current Charging Time (CCCT)

# Chapter 2: Theory & Data Foundations

## 2-3 DATA PREPROCESSING

The pipeline of Discrete Wavelet Transformation (DWT)

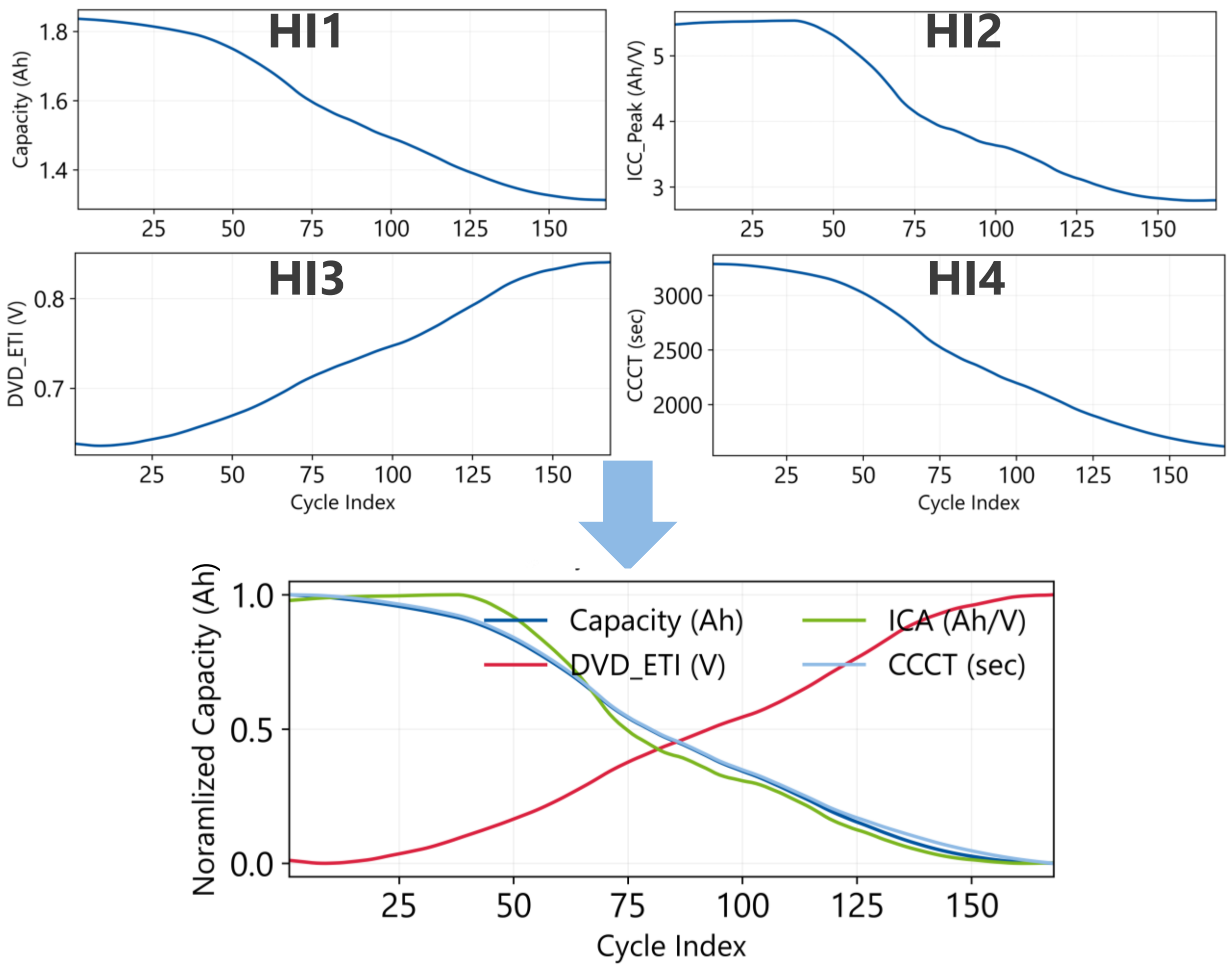





tme | cmp | RWTHAACHEN UNIVERSITY

# Chapter 2: Theory & Data Foundations

## 2-3 DATA PREPROCESSING




### MIN-MAX NORAMLIZATION

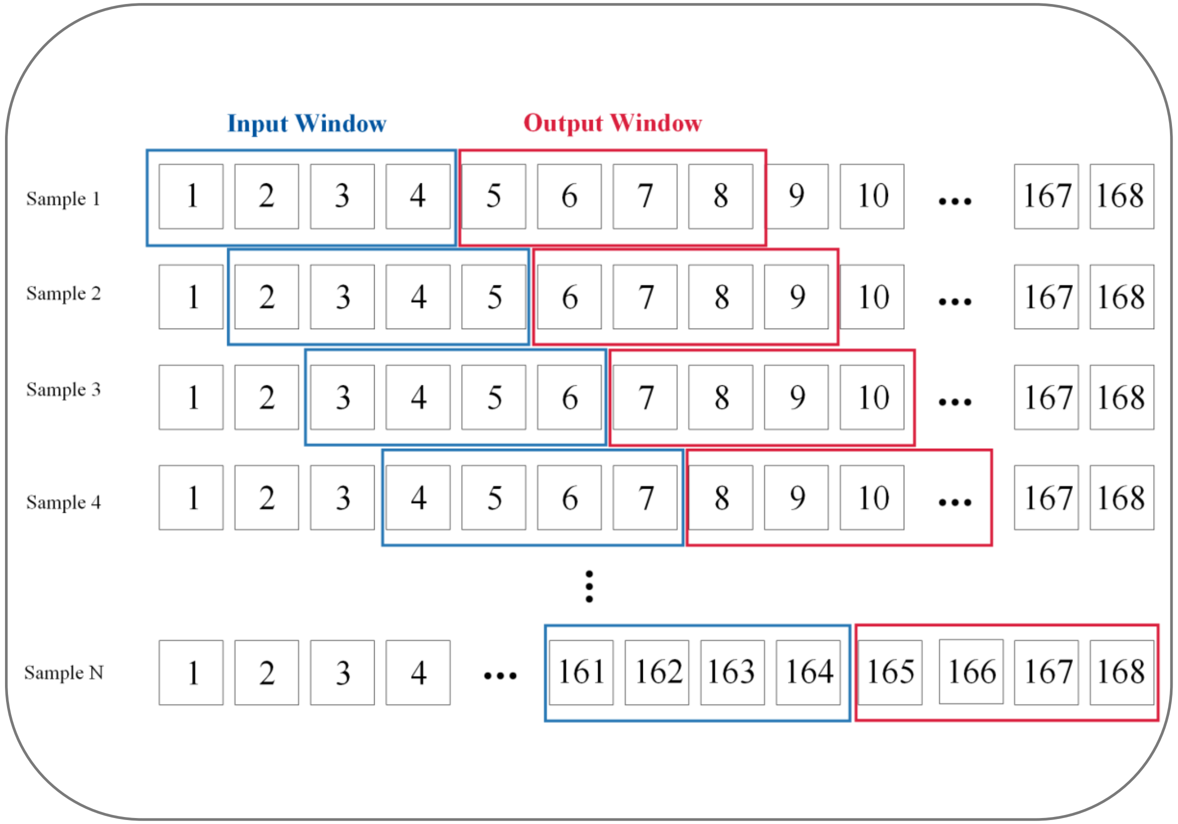
- Formel:

$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

- Purposes:
  - Scale into [0,1]
  - Accelerate convergence
  - Prevent numerical instability

# Chapter 2: Theory & Data Foundations

## 2-4 SLIDING WINDOW SAMPLES CONSTRUCTION

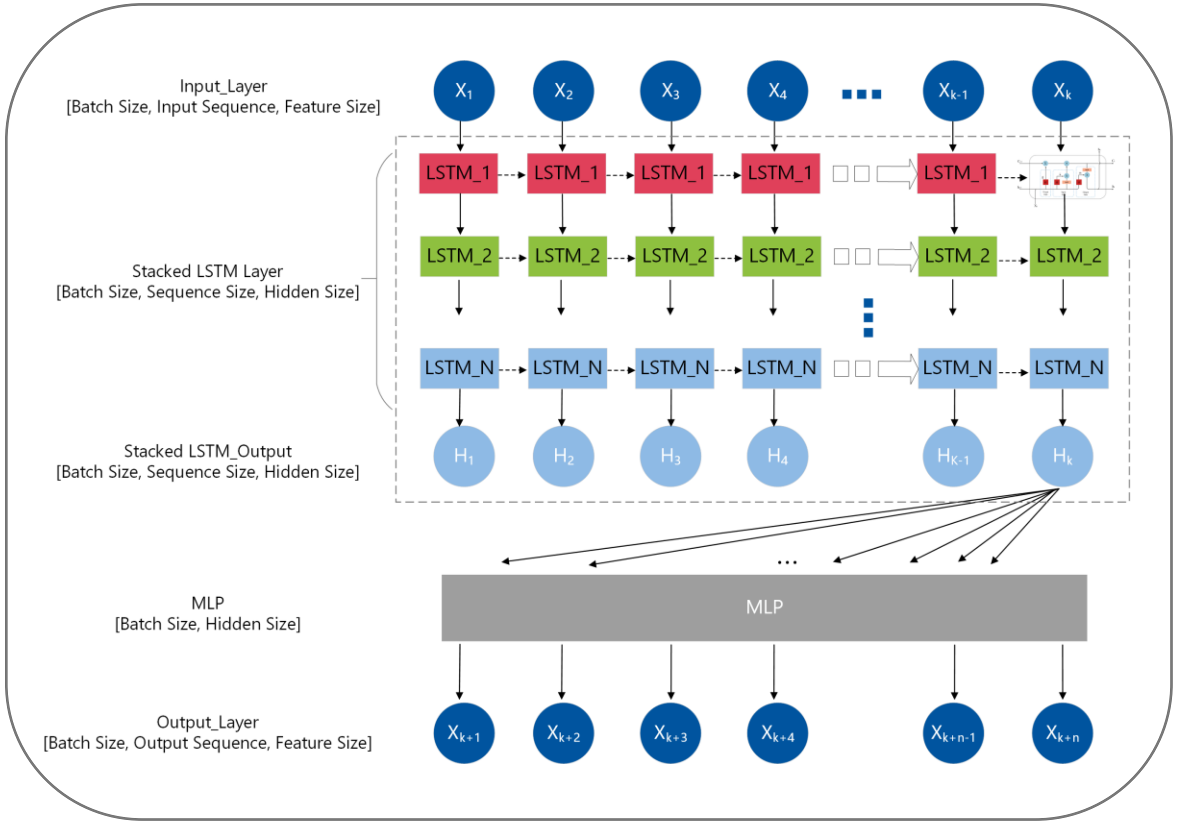



### SLIDING WINDOW

- What is the sliding window?
  - Input Window= Model Input
  - Output Window= Model Label
- Why build sliding window?
  - Break long sequences into short
  - Capture local degradation patterns
  - Improve data utilization

# Chapter 2: Theory & Data Foundations

## 2-5 TIME-SERIES MODEL




### TIME-SERIES MODEL

- Why LSTM?
  - Control information flow through three gates
  - Capture sequential dependencies
- Why Stacked LSTM?
  - Extract deeper features
  - Learn more complex prediction task

MLP: Multi-Layer Preceptron; LSTM: Long Short-Term Memory

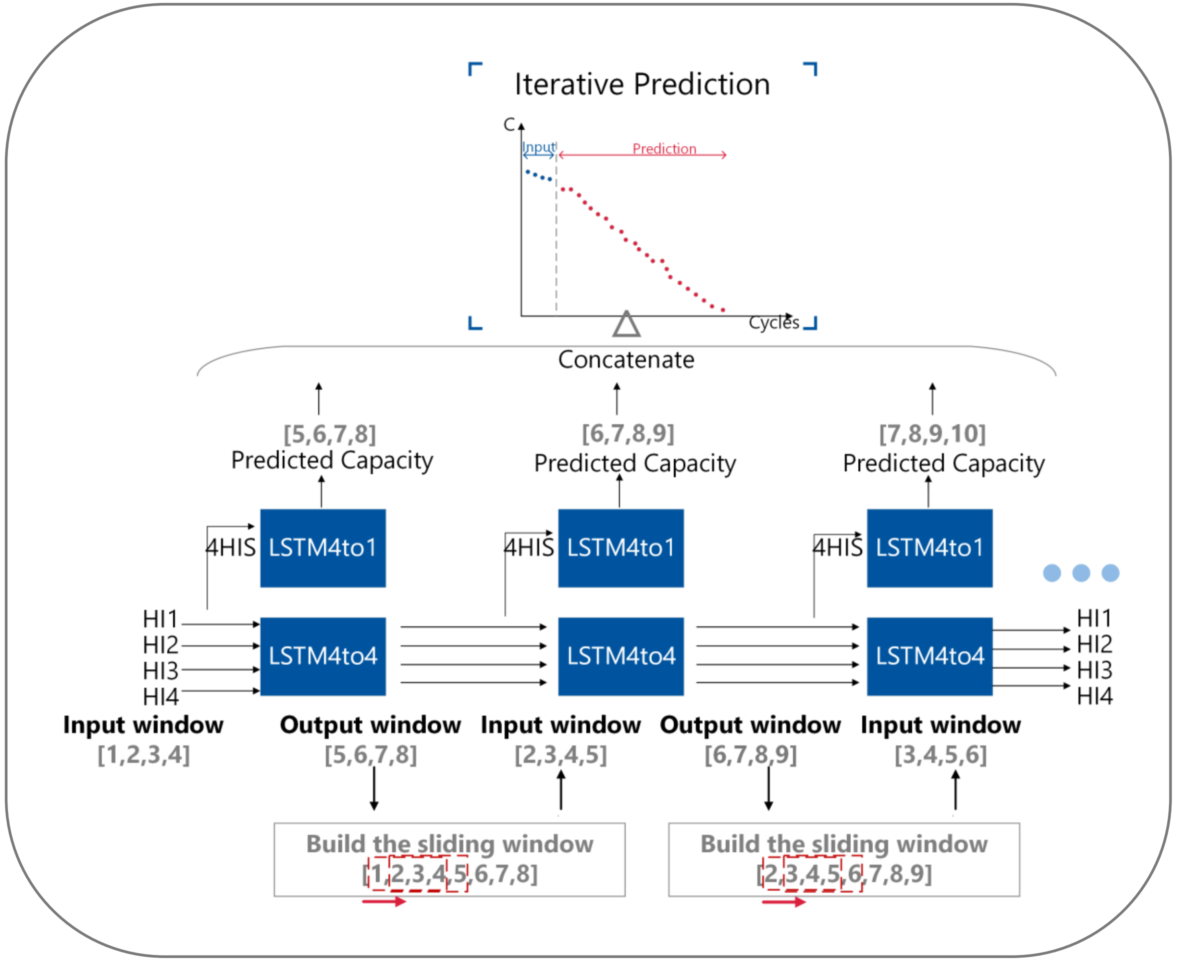
CHAPTER 3

# Methodology

- 3-1 Twin Model Prediction Framework
- 3-2 Our Sampling Traininig Stategy
- 3-3 Pesudo-Battery Transfer Learning
- 3-4 Metrics

# Chapter 3: Methodology

## 3-1 TWIN MODEL PREIDICTION FRAMEWORK

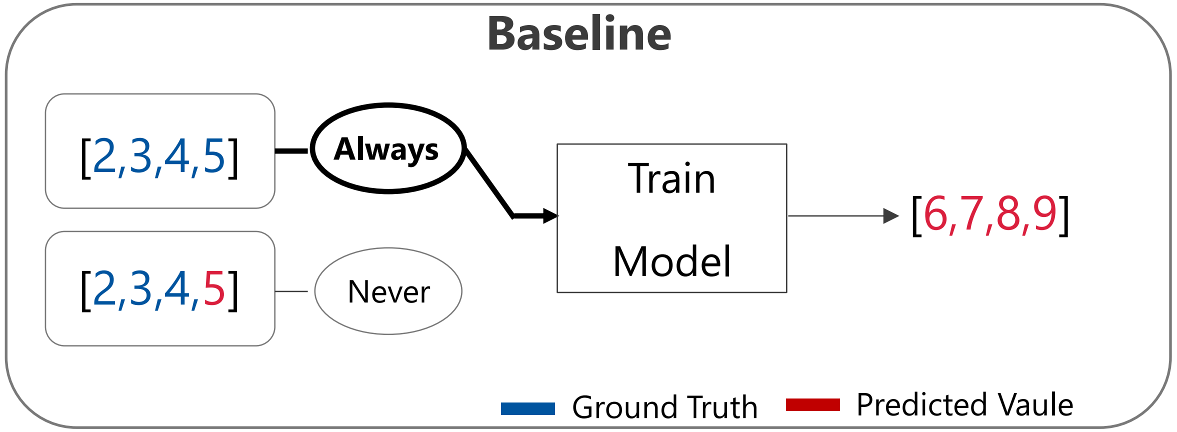


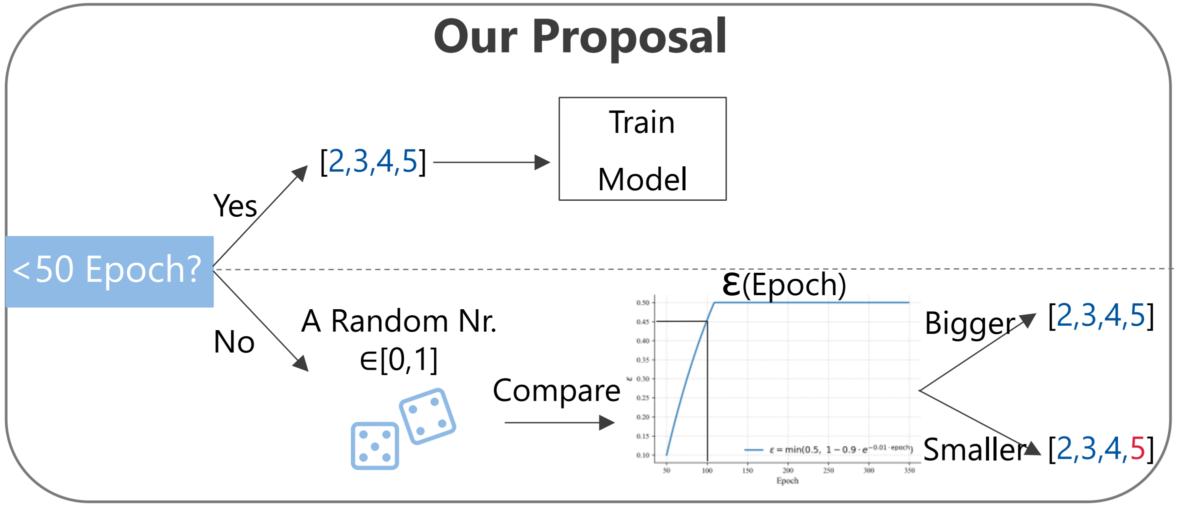

### TWIN MODEL STRUTURE

- LSTM4to4
  - 4 HIs input; Output 4 predicted HIs
  - **Challenge: Error Accumulation** Error → Error → Error → Error
  - Solution: Sampling strategy (Chapter 3-2)
- LSTM4to1
  - 4 HIs input; Output predicted capacity
  - **Challenge: Different Operating Conditions**
  - Solution: Transfer learning (Chapter 3-3)

# Chapter 3: Methodology

## 3-2 OUR SAMPLING TRAININIG STATEGY




### BASELINE

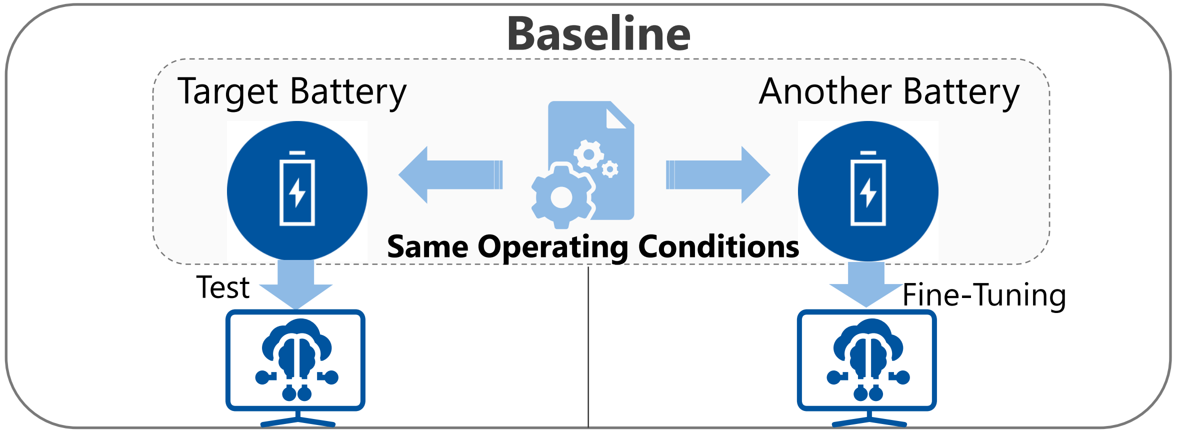
- Adopt always ground truth to train the model
  - Good stability
  - Problem: Conflict with real-world scenario

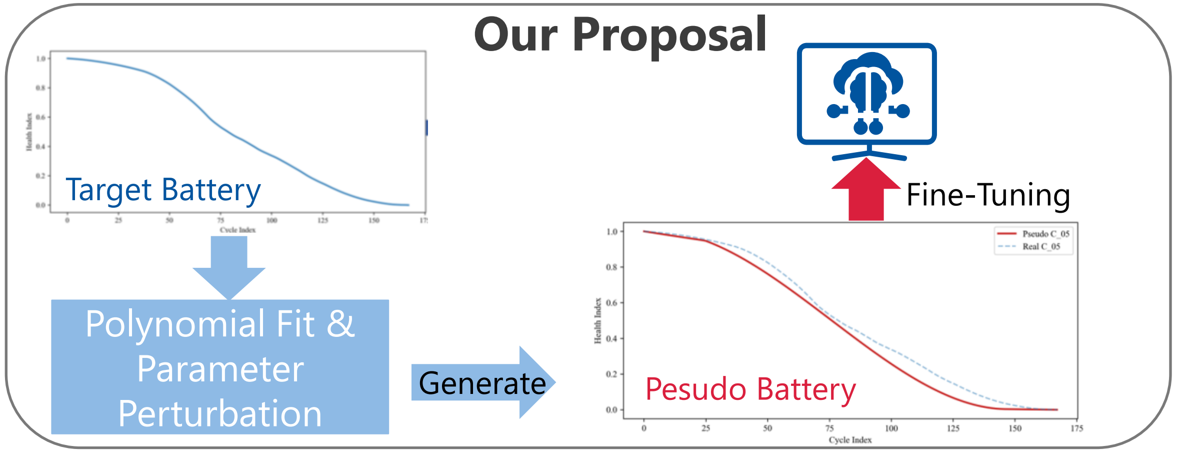
### OUR PROPOSAL

- Sampling with probability ε
- ε is related to epoch
- Aim: Add self-predicted values gradually into training

# Chapter 3: Methodology

## 3-3 PESUDO BATTERY TRANSFORER LEARNING




### BASELINE

- Adopt another battery with same operating conditions
  - Problem: Lack of data

### OUR PROPOSAL

- Use target battery to generate pseudo battery
  - Pseduo battery can represent same operating conditions

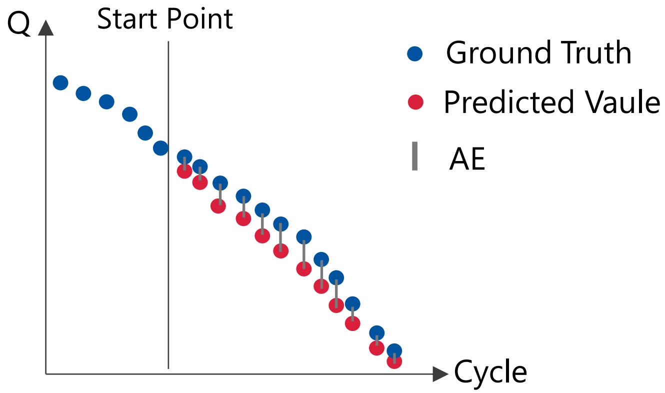
# Chapter 3: Methodology

## 3-4 METRICS

### 1 Absoult Error (AE)

- AE: To evaluate the predicition performance in every cycle

$$AE = |y_i - \hat{y}_i|$$




### 2 Mean Absolute Error (MAE)

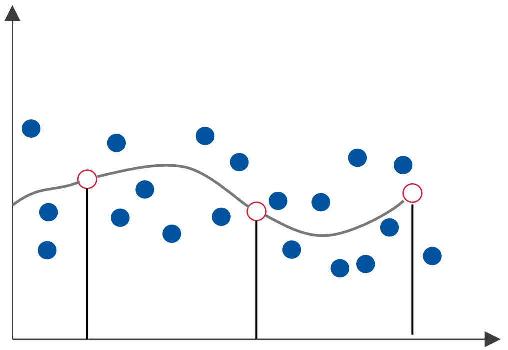
- MAE: To evaluate general predicition performance in all cycles

$$MAE = \frac{1}{n} * \sum_{1}^{n} |y_i - \hat{y}_i|$$

### 3 R Square ($R^2$)

- $R^2$ : To evaluate the generalization ability of the model
- $R^2$ more closer to 1, the better the model learn the degradation pattern.

$$R^2 = 1 - \frac{\sum_1^n (y_i - \hat{y}_i)^2}{\sum_1^n (y_i - \bar{y})^2}$$



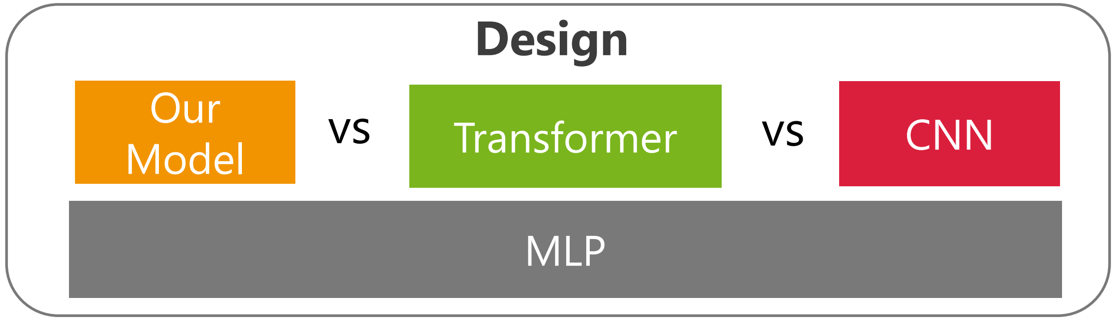
CHAPTER 4

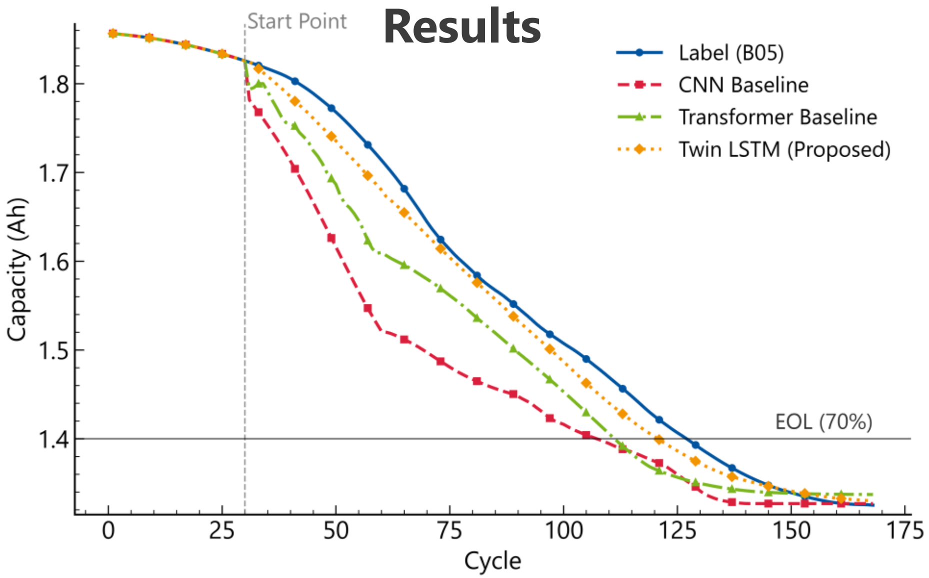
# Experiments & Results

- 4-1 The Comparison of Models
- 4-2 Effect of Varying Start Points
- 4-3 Different Fine-Tuning Data Proportions
- 4-4 Ablation Experiments

# Chapter 4: Experiments & Results

## 4-1 MODEL COMPARSION




### DESIGN

- Different encoder as baselines
- Compare best pefermence

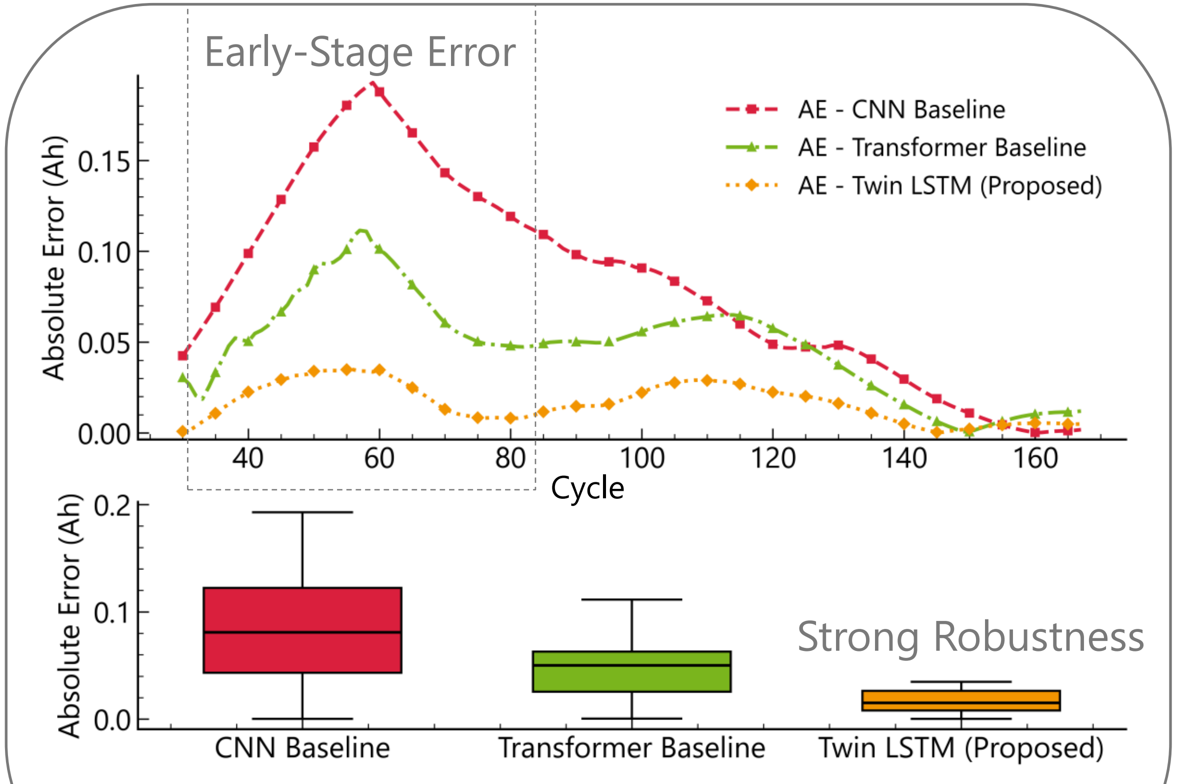
### RESULTS

- Capacity Prediction
  - Our Model: High performence
  - Transformer: Acceptable performence
  - CNN: Unacceptable performence
- RUL Prediction
  - Real RUL: 98 Cycles
  - Our Model: 92 cycles -> 6% (Relative Error)
  - Transformer: 81 cycles -> 17%
  - CNN: 79 cycles -> 19%

CNN: Convolutional Neural Network

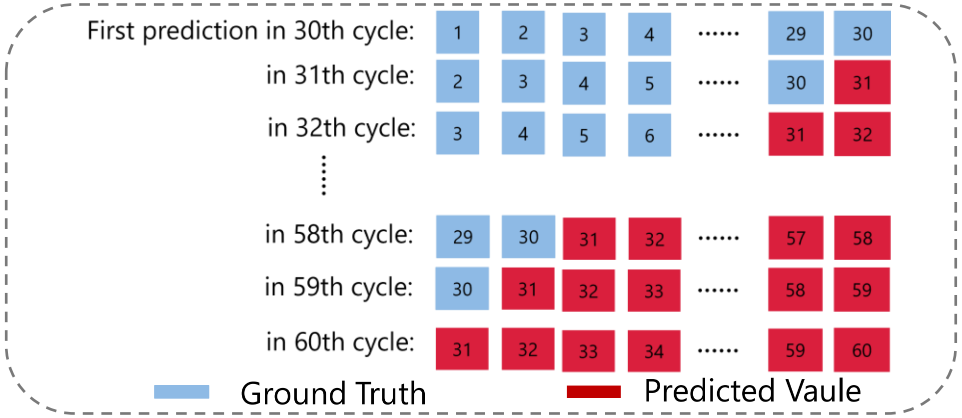
# Chapter 4: Experiments & Results

## 4-1 MODEL COMPARSION




## DISCUSSION

- Effect of different input distributions

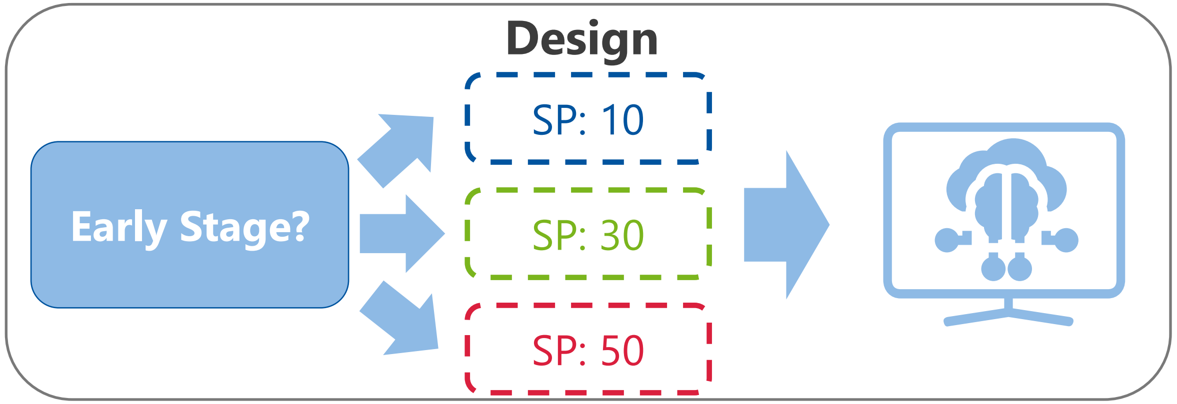


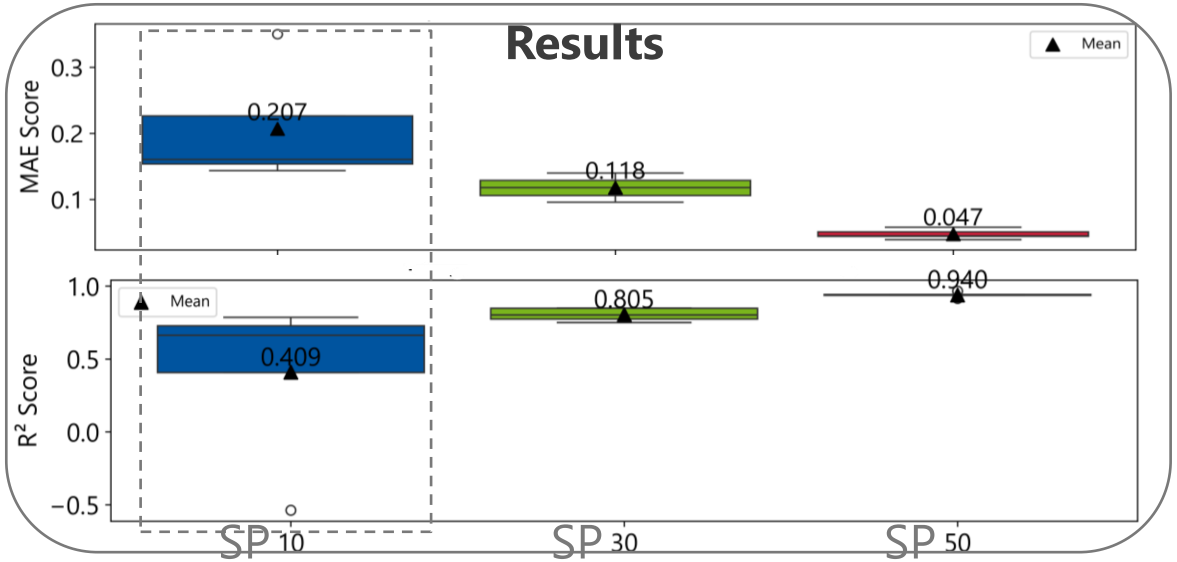

## RESULTS

- Our model exhibits strong Robustness

# Chapter 4: Experiments & Results

## 4-2 VARYING START POINTS




### DESIGN

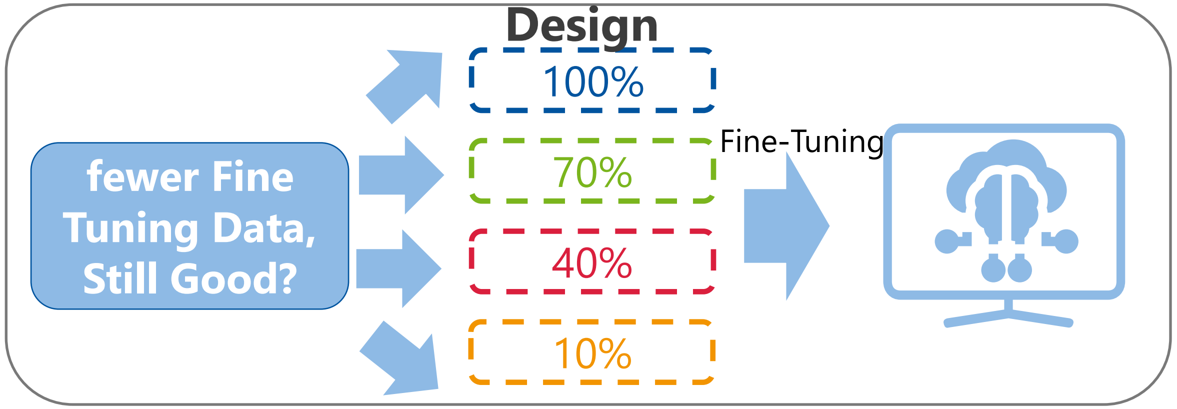
- Try different Start Points (SP)
- Each run 5 times and compare MAE and $R^2$

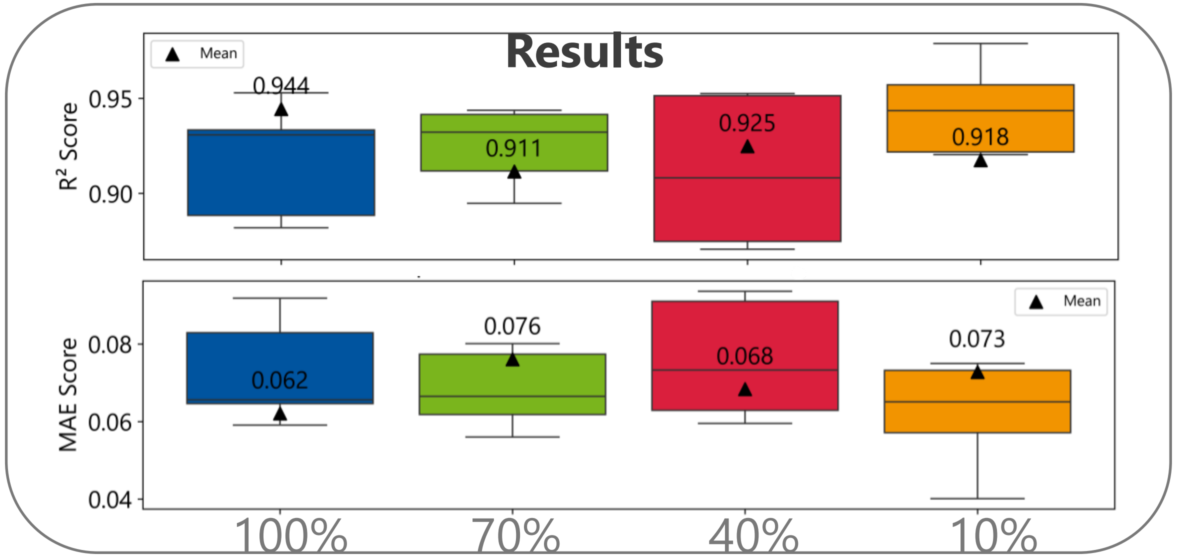
### RESULTS

- Our model has good early-stage prediction capability

# Chapter 4: Experiments & Results

## 4-3 DIFFERENT FINE-TUNING DATA PROPORTIONS




### DESIGN

- Try different fine-tuning data proportions
- Each run 5 times and compare MAE and $R^2$

### RESULTS

- No significant difference of mean values
- Effective fine-tuning can with only 10%

# Chapter 4: Experiments & Results

## 4-4 ABLATION EXPERIMENTS

**Design**

| | DWT | Stacked | Sampling | Transfer Learning |
|---|---|---|---|---|
| Type1 | ✗ | ● | ● | ✗ |
| Type2 | ● | ✗ | ● | ✗ |
| Type3 | ● | ● | ✗ | ✗ |
| Type4 | ● | ● | ● | ✗ |

**Results**

| | 1st Run | 2nd Run | 3rd Run | 4th Run | 5th Run | Mean $R^2$ |
|---|---|---|---|---|---|---|
| Type1 | 0.9518 | 0.8125 | 0.7153 | 0.9135 | -1.3776 | 0.4031 |
| Type2 | 0.3248 | 0.9524 | 0.3102 | 0.9359 | 0.9249 | 0.6896 |
| Type3 | 0.9069 | 0.7588 | 0.8224 | 0.9340 | 0.8506 | 0.8546 |
| Type4 | 0.9559 | 0.8782 | 0.9220 | 0.9364 | 0.9231 | 0.9232 |

### DESIGN

- Different 4 types
  - Type 1: Without DWT denoise
  - Type 2: Without stacked LSTM
  - Type 3: Without sampling strategy
  - Type 4: Our proposal
- Each run 5 times and compare mean $R^2$

### RESULTS

- Type 1:
- Type 2:
- Type 3:
- Type 4:

Low Performance
Normal Performance
High Performance

CHAPTER 5

# Conclusion

- 5-1: Final Conclusion

# Chapter 5: Conclusion

## 5-1 FINAL CONCLUSION

**01** We proposed a denoising pipeline to obtain **cleaner and more reliable health indicators**.

**02** We developed a twin-model framework **for robust iterative prediction**.
It also worked well with **limited target battery data**, reducing data requirements.

**03** Our approach achieved **higher accuracy and stability** compared to common baselines.

Operating conditions of NASA dataset differ slightly
How about other battery dataset?

# Thank you for your attention!

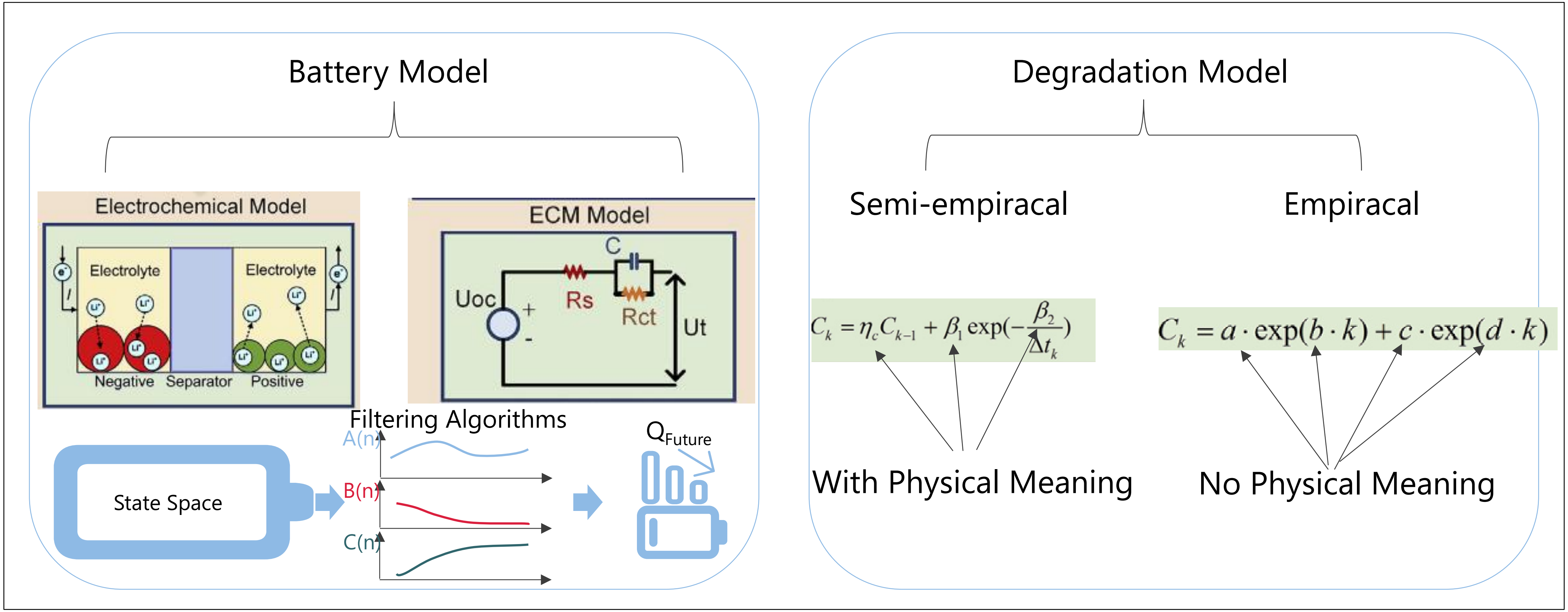
**Look forward to questions and suggestions!**

Institute für Combustion Engines – Forckenbeckstraße 4 – 52074 Aachen – Germany – www.TME.rwth-aachen.de

# BACKUP

1. Model Based RUL Prediction
2. Discret Wavelet Transformation
3. Three-Gate Structure of LSTM
4. Pesudo-Curve Generation

# 1. MODEL BASED RUL PREDICTION [1)]

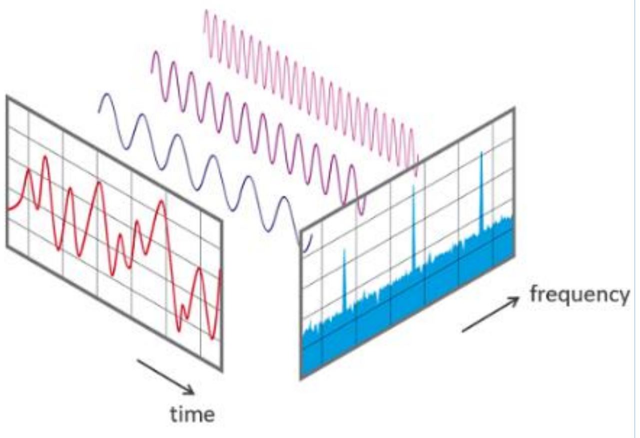



Source: 1) Battery Lifetime Prognostics

# 2. DISCRET WAVELET TRANSFORMATION

**Baseline**

**Fourier Transformation**

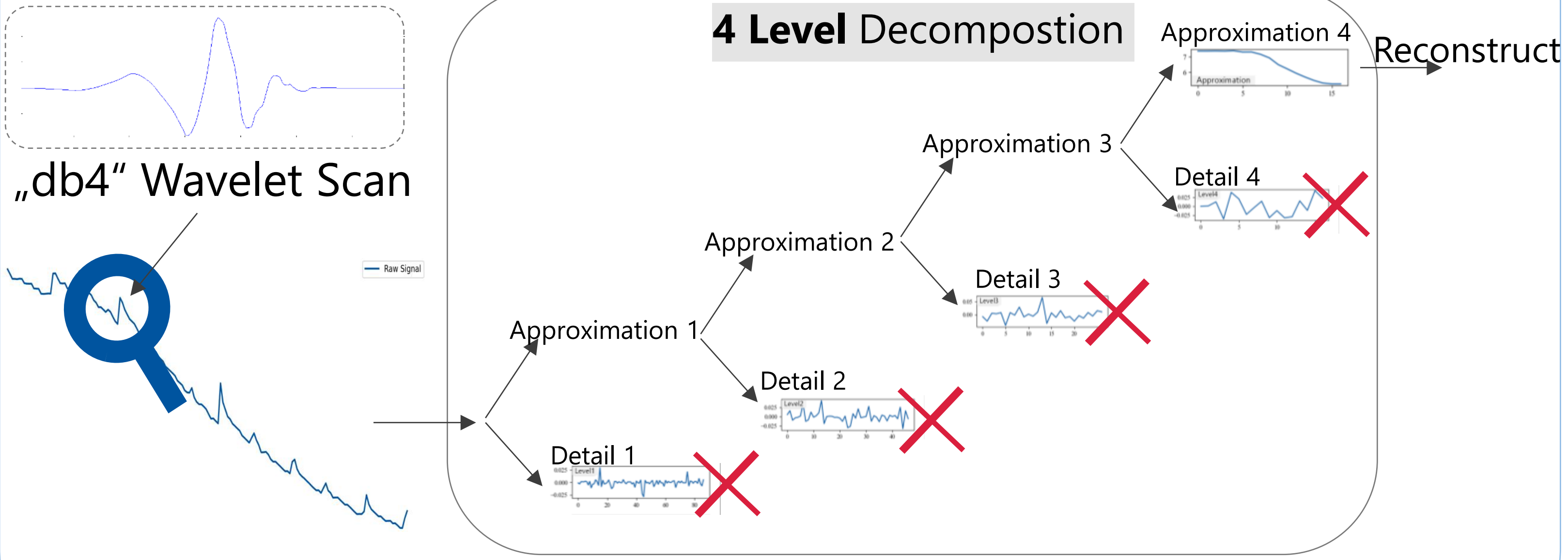



*View of a signal in the time and frequency domain*

Problem: No time domain.

**Our Proposal**

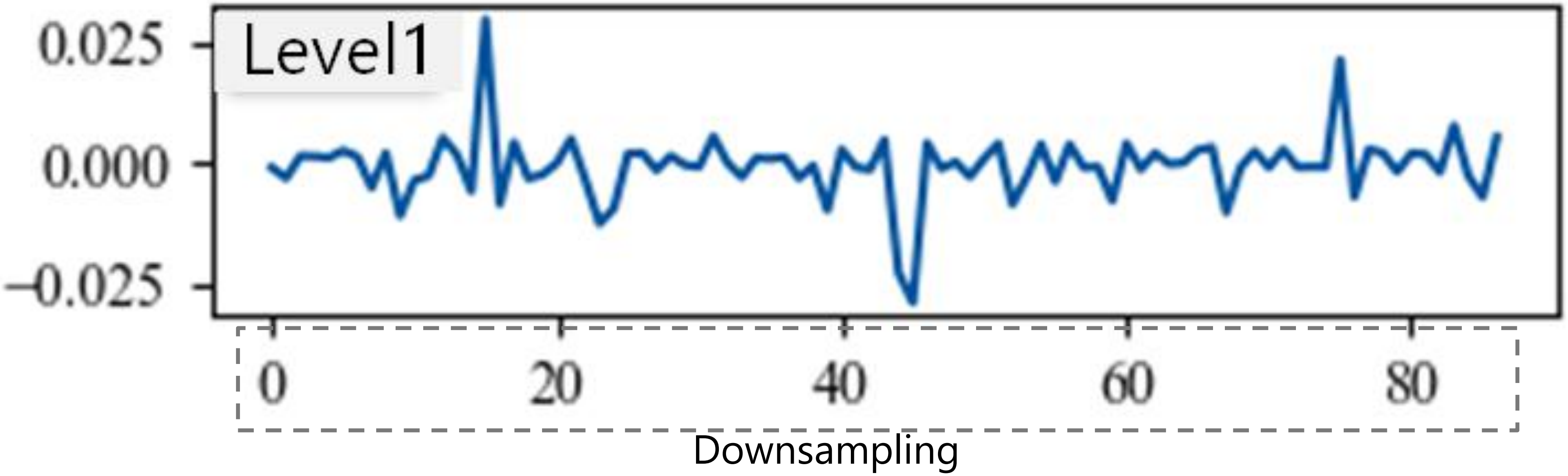
**DWT**




Scan → Decompostion → Reconstruction

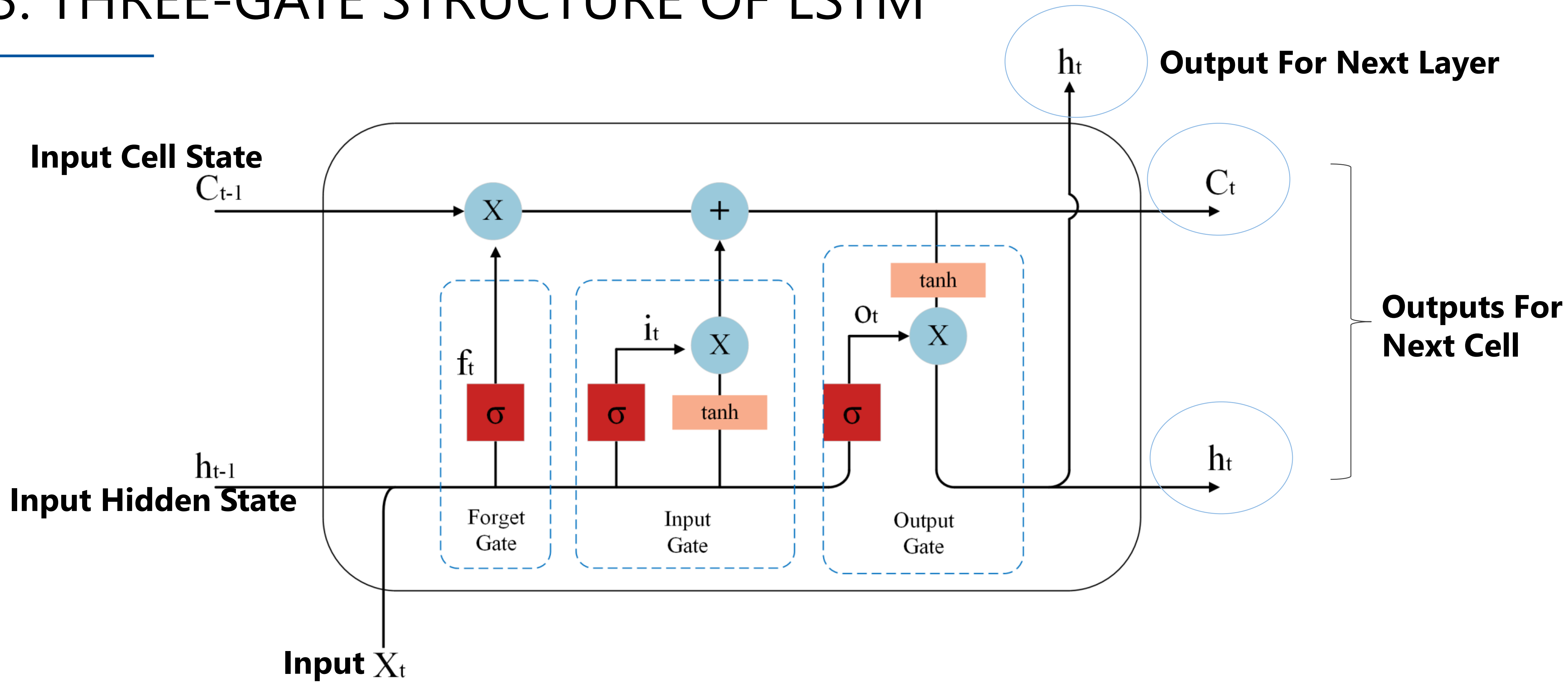
Figure Source: 1) Link of Left Figure 2) Link of Right Figure “db4”

# WHY CYCLES BECOME HALF?




Original
168 Cycles
Lv1 Decompositon →
Level1 Detail
168 Cycles
Downsampling →
Plotted
84 Cycles

# 3. THREE-GATE STRUCTURE OF LSTM

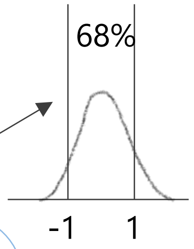


# 4. PESUDO-CURVE GENERATION

## Pesudo- Code

| Step | Description |
|---|---|
| 1 | Input the target capacity trajectory $y = [y_1, y_2, \ldots, y_T]$ of cycles-length $T$ |
| 2 | Construct cycle index $x = [1, 2, \ldots, T]$ |
| 3 | Fit a degree-3 polynomial ridge regression model: |
| | - Feature construction: `PolynomialFeatures(degree=3)` |
| | - Model: `Ridge(alpha=0.1)` |
| | - Fitting: `model.fit(x, y)` |
| 4 | Extract regression coefficients $\beta = [\beta_0, \beta_1, \beta_2, \beta_3]$ |
| 5 | Apply parameter perturbation with $\epsilon = 0.10$: |
| | $\tilde{\beta}_i = \beta_i \cdot (1 + \epsilon \cdot \mathcal{N}(0,1))$ for each $i \in \{0, 1, 2, 3\}$ |
| 6 | Compute pseudo curve: $\tilde{y} = x \cdot \tilde{\beta} + \tilde{\beta}_0$ |
| 7 | Normalize $\tilde{y}$ to $[0, 1]$ using MinMax scaling |
| 8 | Output the pseudo curve $\tilde{y}$ for model fine-tuning |

- Target capacity trajectory as input.
  - Cycle as x
  - Capacity as y
- Fitting function: $Q = \beta_0 + \beta_1 \cdot x^1 + \beta_2 \cdot x^2 + \beta_3 \cdot x^3$
- Extract parameters: $\beta_0, \beta_1, \beta_2, \beta_3$
- ***Parameter perturbation**




4 Random Nrs In Normal Distribution $N(0,1)$ :
[**0.497**,**−0.138**,**0.648**,**1.523**]

*0.1

* $\beta_0$ * $\beta_1$ * $\beta_2$ * $\beta_3$

\+ $\beta_0$ \+ $\beta_1$ \+ $\beta_2$ \+ $\beta_3$

New parameters, put them back Q(x)