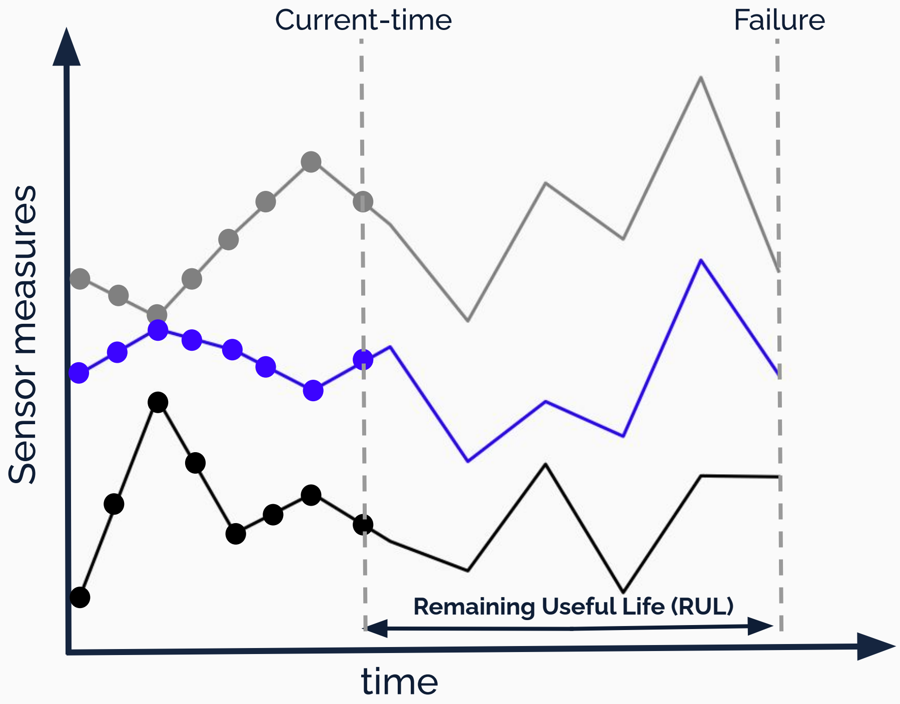
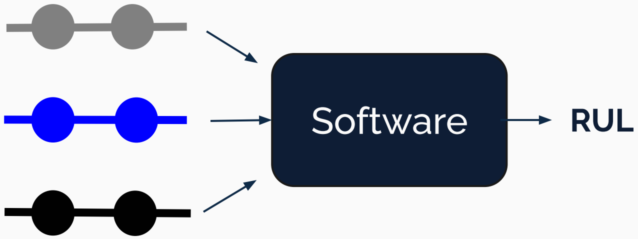
TIME-TO-FAILURE PREDICTION (TTF)

Forecasting plane engine failure with sensors data

Goal: Predicting airplane's engine failure before it happens





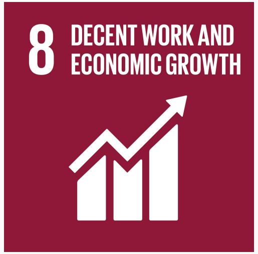
Objectives:

- Offer continuous maintenance assistance
- Decrease components production
- Decrease maintenance time
- Improve the engine efficiency

Value proposition:

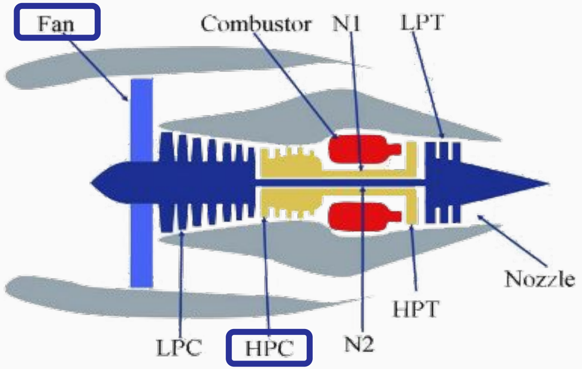
SUSTAINABLE DEVELOPMENT GOALS

A survival analysis framework for aircraft components supports the UN Sustainable Development Goals by **improving component lifespan**, **reducing waste**, and **fostering innovation**.



Challenges of the data

How to deal with different types of failure?



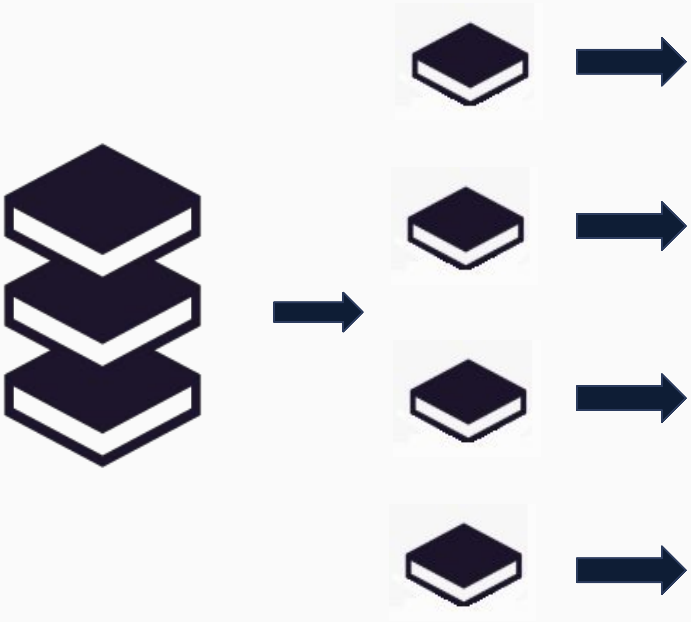
How to adapt forecasts to environment settings?

Air temperature

Altitude

Speed

Challenges of the data



1 failure type, 1 condition

1 failure type, 6 conditions

2 failure types, 1 condition

2 failure types, 6 conditions

Challenges of the data

1 failure type, 1 condition

1 failure type, 6 conditions

2 failure types, 1 condition

2 failure types, 6 conditions

Training, Validation and Testing

Trajectory	Cycle	Features	RUL
1	1	…	0
1	2	…	
1	3	…	
2	1	…	0
2	2	…	
3	1	…	0
3	2	…	
3	3	…	

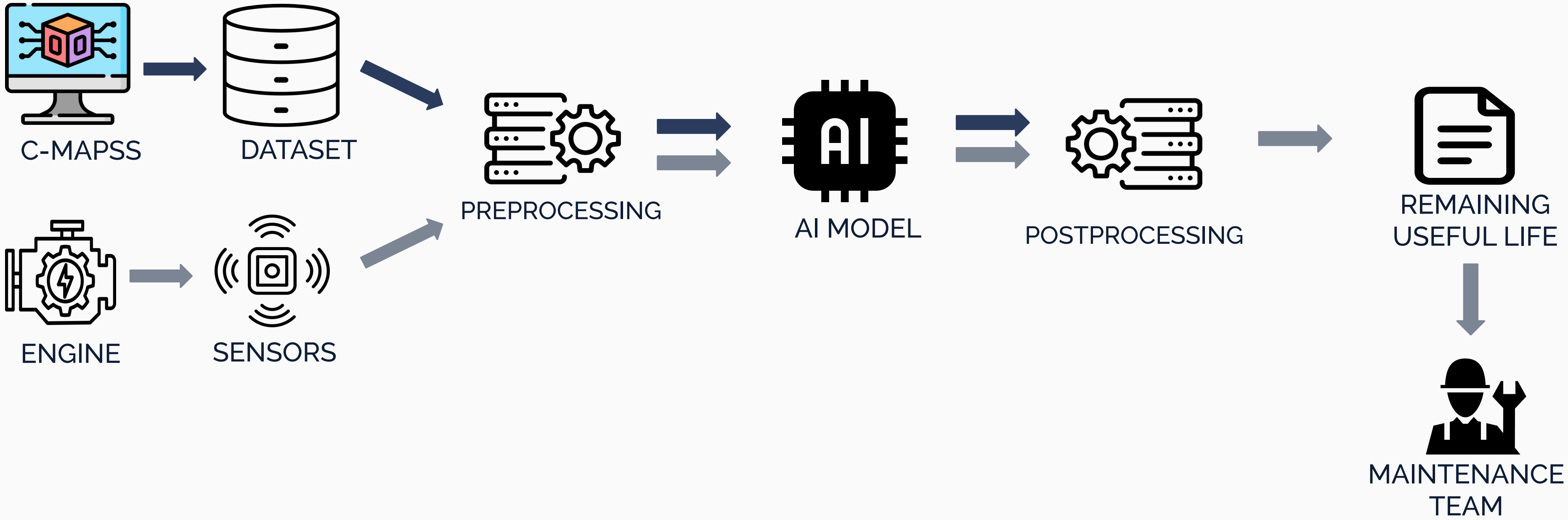
Trajectory	Cycle	Features	RUL
1	1	…	6
1	2	…	
2	1	…	5
2	2	…	
3	1	…	17
3	2	…	
3	3	…	

Training , Validation and Testing

<table>
<tr><th>Trajectory</th><th>Cycle</th><th>Features</th><th>RUL</th></tr>
<tr><td>1</td><td>1</td><td>…</td><td rowspan="3">0</td></tr>
<tr><td>1</td><td>2</td><td>…</td></tr>
<tr><td>1</td><td>3</td><td>…</td></tr>
<tr><td>2</td><td>1</td><td>…</td><td rowspan="2">0</td></tr>
<tr><td>2</td><td>2</td><td>…</td></tr>
<tr><td>3</td><td>1</td><td>…</td><td rowspan="3">0</td></tr>
<tr><td>3</td><td>2</td><td>…</td></tr>
<tr><td>3</td><td>3</td><td>…</td></tr>
</table>

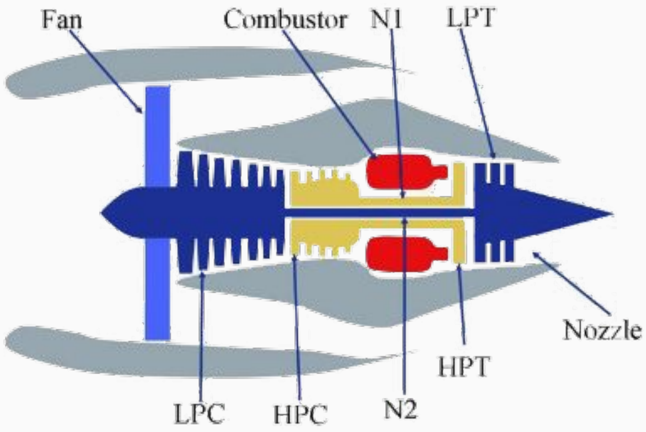
<table>
<tr><th>Trajectory</th><th>Cycle</th><th>Features</th><th>RUL</th></tr>
<tr><td>1</td><td>1</td><td>…</td><td rowspan="2">6</td></tr>
<tr><td>1</td><td>2</td><td>…</td></tr>
<tr><td>2</td><td>1</td><td>…</td><td rowspan="2">5</td></tr>
<tr><td>2</td><td>2</td><td>…</td></tr>
<tr><td>3</td><td>1</td><td>…</td><td rowspan="3">17</td></tr>
<tr><td>3</td><td>2</td><td>…</td></tr>
<tr><td>3</td><td>3</td><td>…</td></tr>
</table>

Functional diagram:



Research questions

- Which are the most relevant features to collect ?

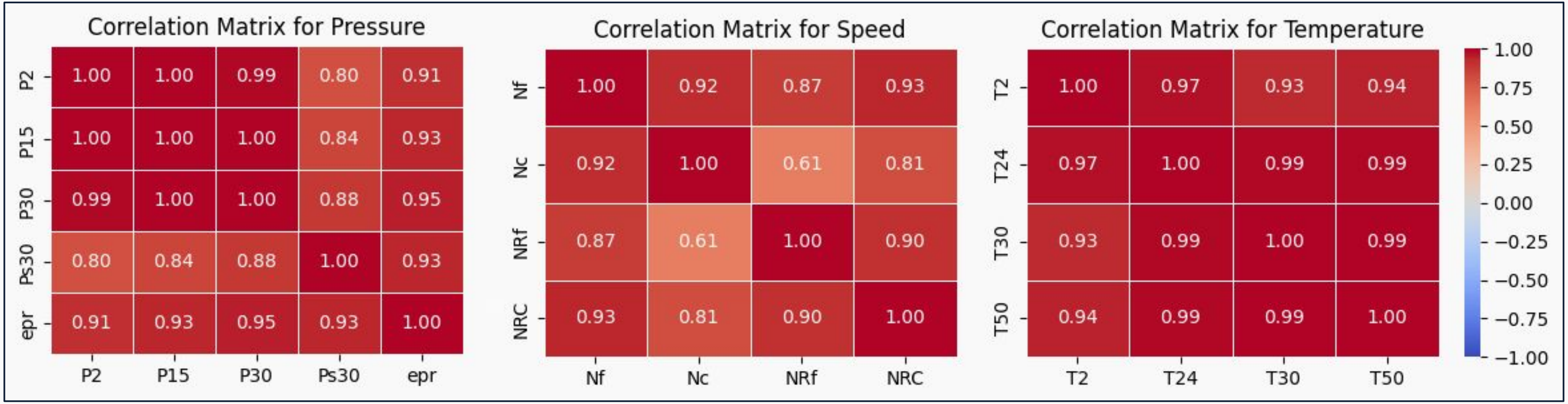


Symbol	Description	Units
T2	Total temperature at fan inlet	°R
T24	Total temperature at LPC outlet	°R
T30	Total temperature at HPC outlet	°R
T50	Total temperature at LPT outlet	°R
P2	Pressure at fan inlet	psia
P15	Total pressure in bypass-duct	psia
P30	Total pressure at HPC outlet	psia
Nf	Physical fan speed	rpm
Nc	Physical core speed	rpm
epr	Engine pressure ratio (P50/P2)	--
Ps30	Static pressure at HPC outlet	psia
phi	Ratio of fuel flow to Ps30	pps/psi
NRf	Corrected fan speed	rpm
NRc	Corrected core speed	rpm
BPR	Bypass Ratio	--
farB	Burner fuel-air ratio	--
htBleed	Bleed Enthalpy	--
Nf_dmd	Demanded fan speed	rpm
PCNfR_dmd	Demanded corrected fan speed	rpm
W31	HPT coolant bleed	lbm/s
W32	LPT coolant bleed	lbm/s

Research questions

- Which are the most relevant features to collect ?

➔ Correlation Matrices

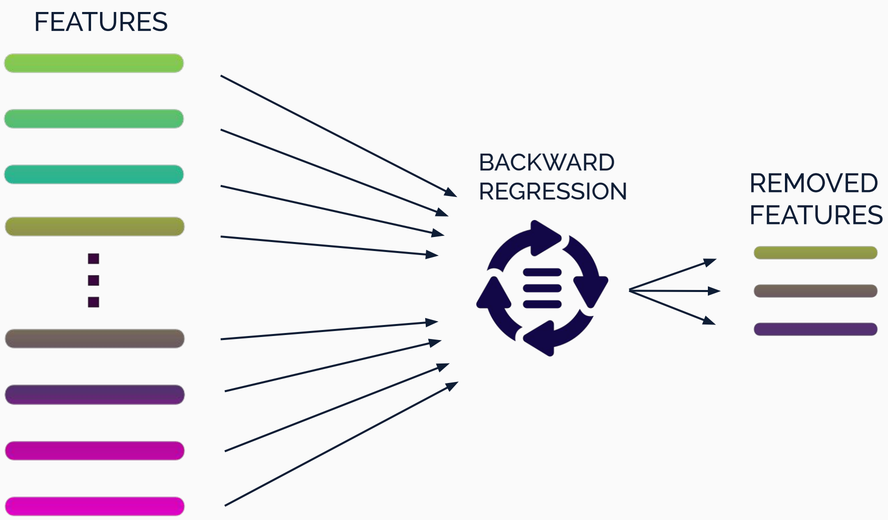


Research questions

- Which are the most relevant features to collect ?

➔ Backward Regression Analysis

Start with all features and iteratively remove the least significant ones (based on p-value) until only the most important features remain.

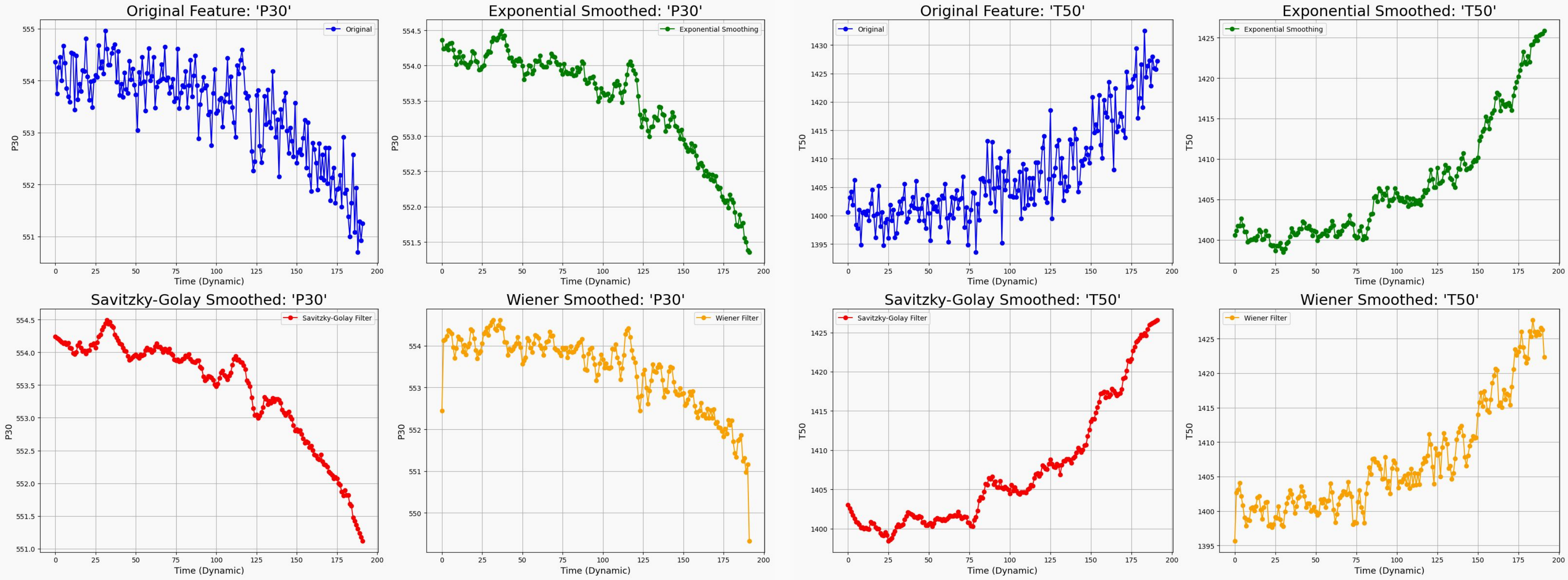


In our dataset:

- P15
- NRC
- Mach Number

Research questions

- How can sensor noise in time-series data be effectively mitigated to improve the accuracy in our RUL predictions?



Research questions

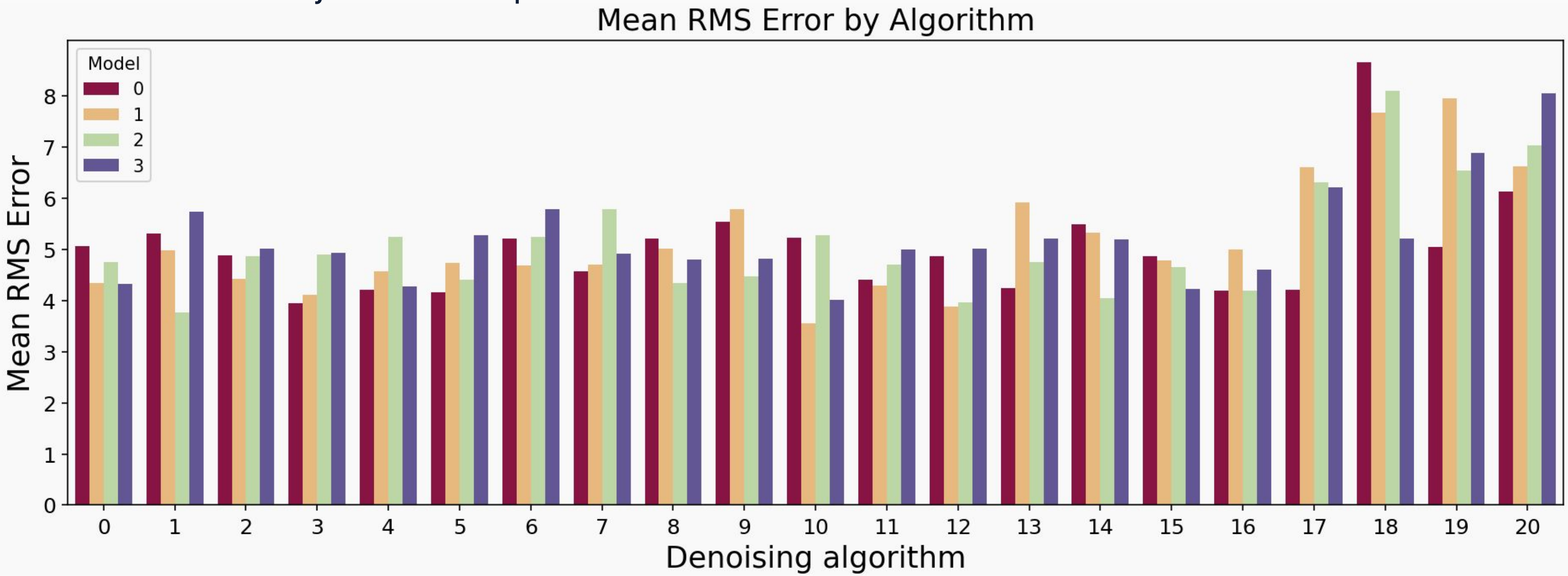
- How can sensor noise in time-series data be effectively mitigated to improve the accuracy in our RUL predictions?

    - Visually, the savitzky-Golay offers slightly overall better performance.
      In a nutshell it uses a polynomial smoothing technique that applies a moving window of points and fits a polynomial (usually quadratic or cubic) to each window of data, then replaces the central value in the window with the value predicted by the polynomial.

    - Indeed is a generalizable tool for smoothing and denoising data, though its effectiveness depends on the type of data and the noise present. It is versatile and widely applicable across various domains, especially where signal processing and data smoothing are required.

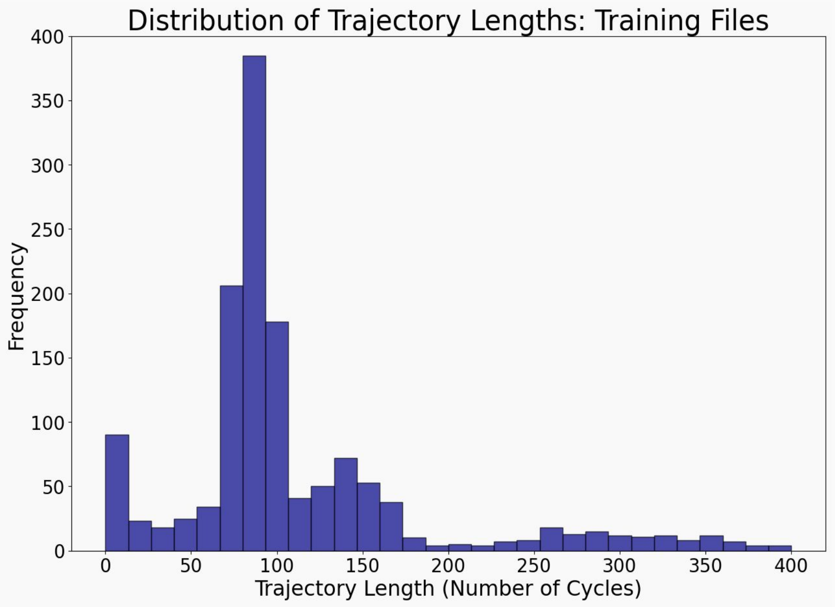
Research questions

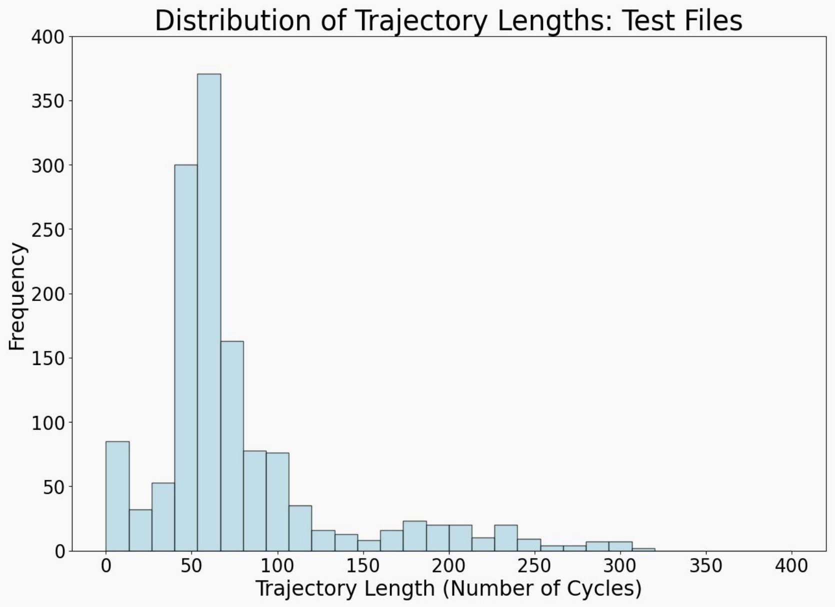
- How can sensor noise in time-series data be effectively mitigated to improve the accuracy in our RUL predictions?



Research questions

- How can we deal with trajectories having different length ?

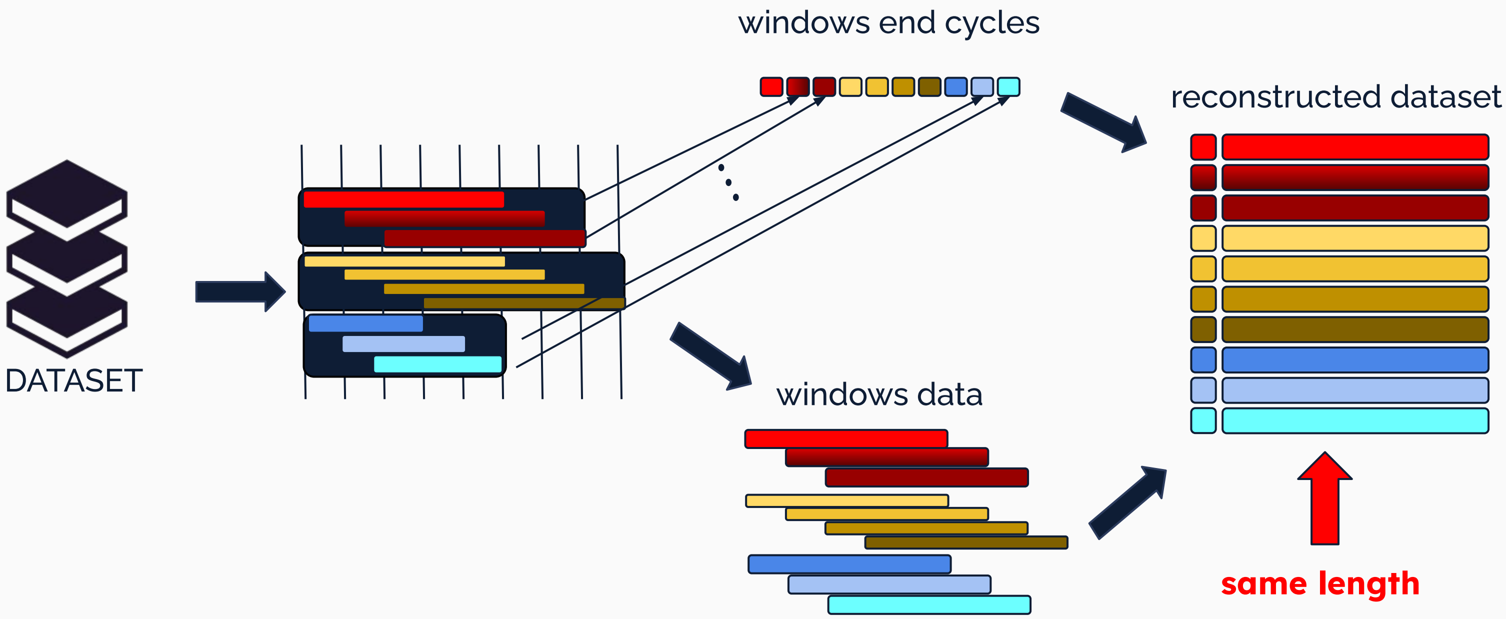




Research questions

- How can we deal with trajectories having different length ?

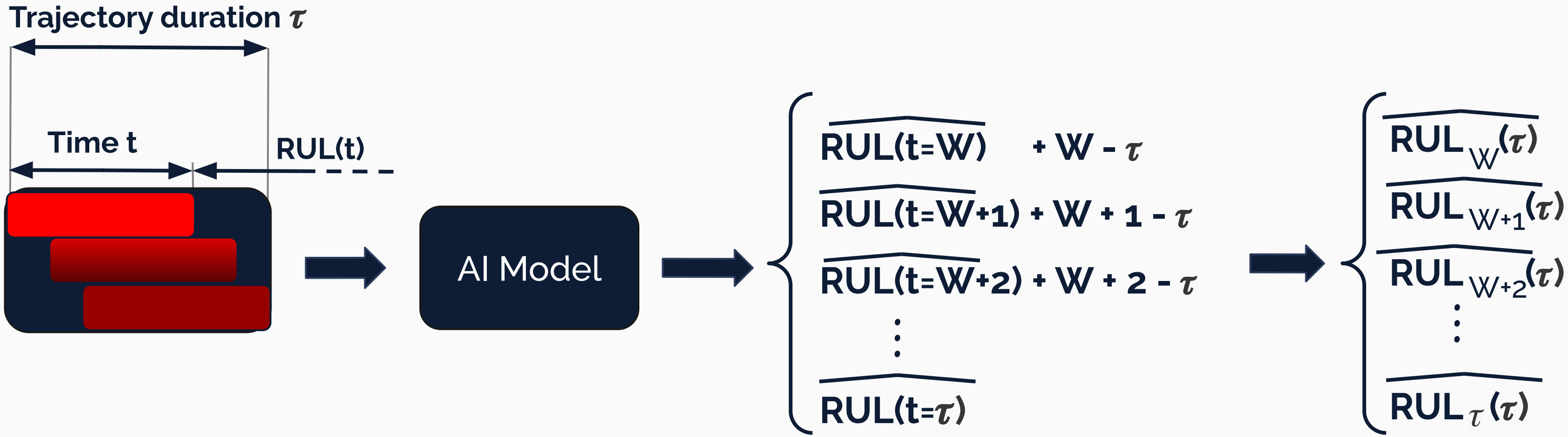
→ Windows



Research questions

- How can we deal with trajectories having different length ?

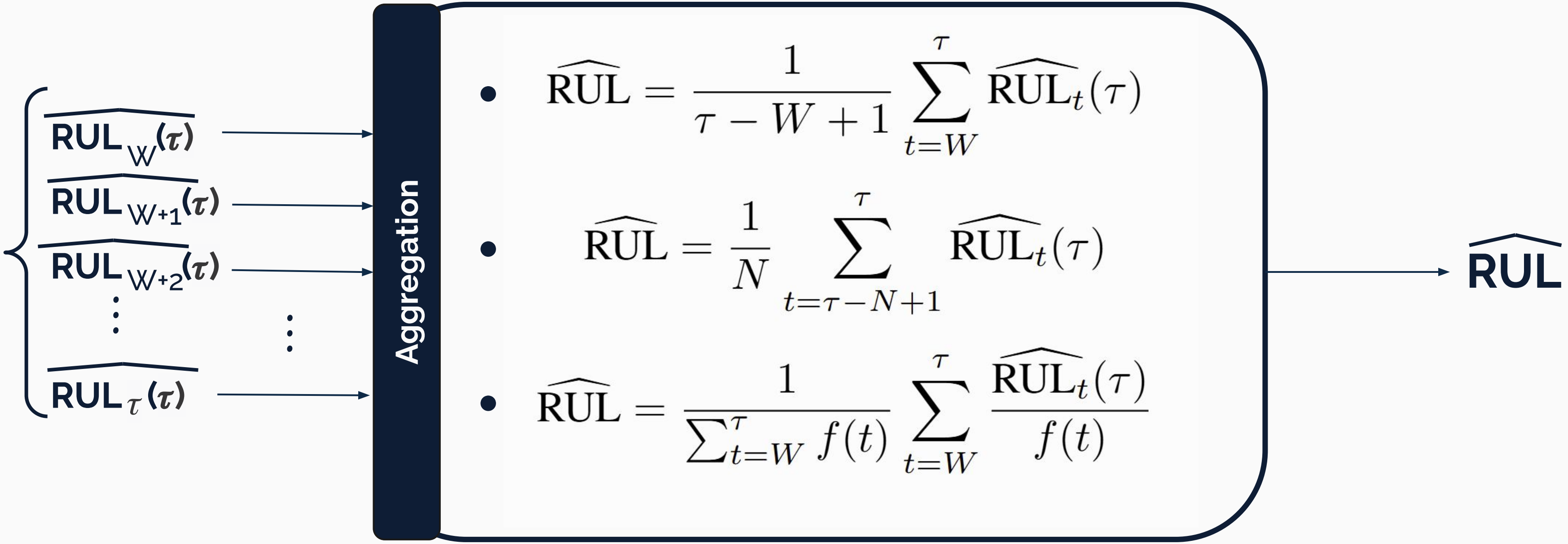
→ Aggregation



Research questions

- How can we deal with trajectories having different length ?

➔ Aggregation

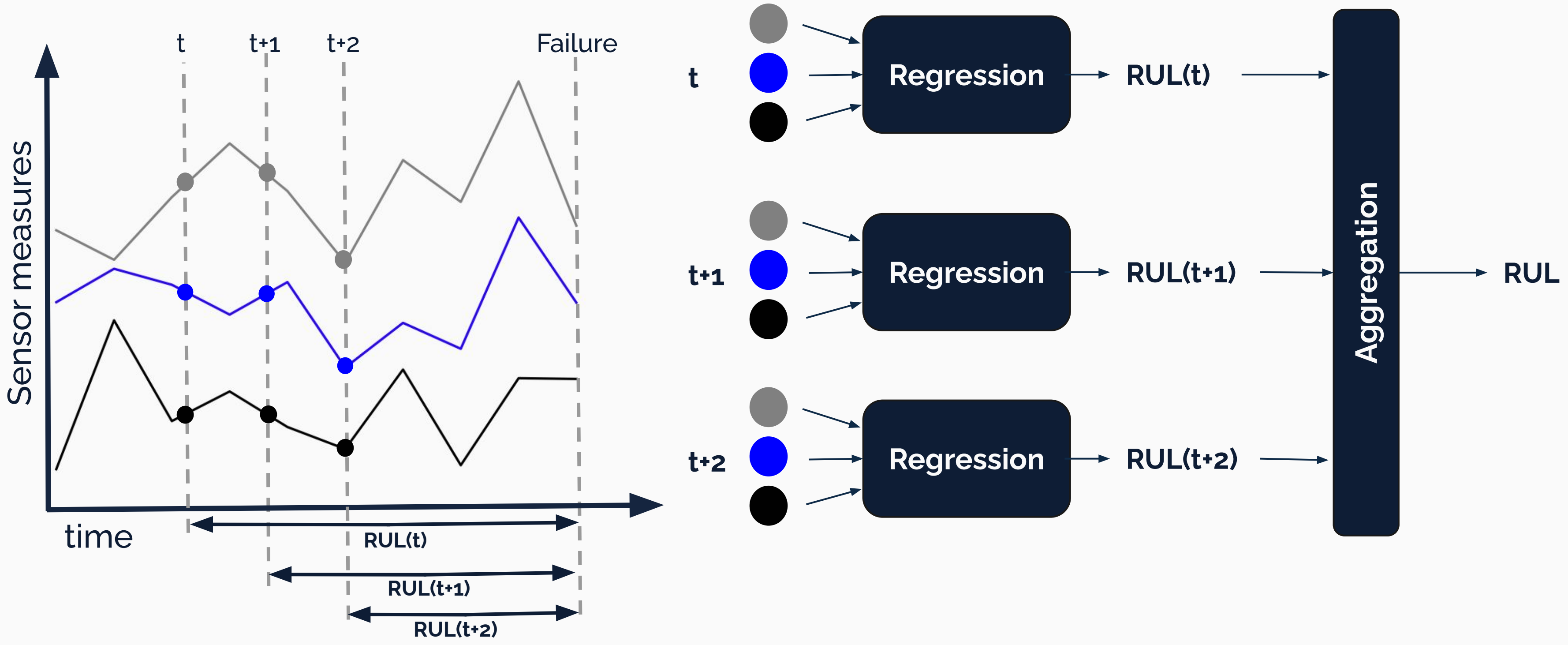


- $$\widehat{\mathrm{RUL}} = \frac{1}{\tau - W + 1} \sum_{t=W}^{\tau} \widehat{\mathrm{RUL}}_t(\tau)$$
- $$\widehat{\mathrm{RUL}} = \frac{1}{N} \sum_{t=\tau-N+1}^{\tau} \widehat{\mathrm{RUL}}_t(\tau)$$
- $$\widehat{\mathrm{RUL}} = \frac{1}{\sum_{t=W}^{\tau} f(t)} \sum_{t=W}^{\tau} \frac{\widehat{\mathrm{RUL}}_t(\tau)}{f(t)}$$

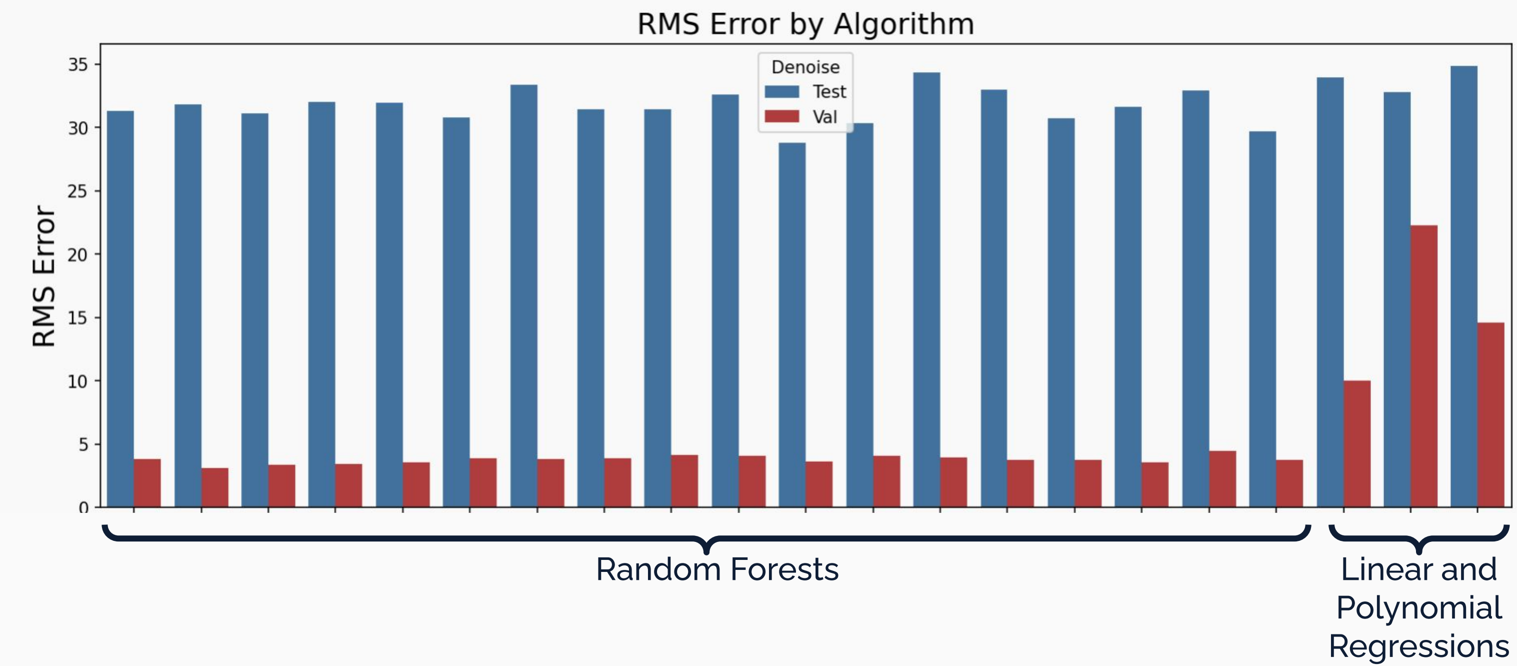
Strategies:

- Machine Learning Models
- Long Short-Term Memory
- Transformers

Machine Learning Models



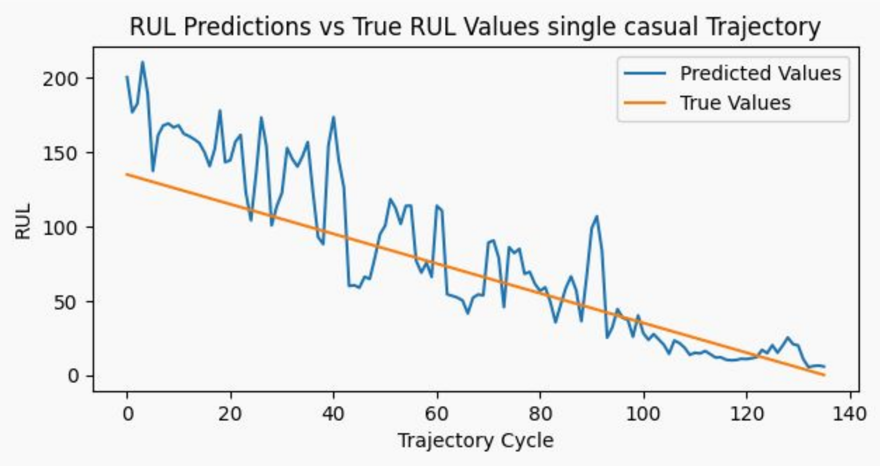
Machine Learning Models: features



Best: RandomForest(200 trees, 50 max depth, 5 features)

Long Short-Term Memory

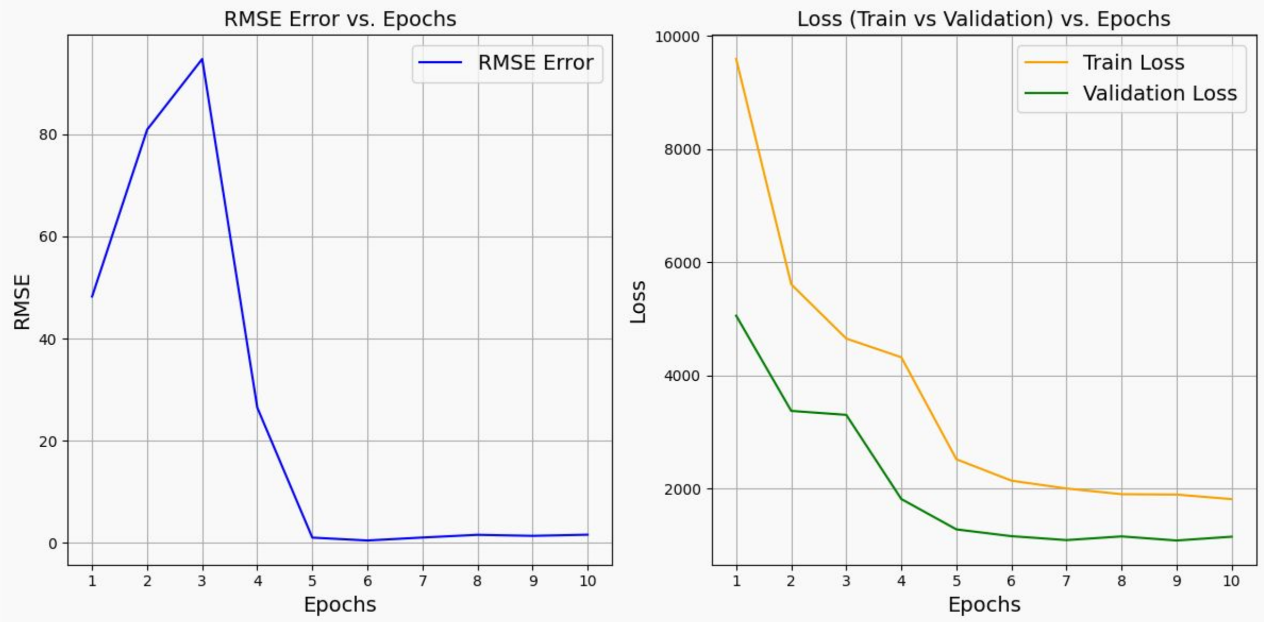
➔ Training and Evaluation Phase



Aggregation method	RMS Error
Average	16.79
Average weighted by cycle	15.82
Average weighted by cycle (sqrt)	15.93
Average weighted by cycle (exp)	1.21
Average weighted by cycle (log)	16.18
Average on the last 5 cycles	3.61
Average on the last 10 cycles	11.89
Average on the last 15 cycles	13.88

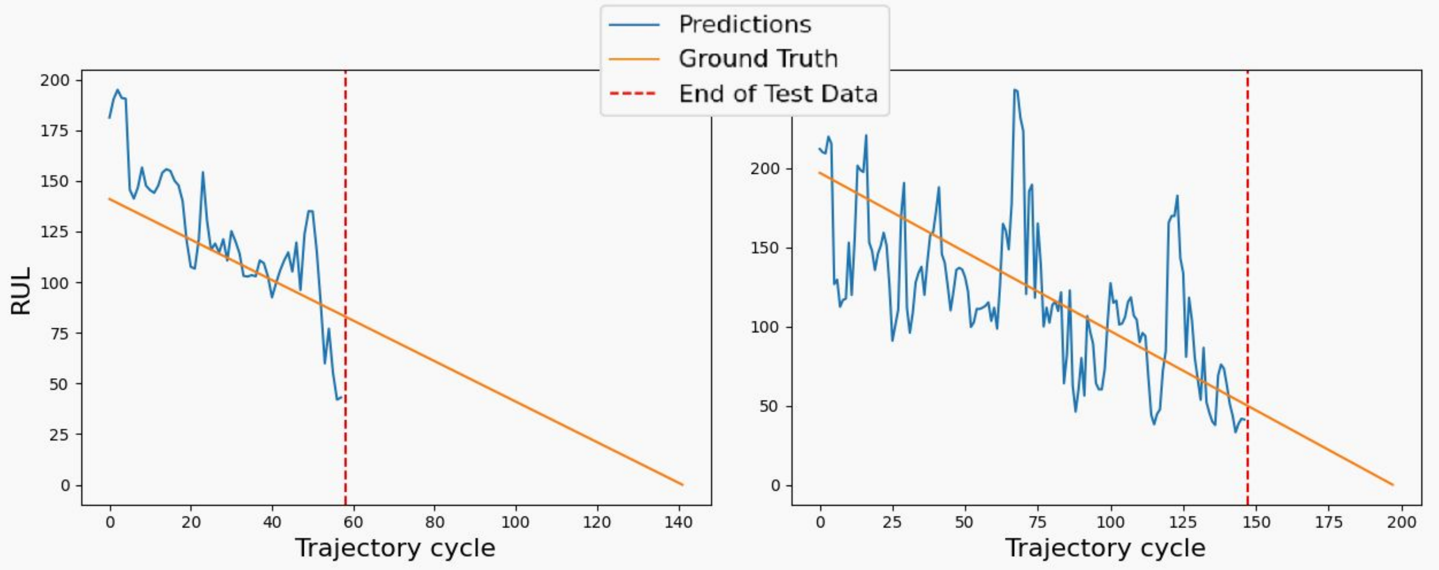
Long Short-Term Memory

➔ How many training epochs the model needs to converge?



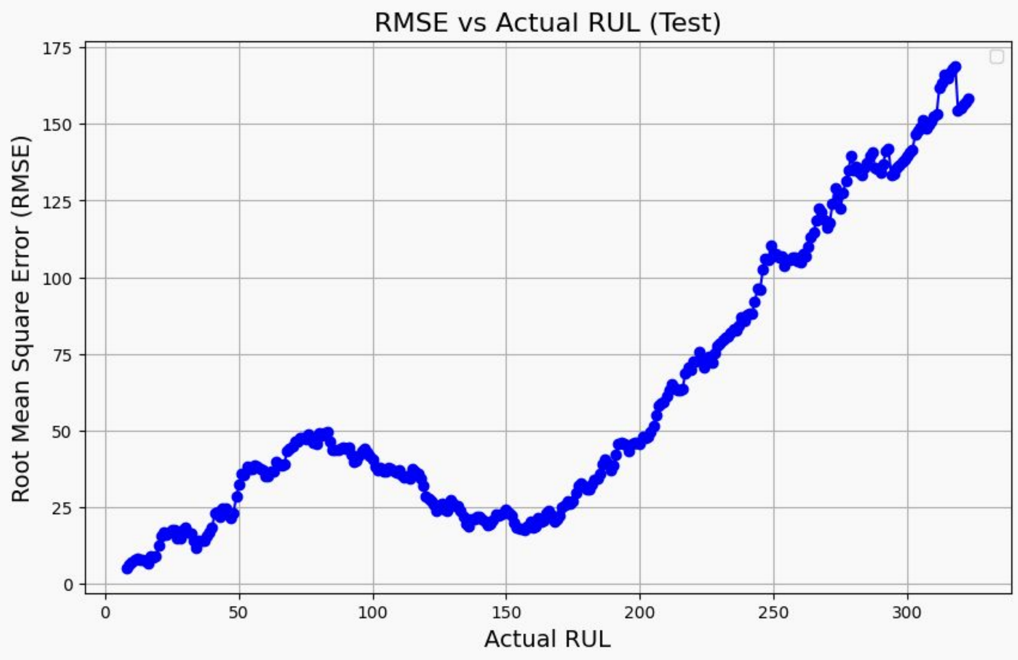
Long Short-Term Memory

➔ Test Phase, the trajectory are not complete



Long Short-Term Memory

- ➔ The closer to failure, the more evident the signs of failure become.

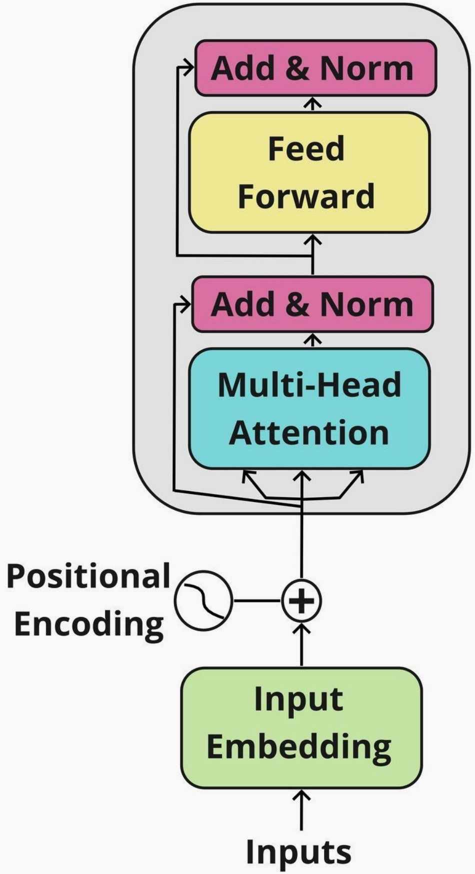


Transformers

The method adapts the Transformer architecture by detaching its encoder and modifying it.

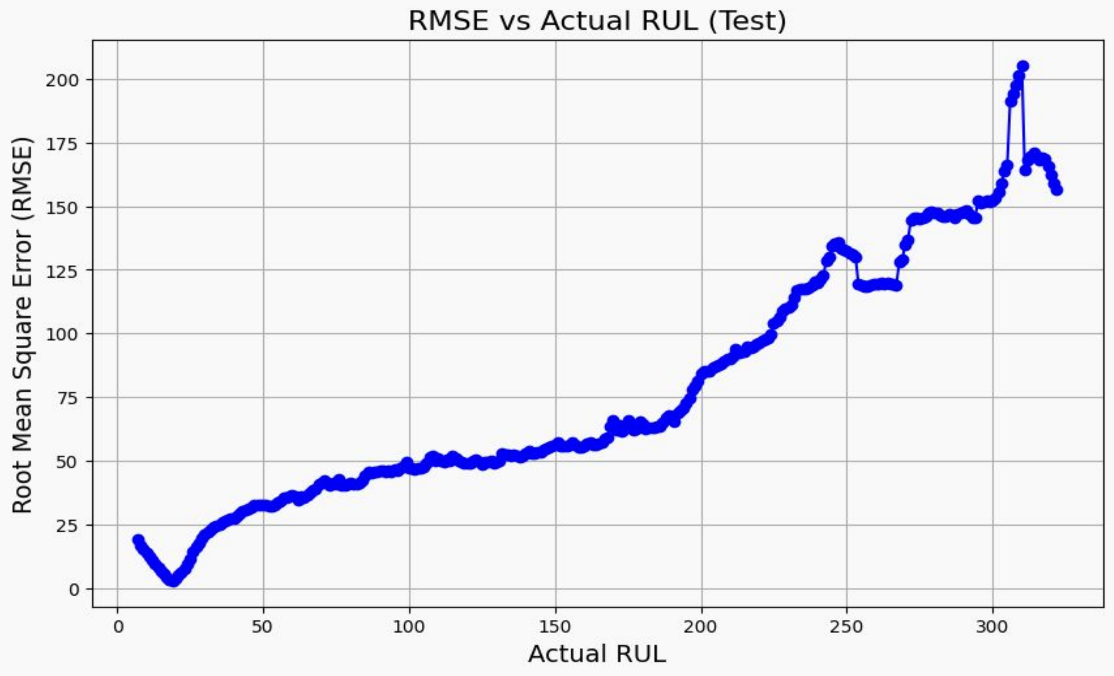
-Batch normalization replaces layer normalization after to handle outliers better, unlike NLP.

-The last layer is a fully connected layer to predict a single value.



Transformers

Encouraging results regarding the model's ability to predict quite accurate RULs near failures



Indeed:
RMSE for RUL > 20 : 61.1
RMSE for RUL <= 20: 8.6

Any question ?

- Tanguy Dugas du Villard
- Vito Perrucci
- Lorenzo Suppa