POLITECNICO DI MILANO

Ensembles of Models for Prognostics and Health Management

Piero Baraldi[1], Francesco Di Maio[1], Enrico Zio[2,1]

[1]Dipartimento di Energia, Politecnico di Milano, Italy
[2]Chair on Systems Science and the Energetic challenge, European Foundation for New Energy-EDF, Ecole Centrale Paris and Supelec, France

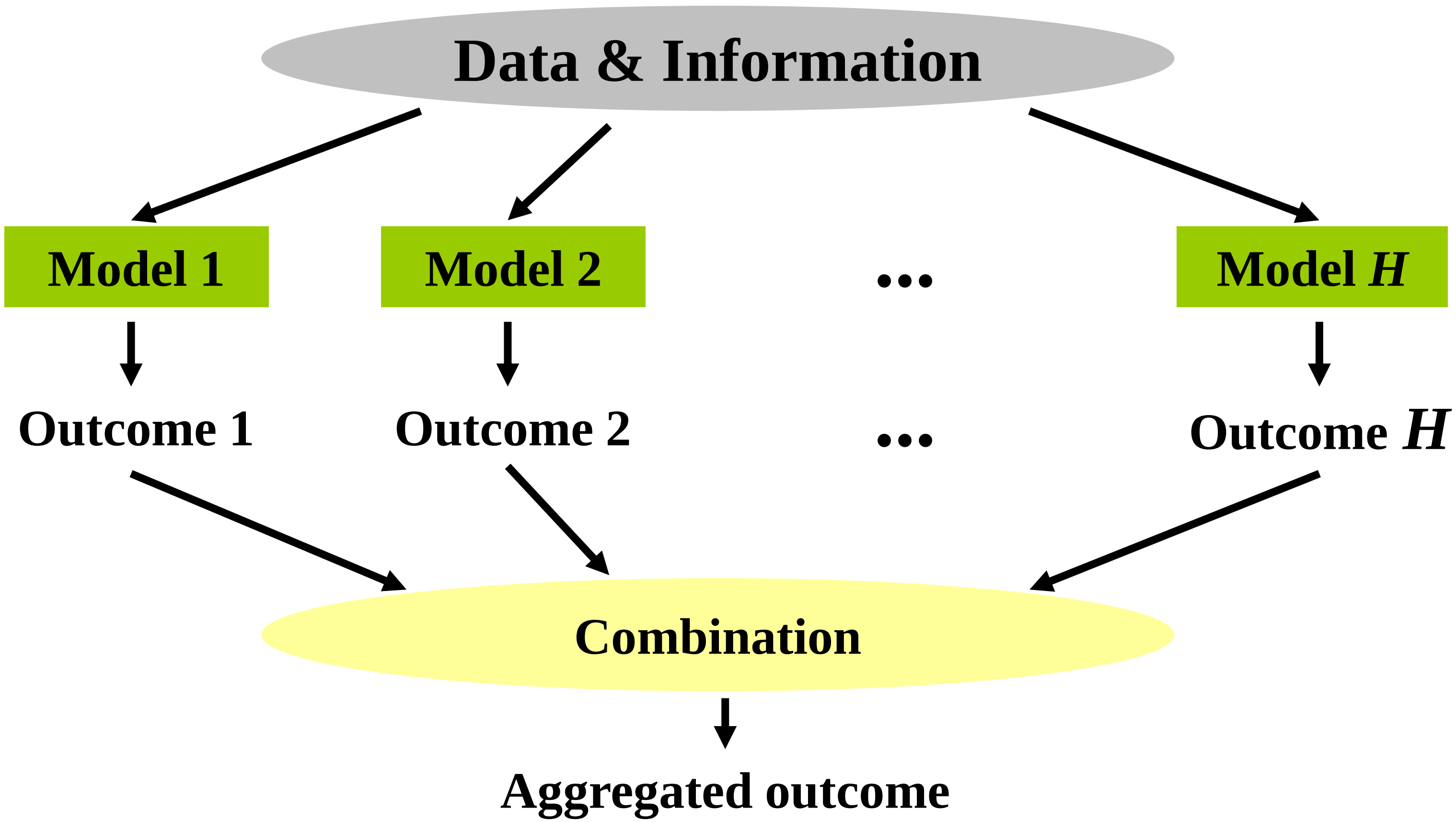
PHME 2016, July 6, Bilbao, Spain

Tutorial Contents

- Ensembles of Models for Prognostics and Health Management (PHM):
  - Accuracy
  - Confidence Estimation
  - Incremental Learning

ENSEMBLES OF MODELS

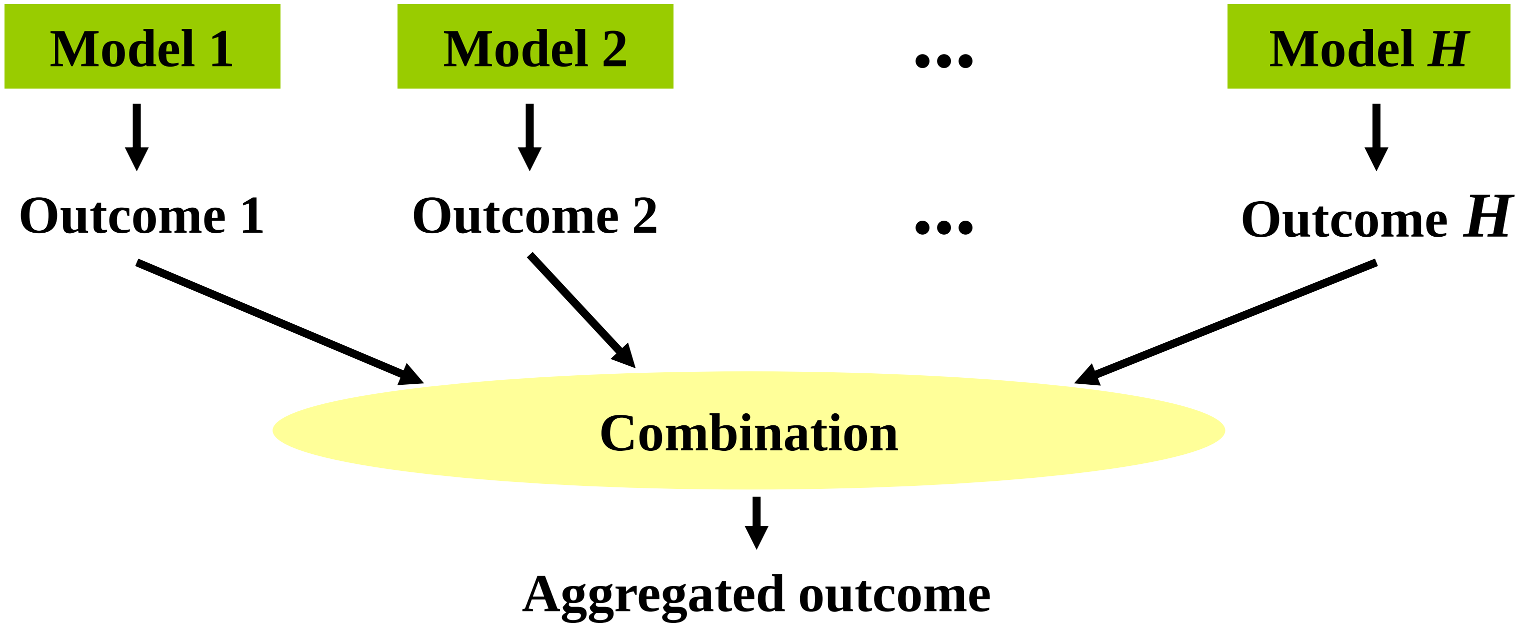
Ensembles of Models



Developing an Ensemble of Models

1. Generate the individual models of the ensemble

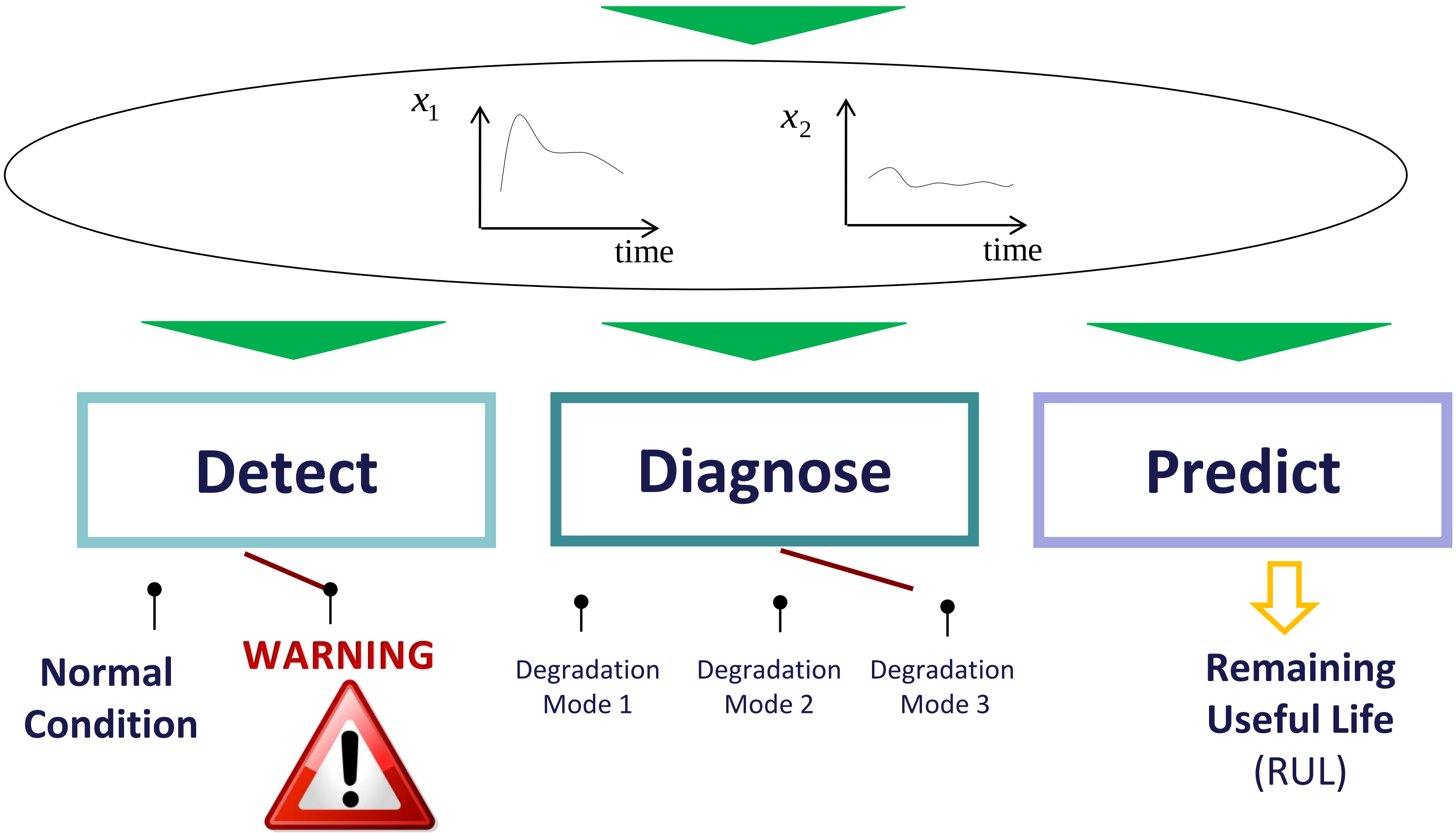
2. Combine the outputs of the individual models



ENSEMBLES OF MODELS FOR PROGNOSTICS AND HEALTH MANAGEMENT

Prognostics and Health Management





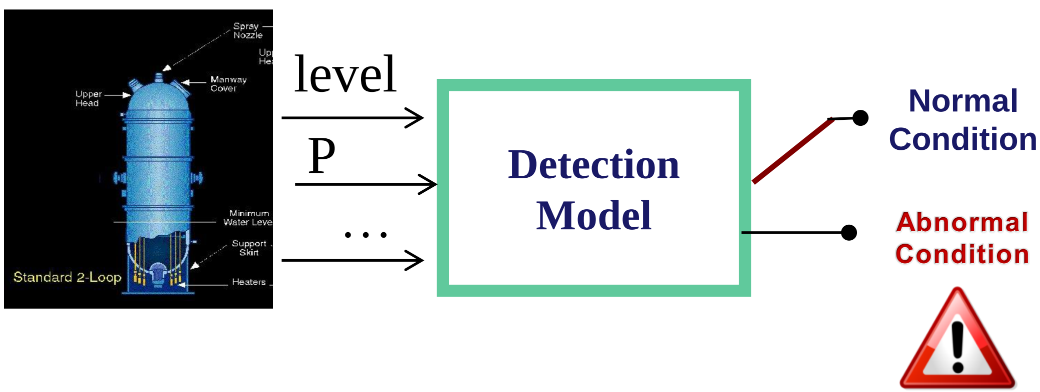
Tutorial Contents

- Ensembles of Models for Prognostics and Health Management (PHM):
  - Accuracy
  - Confidence Estimation
  - Incremental Learning

- **Accuracy:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
    - **A. Generate the individual models of the ensemble**
    - **B. Combine the outputs of the individual models**
  - **Application: prognostics of turbofan engines**

- Fault Detection:
  - Low rate of False Alarm
  - Low rate of Missing Alarm

Example:



False Alarm Rates	Missing Alarm Rates
0,24%	0,38%

Accuracy: What is it?

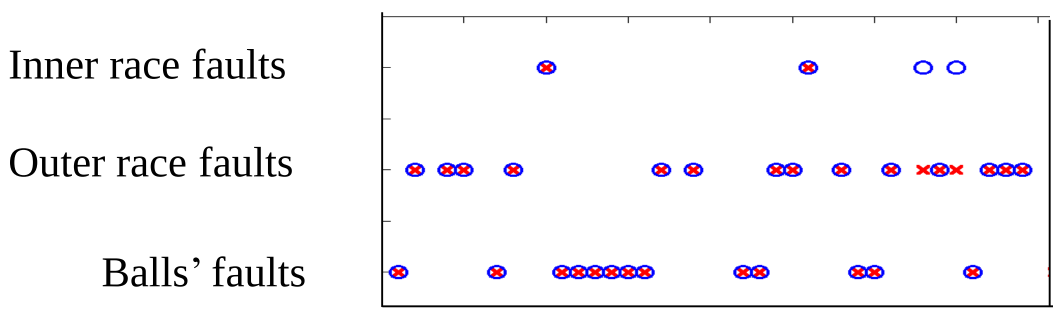
- Fault Diagnostics:
  - Low Misclassification rate

Example:

Vibrational signals

Diagnostic Model

- **C_1 = Inner race**
- **C_2 = Balls**
- **C_3 = Outer race**



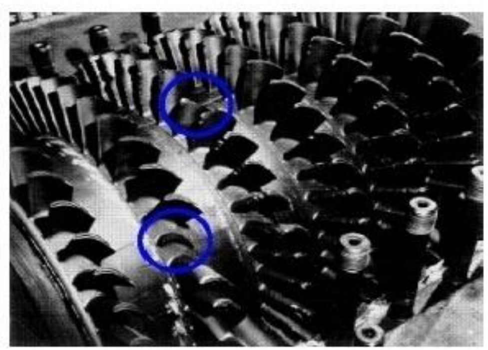
o = true
× = diagnostic model

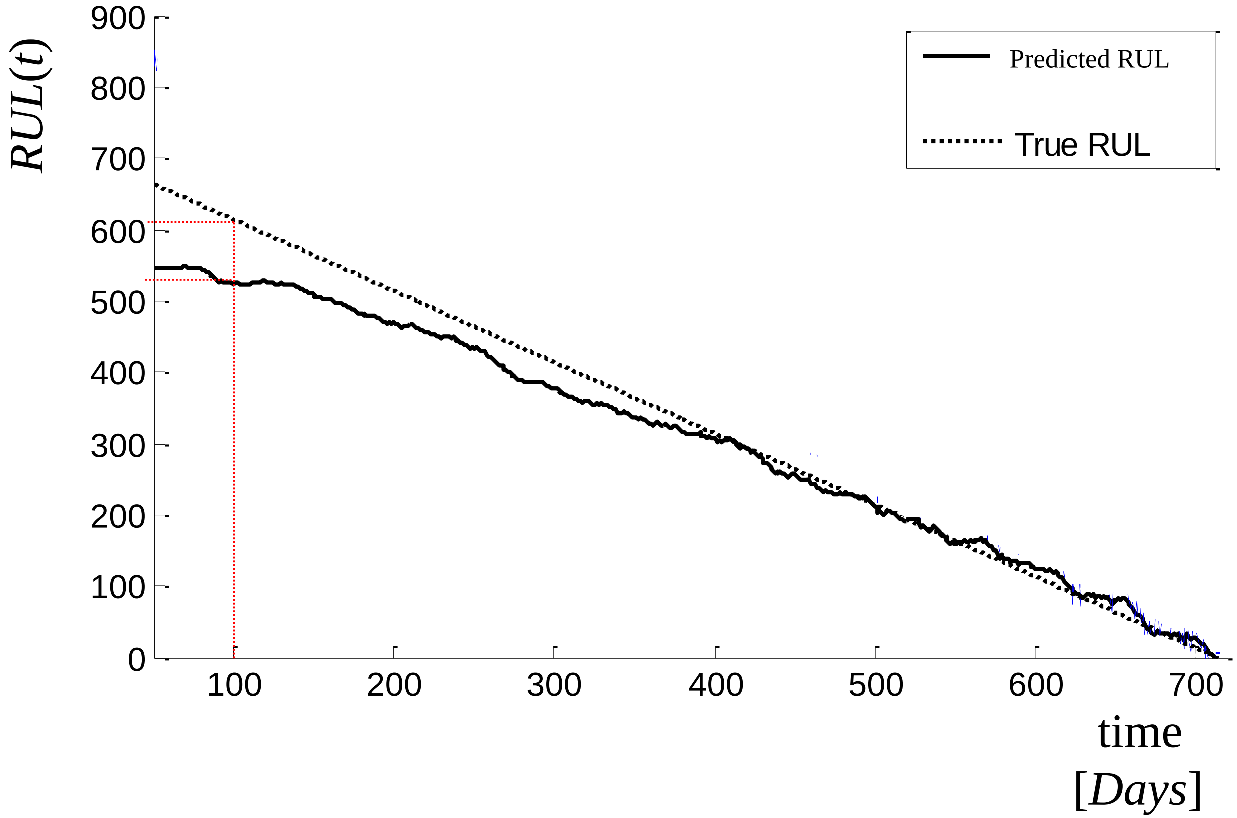
Misclassification rate = 0,78%

Accuracy: What is it?

➢ Prognostics

Example:





- **Accuracy:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
    - **A. Generate the individual models of the ensemble**
    - **B. Combine the outputs of the individual models**
  - **Application: prognostics of turbofan engines**

Why Ensembles?

"I'll ask the audience"

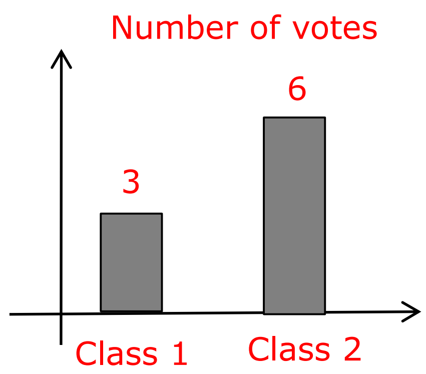
multiple performers

individual answer

combination of the answers

Why Ensembles? [Diagnostics]

- Condorcet Jury Theorem (1786)
  - *2 class problem*
  - *H* classifiers (*H* is an odd number)
  - p_1 = probability that an individual classifier chooses the correct class (equal for all the classifiers and all the patterns)
  - the classifier outcomes are independent
  - ensemble outcome: simple majority voting of the *H* classifiers



Ensemble outcome = class 2

Why Ensembles? [Diagnostics]

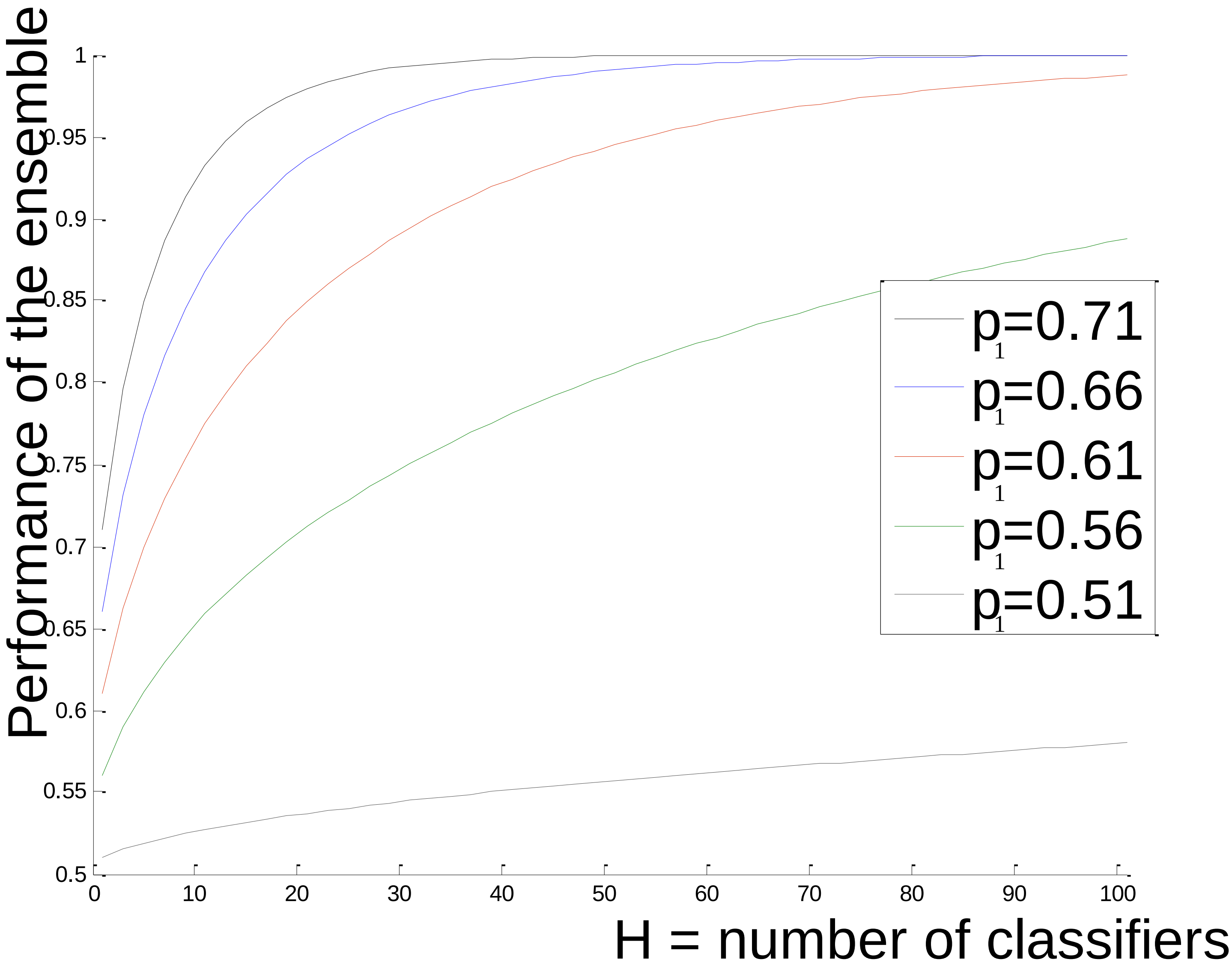
- Condorcet Jury Theorem (1786)
  - *2 class problem*
  - *H* classifiers (*H* is an odd number)
  - p_1 = probability that an individual classifier chooses the correct class (equal for all the classifiers and all the patterns)
  - the classifier outcomes are independent
  - ensemble outcome: simple majority voting of the *H* classifiers

- p_{ens} = probability that the ensemble chooses the correct class

$$p_{ens} = P\{\text{majority of the classifiers choose the correct class}\} =$$
$$P\left\{at\ least\ \frac{H+1}{2}\ \text{classifiers choose the correct class}\right\} =$$
$$\sum_{k=\frac{H+1}{2}}^{H} P\{k\ \text{classifiers choose the correct class}\} = \sum_{k=\frac{H+1}{2}}^{H} \binom{H}{k} p_1^k (1-p_1)^{H-k}$$

- If $\begin{cases} p_1 > 0.5 \\ H \to \infty \end{cases}$ $\boxed{p_{ens} \to 1}$

Why Ensembles? [Diagnostics]



Why Ensembles?

- Questionable outcomes provided by ensemble of models…

Who will you choose…
Jesus or Barabbas?

Why Ensembles?

- Questionable outcomes provided by ensemble of models…

Indipendence of the classifiers?

Matthew 27:20: But the **chief priests and the elders persuaded the crowd** to ask for Barabbas and to have Jesus executed.

- **Accuracy:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
    - **A. Generate the individual models of the ensemble**
    - **B. Combine the outputs of the individual models**
  - **Application: prognostics of turbofan engines**

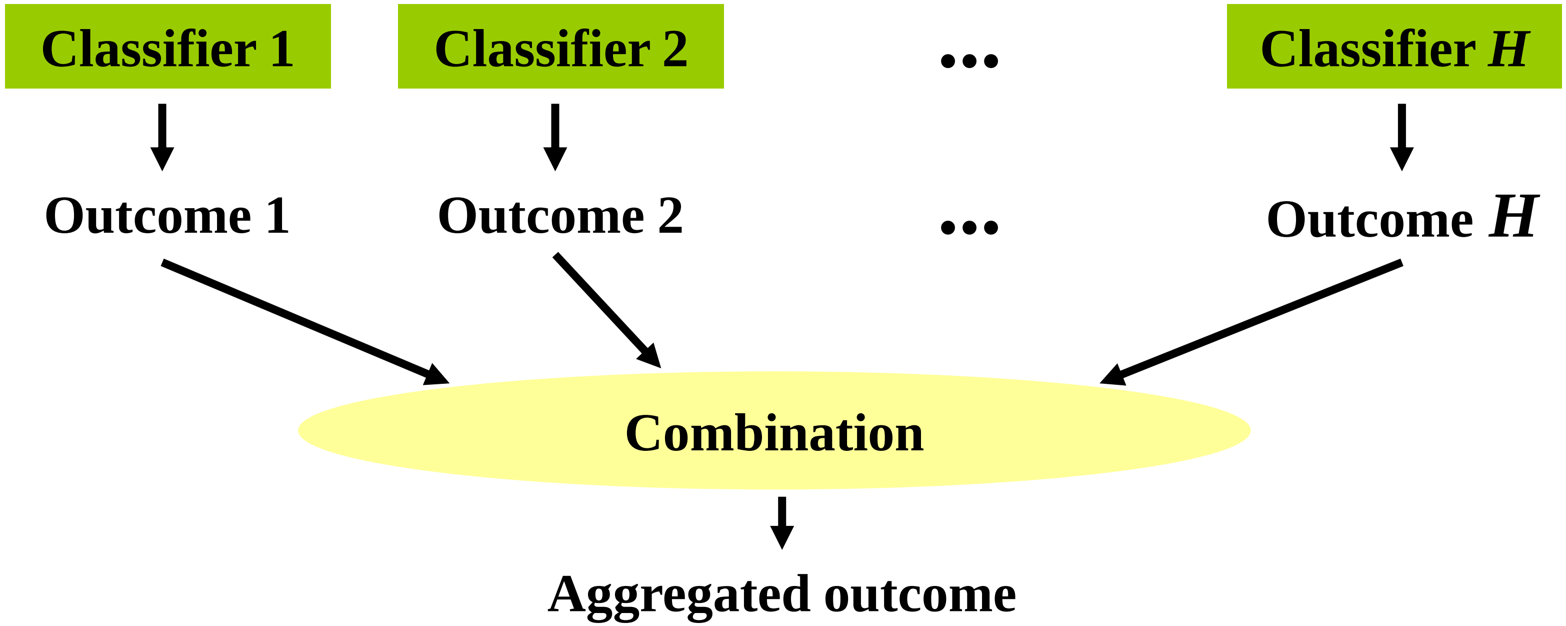
- **Accuracy:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
    - **A. Generate the individual models of the ensemble**
    - **B. Combine the outputs of the individual models**
  - **Application: prognostics of turbofan engines**

A: Generate the Individual Models of the Ensemble

A. Generate the individual models of the ensemble

Classifier 1 **Classifier 2** **…** **Classifier *H***

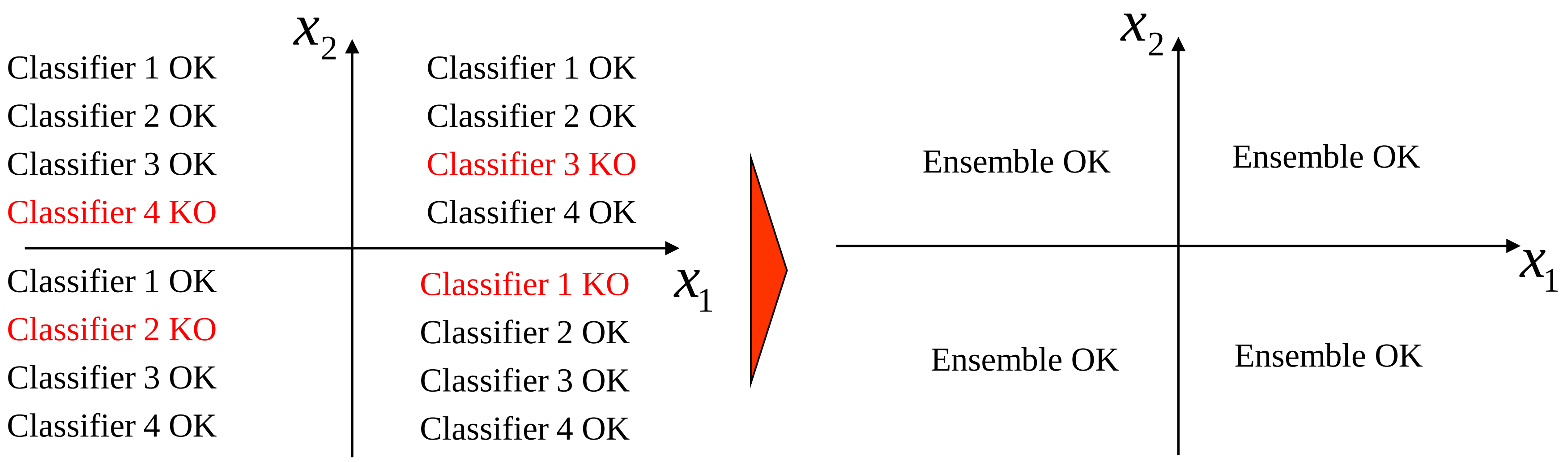
B. Combine the outcomes of the individual classifiers



A: Generate the Individual Models of the Ensemble

- Diversity of the classifiers (individual classifiers make errors on different patterns)

- The combinations improve upon the performance of the single classifier



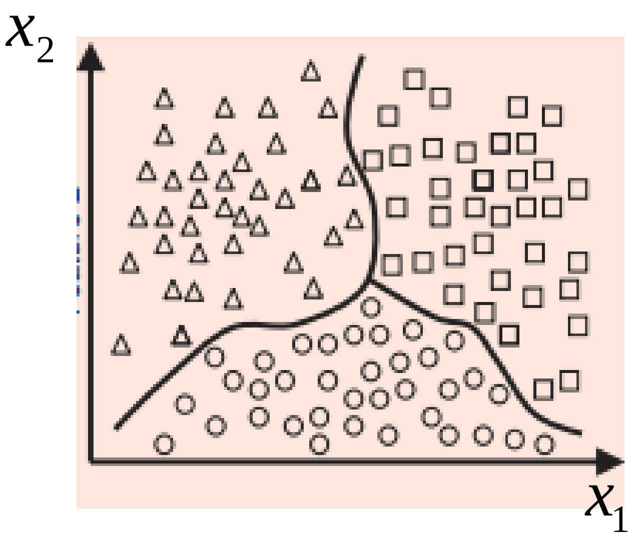
A: Generate the Individual Models of the Ensemble

- Training each classifier with different training patterns:
  - Bagging
  - Boosting
  - Adaboost

A: Generate the Individual Models of the Ensemble

- The bagging algorithm

Training Set: $S = \{(\vec{x}_k, c_k), k = 1, \ldots, N\}$

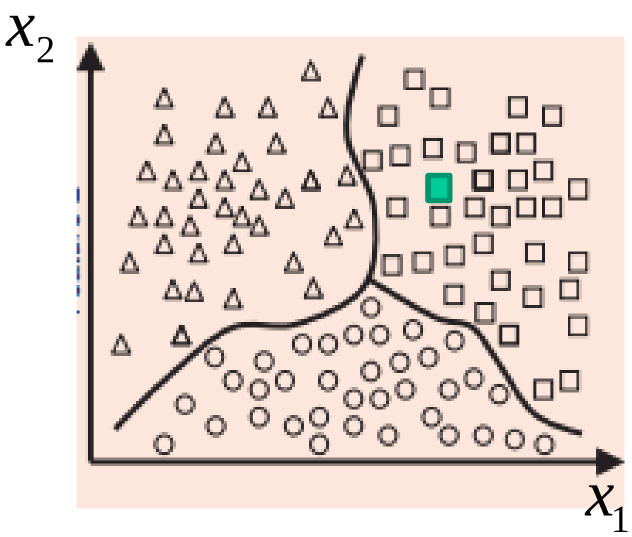


Adapted from: Robi Polikar (2009), Scholarpedia, 4(1):2776 doi:10.4249/scholarpedia.2776
http://www.scholarpedia.org/article/Ensemble_learning

A: Generate the Individual Models of the Ensemble

- The bagging algorithm

Training Set: $S = \{(\vec{x}_k, c_k), k = 1, \ldots, N\}$



% build bootstrap set *1* (S_1)

Do *k*= 1:*N*

- Random sample a patterns of *S*, with replacement

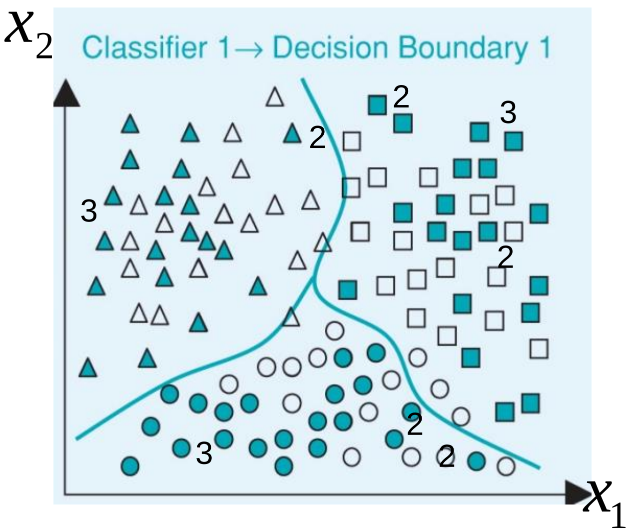
End

Adapted from: Robi Polikar (2009), Scholarpedia, 4(1):2776 doi:10.4249/scholarpedia.2776
http://www.scholarpedia.org/article/Ensemble_learning

A: Generate the Individual Models of the Ensemble

➢ The bagging algorithm

Training Set: $S = \{(\vec{x}_k, c_k), k = 1,\ldots, N\}$



% build bootstrap set 1 (S_1)

Do *k*= 1:*N*

- Random sample a patterns of *S*, with replacement

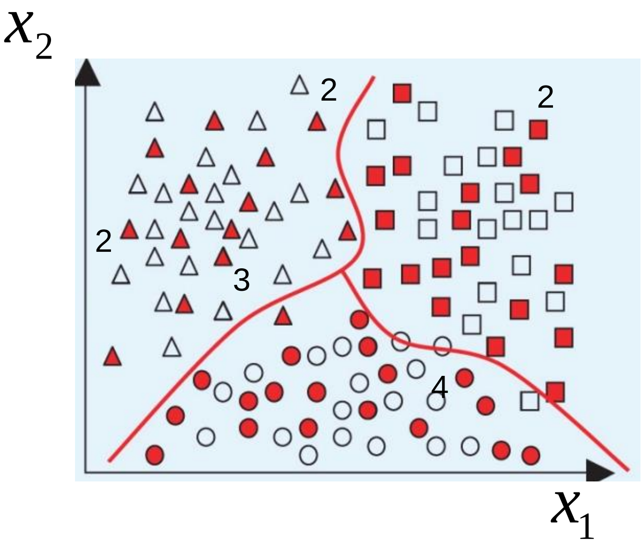
End

- Train a classifier on S_1
- Add the classifier to the ensemble

Adapted from: Robi Polikar (2009), Scholarpedia, 4(1):2776 doi:10.4249/scholarpedia.2776
http://www.scholarpedia.org/article/Ensemble_learning

➢ The bagging algorithm

Training Set: $S = \{(\vec{x}_k, c_k), k = 1,\ldots, N\}$



Do *i*=1,..,*H*

% build bootstrap set *i* (S_i)

Do *k*= 1:*N*

- Random sample a patterns of *S*, with replacement

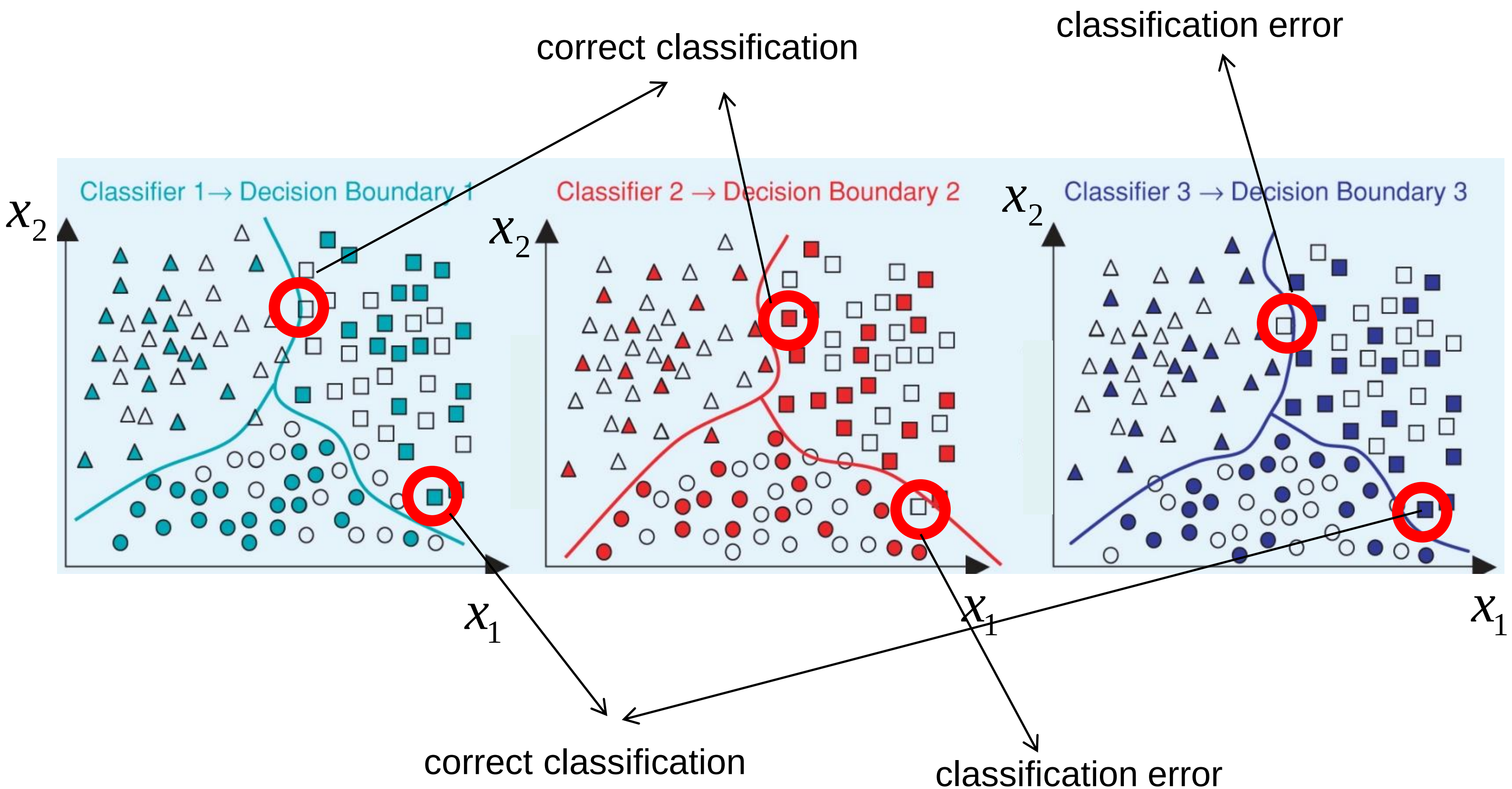
End

- Train a classifier on S_i
- Add the classifier to the ensemble

End

Adapted from: Robi Polikar (2009), Scholarpedia, 4(1):2776 doi:10.4249/scholarpedia.2776
http://www.scholarpedia.org/article/Ensemble_learning

A: Generate the Individual Models of the Ensemble



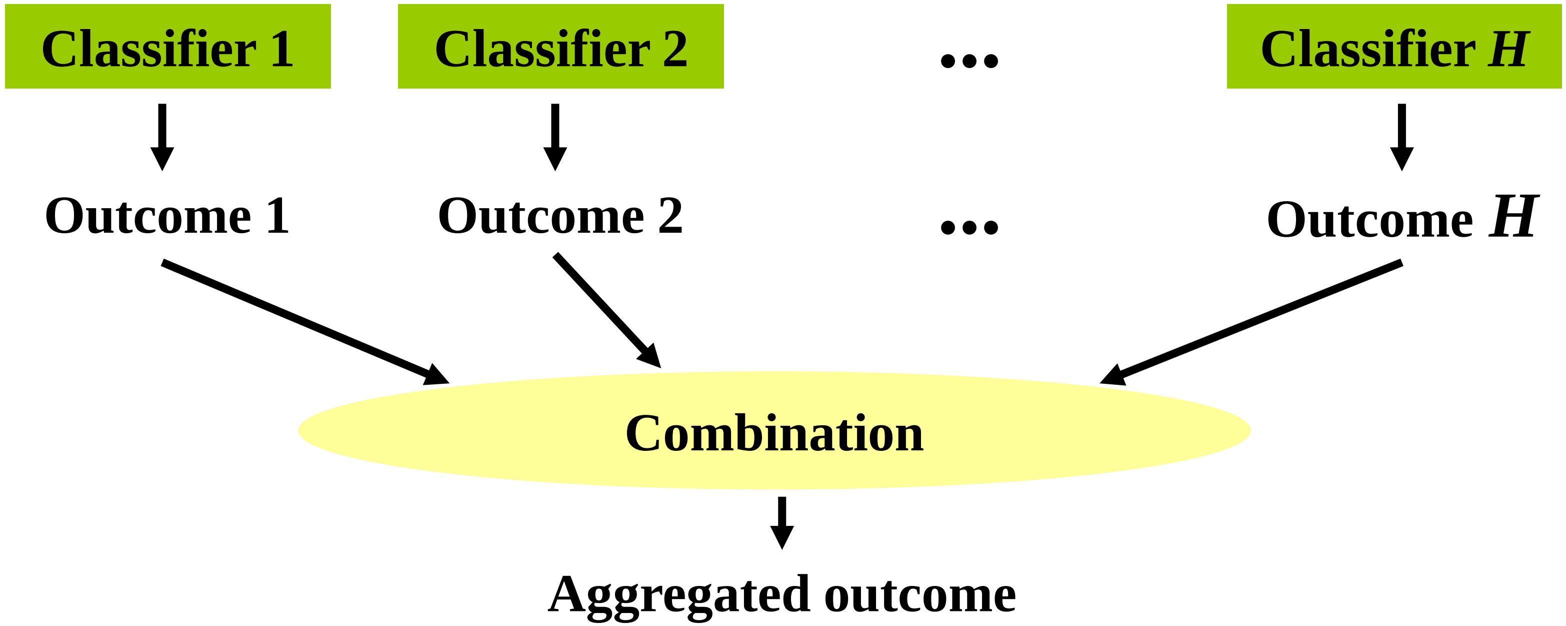
Adapted from: Robi Polikar (2009), Scholarpedia, 4(1):2776 doi:10.4249/scholarpedia.2776
http://www.scholarpedia.org/article/Ensemble_learning

- **Accuracy:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
    - A. **Generate the individual models of the ensemble**
    - B. **Combine the outputs of the individual models**
  - **Application: prognostics of turbofan engines**

B: Combine the Outcomes of the Individual Models

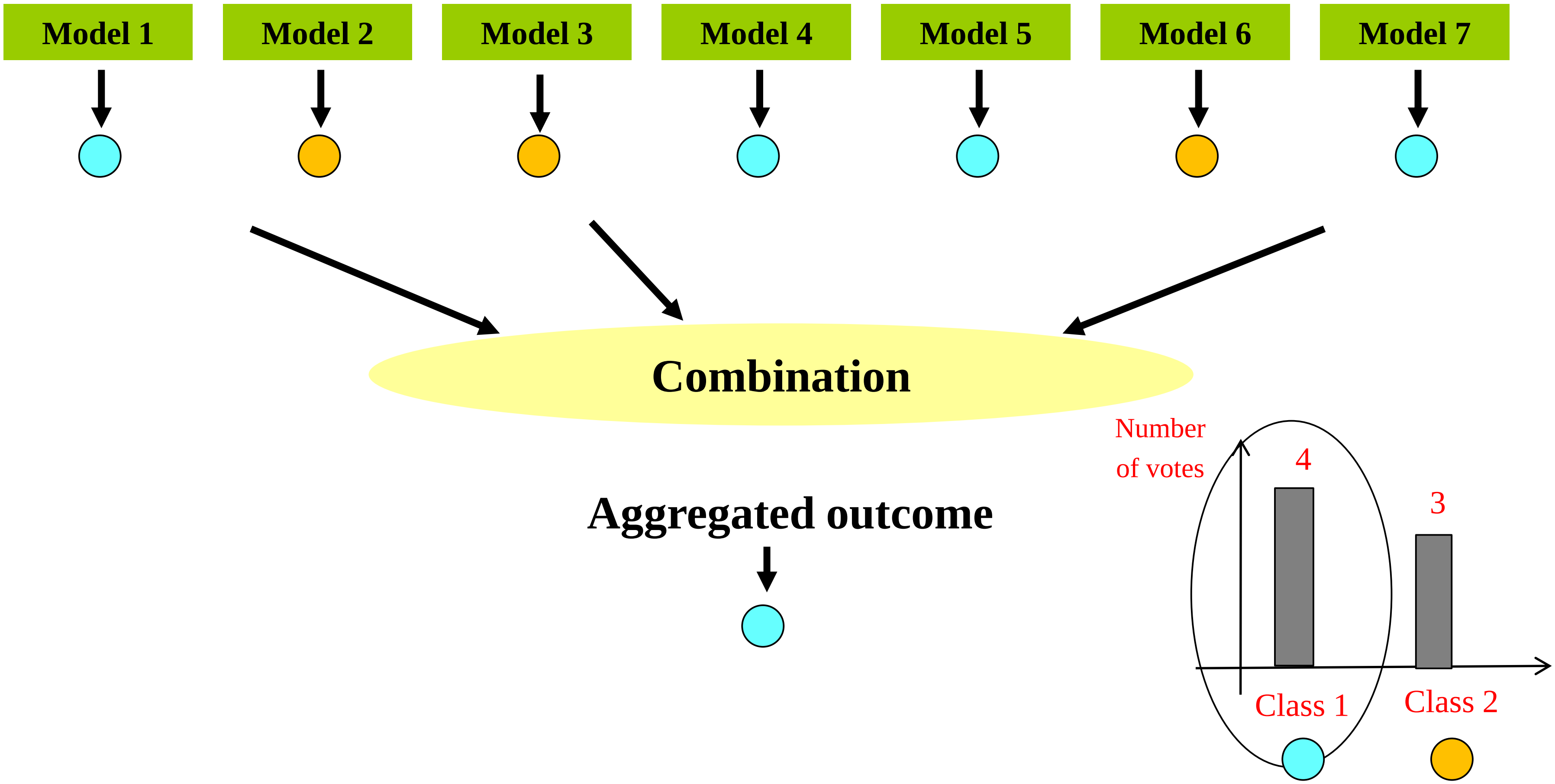
A. Generate the individual models of the ensemble

B. Combine the outcomes of the individual classifiers



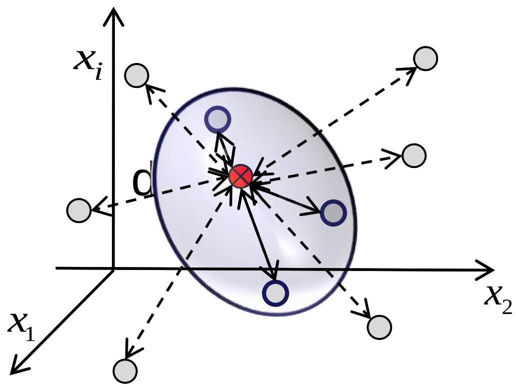
B: Combine the Outcomes of the Individual Models

- **Simple majority voting**: choose the class with the associated largest number of votes



B: Combine the Outcomes of the Individual Models

- **Dynamic Approach**:



Test pattern:
Validation set:

Step 1: Identify the k-nearest neighbours of the test pattern in a validation set

Step 2: Estimate local performance of the ensemble individual classifiers on the k-nearest neighbours

Model vote ∝ local performance

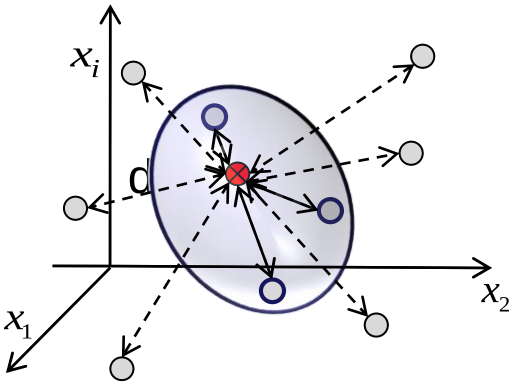
classifier	1	…	h	…	H
1st nearest neighbor	OK		OK		KO
2nd nearest neighbor	KO		OK		KO
3rd nearest neighbor	OK		OK		OK
Model vote ∝ Local performance	2/3		1		1/3

Step 3: Assign the test pattern to the class with the associated largest vote

classifier	1	…	h	…	H
vote	2/3		1		1/3
Class assigned to the test pattern					

B: Combine the Outcomes of the Individual Models

- **Dynamic Approach**:
  - model vote depends from the test pattern position
  - model *h* vote proportional to model *h* classification performance on patterns similar to the test pattern

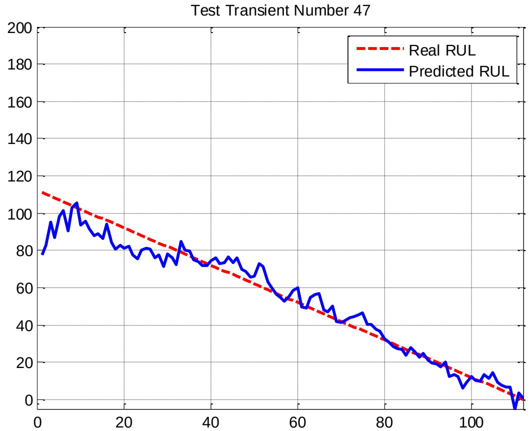


Test pattern:
Validation set:

- **Accuracy:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
    - **A. Generate the individual models of the ensemble**
    - **B. Combine the outputs of the individual models**
  - **Application: prognostics of turbofan engines**

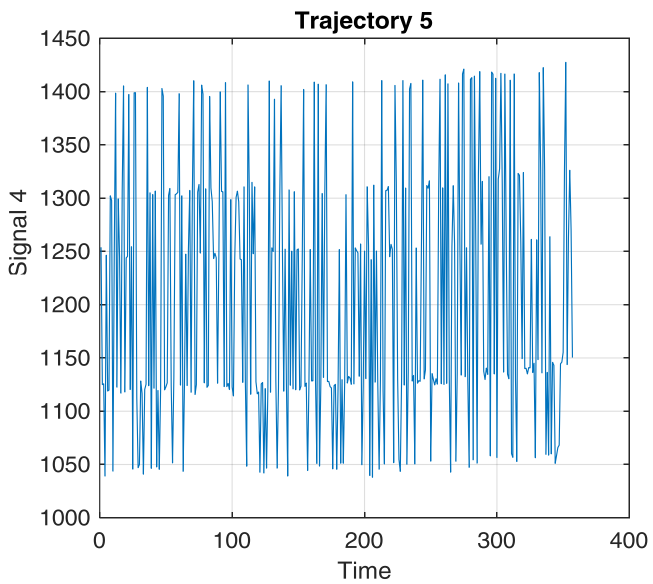
Application: Prognostics of Turbofan Engines

Ensemble of Echo-State Networks

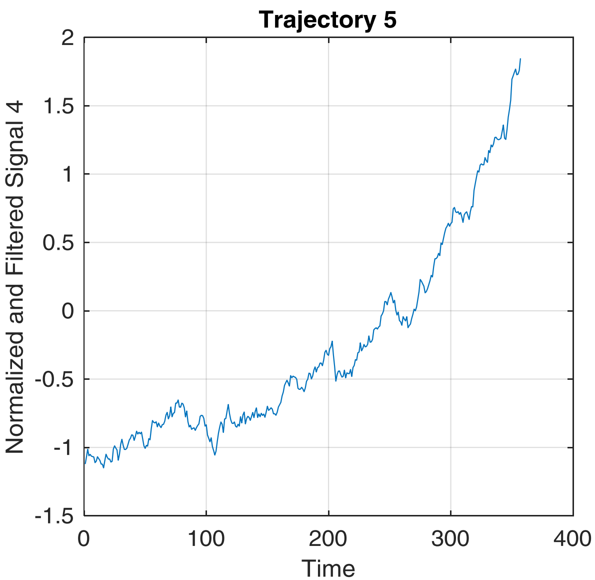


Application: Prognostics of Turbofan Engines

- The C-MAPPS dataset*
  - 260 run-to-failure trajectories
  - 21 measured signals + 3 signals representative of the operating conditions
  - 6 different operating conditions



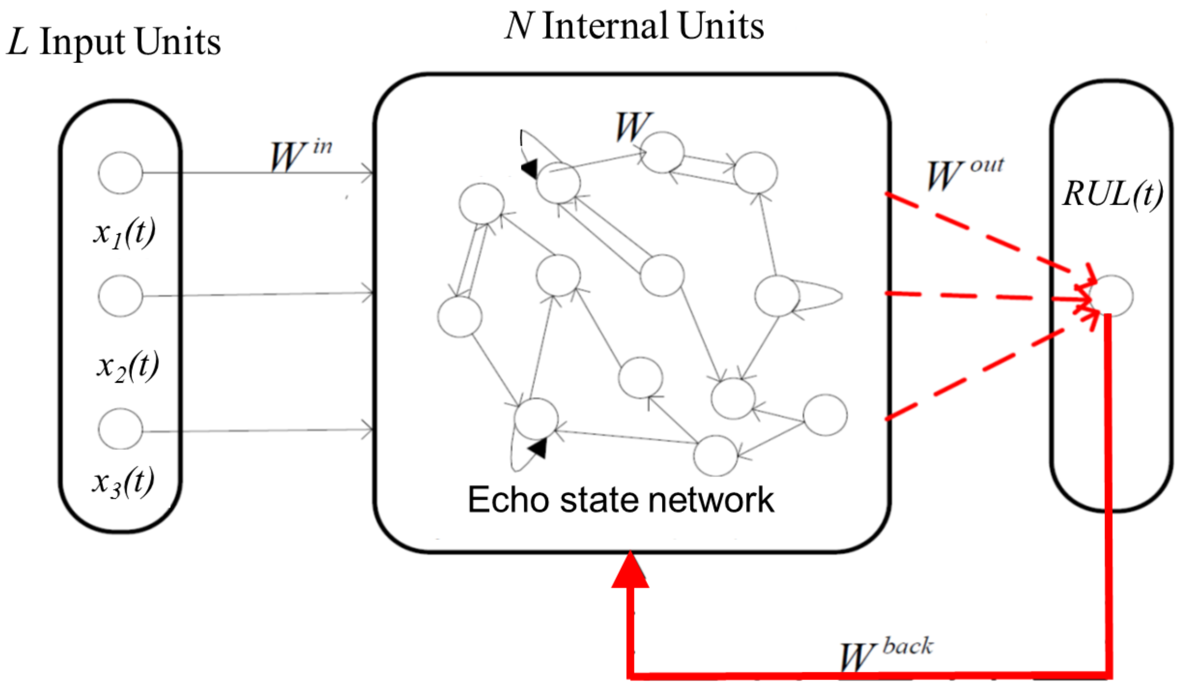
Data Preprocessing**



* A. Saxena, K. Goebel, D. Simon, N. Eklund, *Damage propagation modeling for aircraft engine run-to-failure simulation*, PHM2008
**M. Rigamonti, P. Baraldi, E. Zio, I. Roychoudhury, K. Goebel, S. Poll, *Echo State Network for Remaining Useful Life Prediction of a Turbofan Engine*, PHM 2016, Bilbao

Application: Prognostics of Turbofan Engines

➢ Ensemble individual model: echo state network*



Advantages

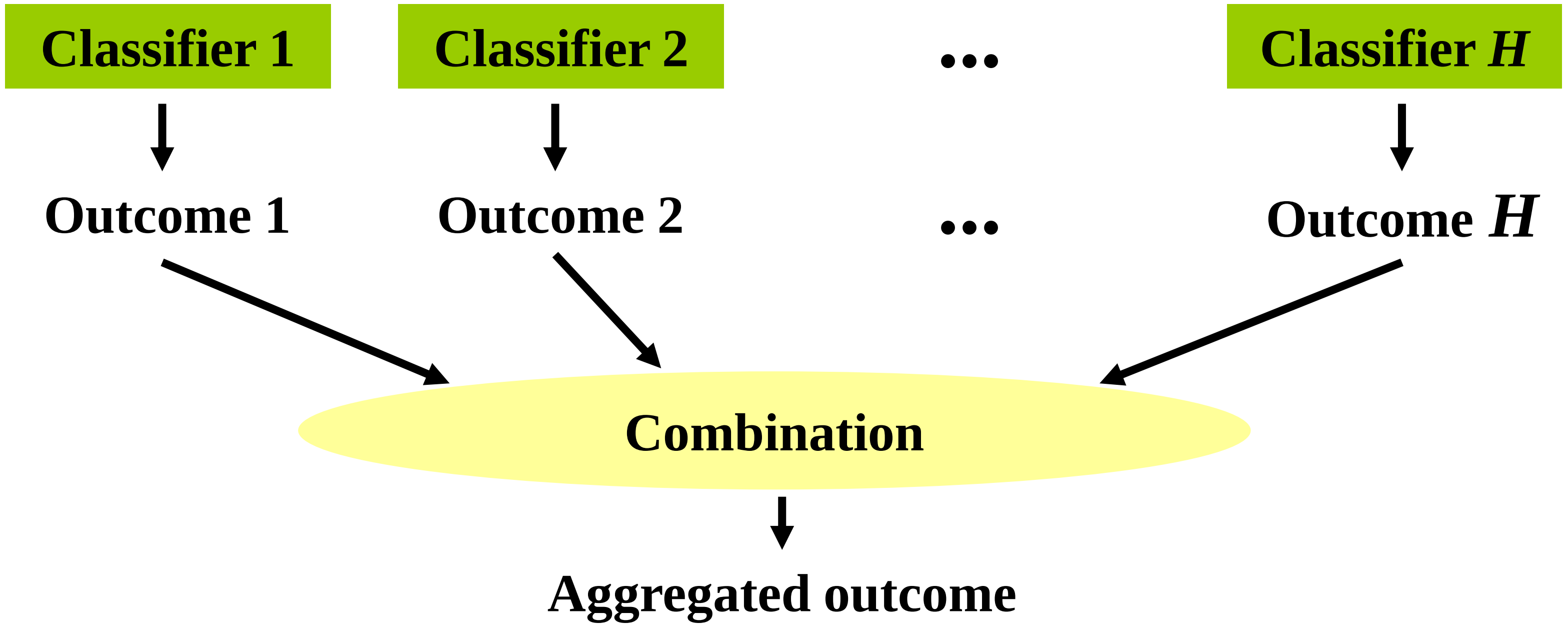
- Non linear modeling
- Memory Property
- Intrinsic dynamic characteristics

*M. Rigamonti, P. Baraldi, E. Zio, I. Roychoudhury, K. Goebel, S. Poll, *Echo State Network for Remaining Useful Life Prediction of a Turbofan Engine*, PHM 2016, Bilbao

A. Generate the individual models of the ensemble

Classifier 1 **Classifier 2** … **Classifier *H***

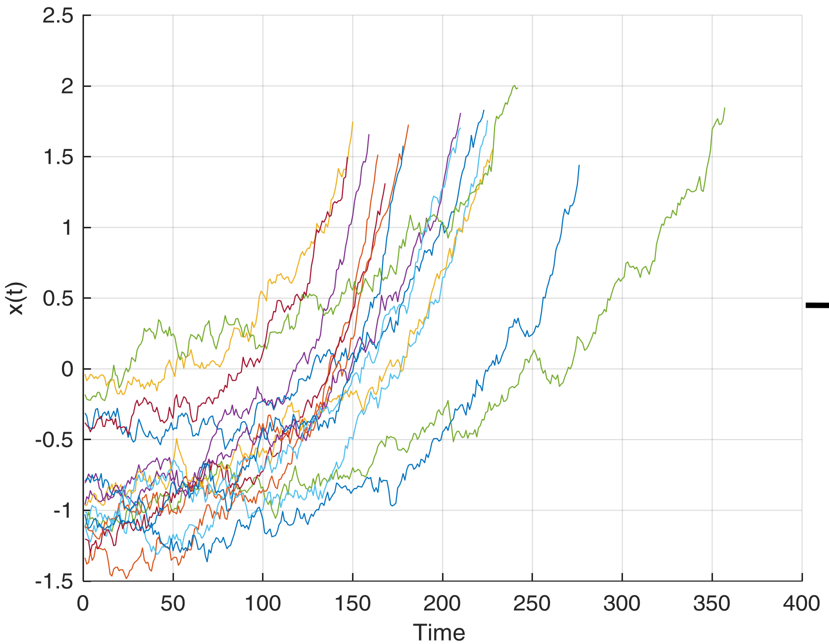
B. Combine the outcomes of the individual classifiers



Application: Prognostics of Turbofan Engines

➢ Generate the Individual Models of the Ensemble: Bagging

70 run-to-failure training trajectories



Bagging Set 1
Trajectory:
2, 15, 24, 54, 68

Echo State Network 1

Bagging Set 2
Trajectory:
5, 11, 44, 54, 55

Echo State Network 2

…

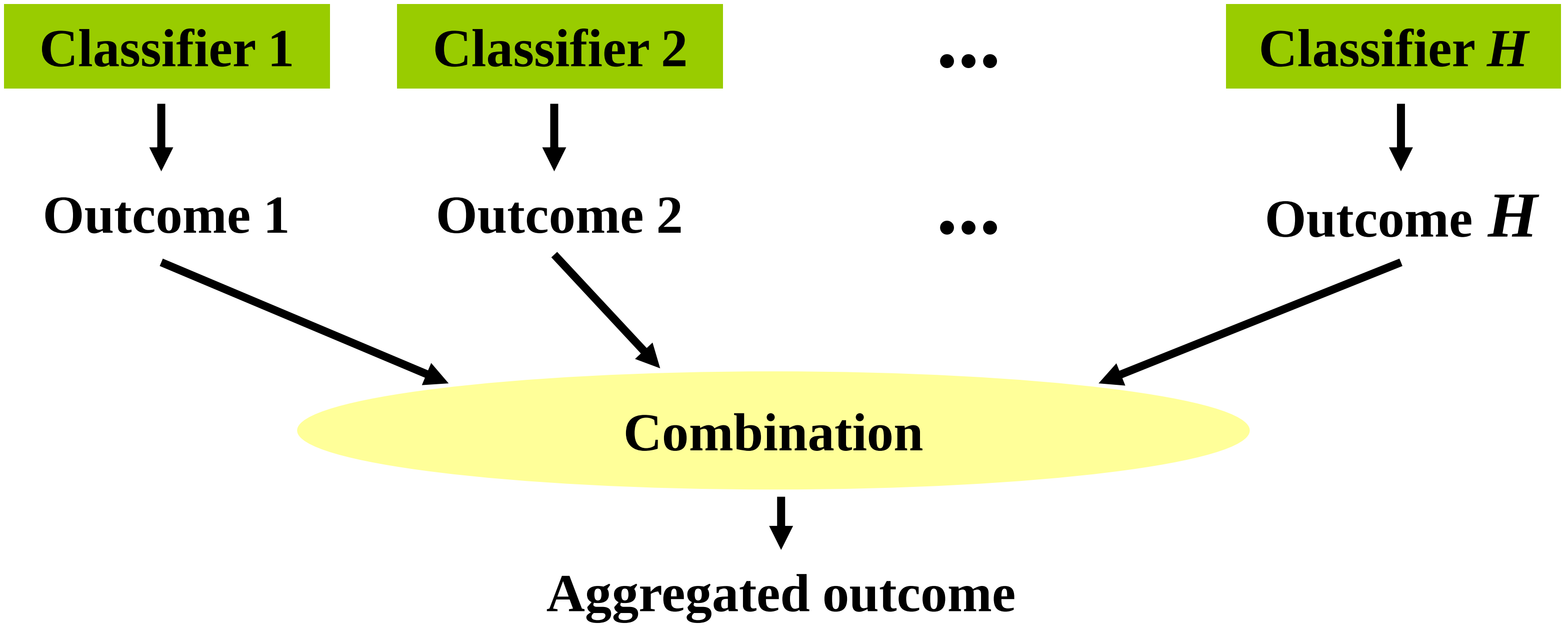
…

Bagging Set *H*
Trajectory:
6, 35, 39, 44, 68

Echo State Network *H=25*

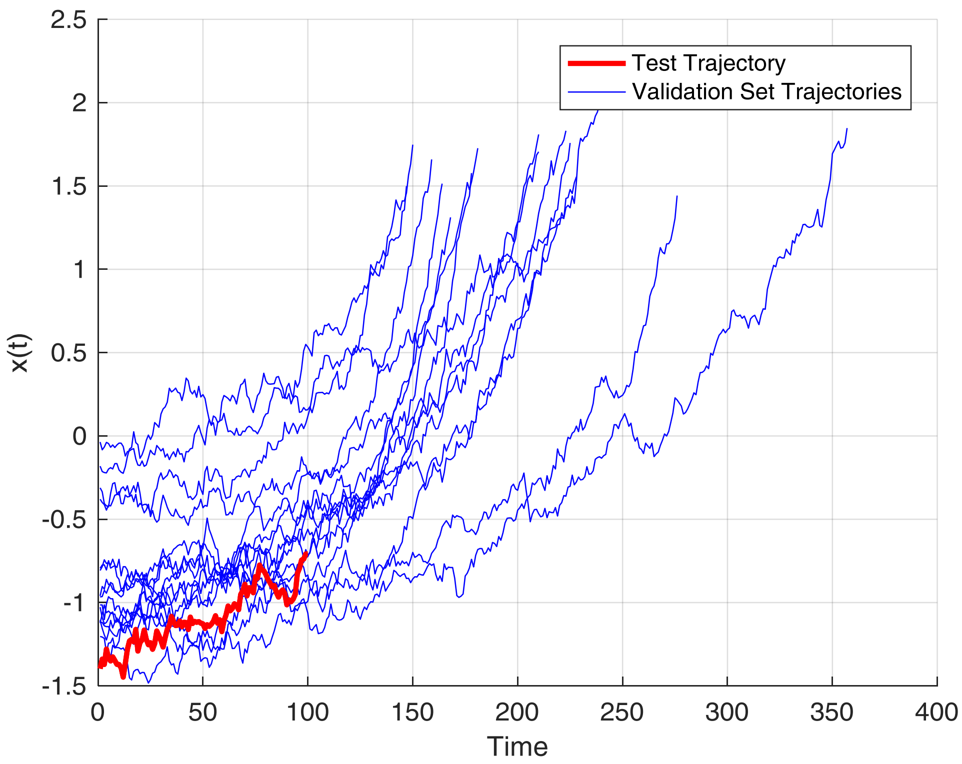
A. Generate the individual models of the ensemble

B. Combine the outcomes of the individual classifiers



Application: Prognostics of Turbofan Engines

➢ Combine the outputs of the individual models: dynamic approach

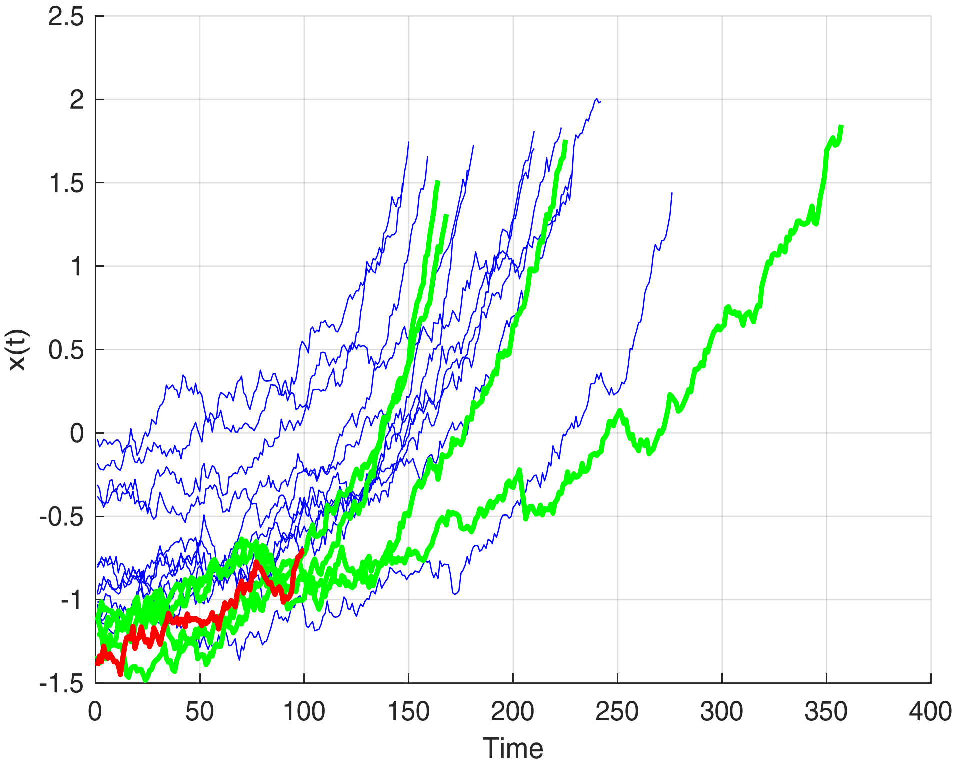


- **Identify the *K* trajectories of the validation set most similar to the test trajectory***

*Zio, E., Di Maio, F. A data-driven fuzzy approach for predicting the remaining useful life in dynamic failure scenarios of a nuclear system (2010) Reliability Engineering and System Safety, 95 (1), pp. 49-57.

Application: Prognostics of Turbofan Engines

➢ Combine the outputs of the individual models: dynamic approach



- **Identify the *K* trajectories of the validation set most similar to the test trajectory**

Application: Prognostics of Turbofan Engines

- Combine the outputs of the individual models: dynamic approach

	ESN 1	ESN 2	…	ESN *H*
RUL Prediction error on the most similar trajectory	3.11	1.88	…	2.01
RUL Prediction error on the 2nd most similar trajectory	1.89	13.13	…	3.90
RUL Prediction error on the 3rd most similar trajectory	2.00	6,99		5,09

- Identify the K trajectories of the validation set most similar to the test trajectory
- **Estimate the individual ESN error on the K trajectories of the validation set most similar to the test trajectory**

Application: Prognostics of Turbofan Engines

- Combine the outputs of the individual models: dynamic approach

	ESN 1	**ESN 2**	**…**	**ESN *H***
RUL Prediction error on the most similar trajectory	3.11	1.88	…	2.01
RUL Prediction error on the 2nd most similar trajectory	1.89	13.13	…	3.90
RUL Prediction error on the 3rd most similar trajectory	2.00	6,99		5,09
Local Error (LE)	**2.66**	**7.33**	**…**	**5.00**

- Identify the K trajectories of the validation set most similar to the test trajectory
- Estimate the individual ESN error on the K trajectories of the validation set most similar to the test trajectory
- **Estimate the individual ESN local errors as average of the individual ESN error on the *K* most similar trajectories**

Application: Prognostics of Turbofan Engines

- Combine the outputs of the individual models: dynamic approach

	ESN 1	ESN 2	…	ESN *H*
RUL Prediction error on the most similar trajectory	3.11	1.88	…	2.01
RUL Prediction error on the 2nd most similar trajectory	1.89	13.13	…	3.90
RUL Prediction error on the 3rd most similar trajectory	2.00	6,99		5,09
Local Error (LE)	**2.66**	**7.33**	**…**	**5.00**
Model Weight	**0.52**	**0.19**		**0.29**

- Identify the *K* trajectories of the validation set most similar to the test trajectory
- Estimate the individual ESN error on the *K* trajectories of the validation set most similar to the test trajectory
- Estimate the individual ESN local error as average of the individual ESN errors on the k most similar trajectories
- **Assign a weight to each model:**

$$\omega_i = \frac{\frac{1}{LE_i}}{\sum_{j=1}^{H} \frac{1}{LE_j}}$$

➢ Combine the outputs of the individual models: dynamic approach

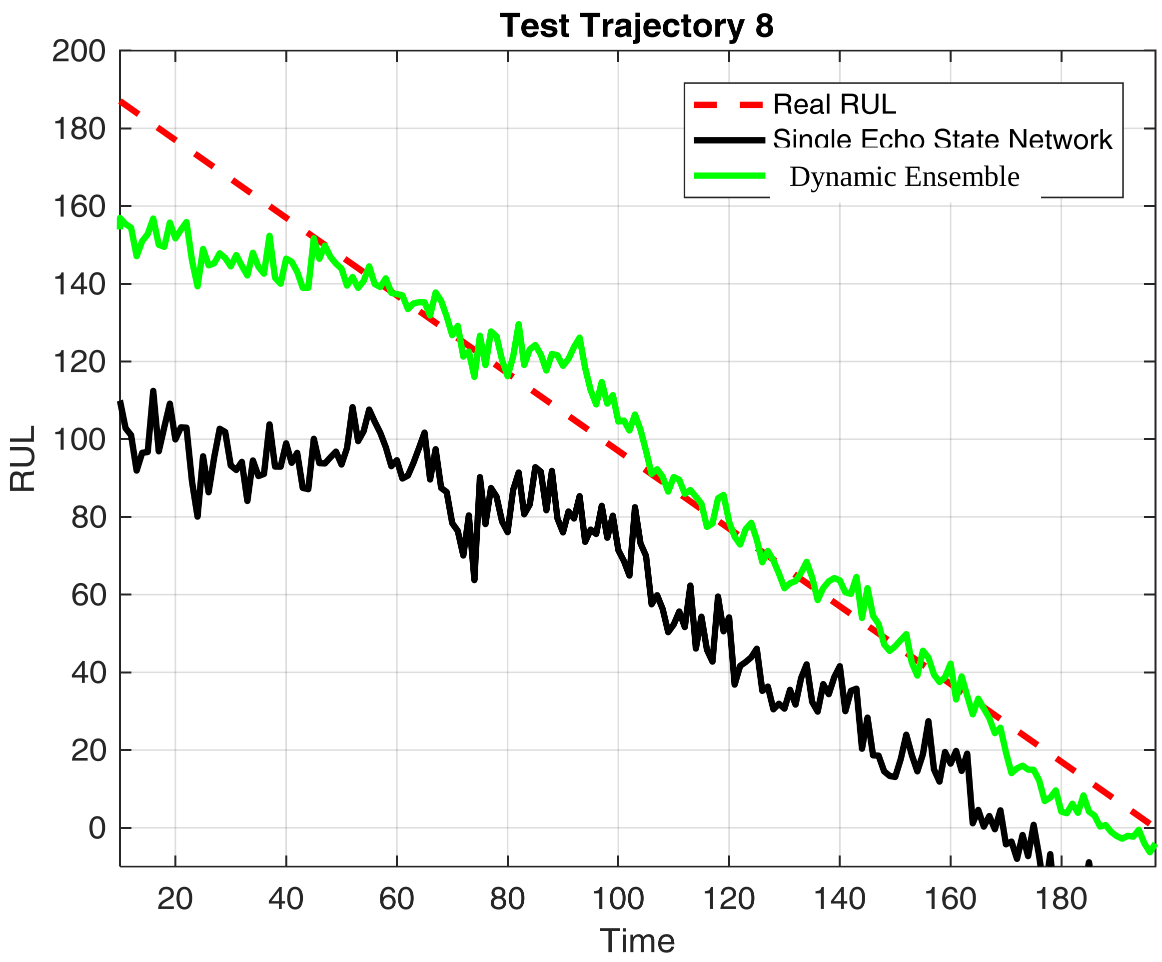
	ESN 1	ESN 2	…	ESN *H*
RUL Prediction error on the most similar trajectory	3.11	1.88	…	2.01
RUL Prediction error on the 2nd most similar trajectory	1.89	13.13	…	3.90
RUL Prediction error on the 3rd most similar trajectory	2.00	6,99		5,09
Local Error (LE)	**2.66**	**7.33**	**…**	**5.00**
Model Weight	**0.52**	**0.19**		**0.29**

- Identify the K trajectories of the validation set most similar to the test trajectory
- Estimate the individual ESN errors on the K trajectories of the validation set most similar to the test trajectory
- Estimate the individual ESN local errors as average of the individual ESN errors on the K most similar trajectories
- Assign a weight to each model
- **Compute the RUL prediction as weighted sum of the individual model RUL predictions**:

$$RUL = \sum_{i=1}^{H} \omega_i RUL_i$$

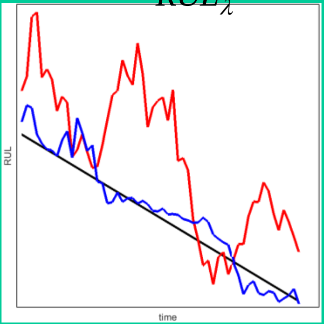
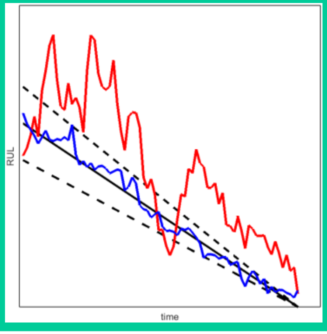
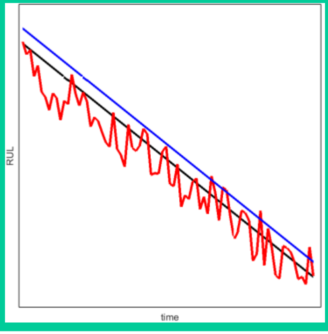
Application: Prognostics of Turbofan Engines

➢ Results



Application: Prognostics of Turbofan Engines

➢ Results – Prognostic Metrics (70 test trajectories)

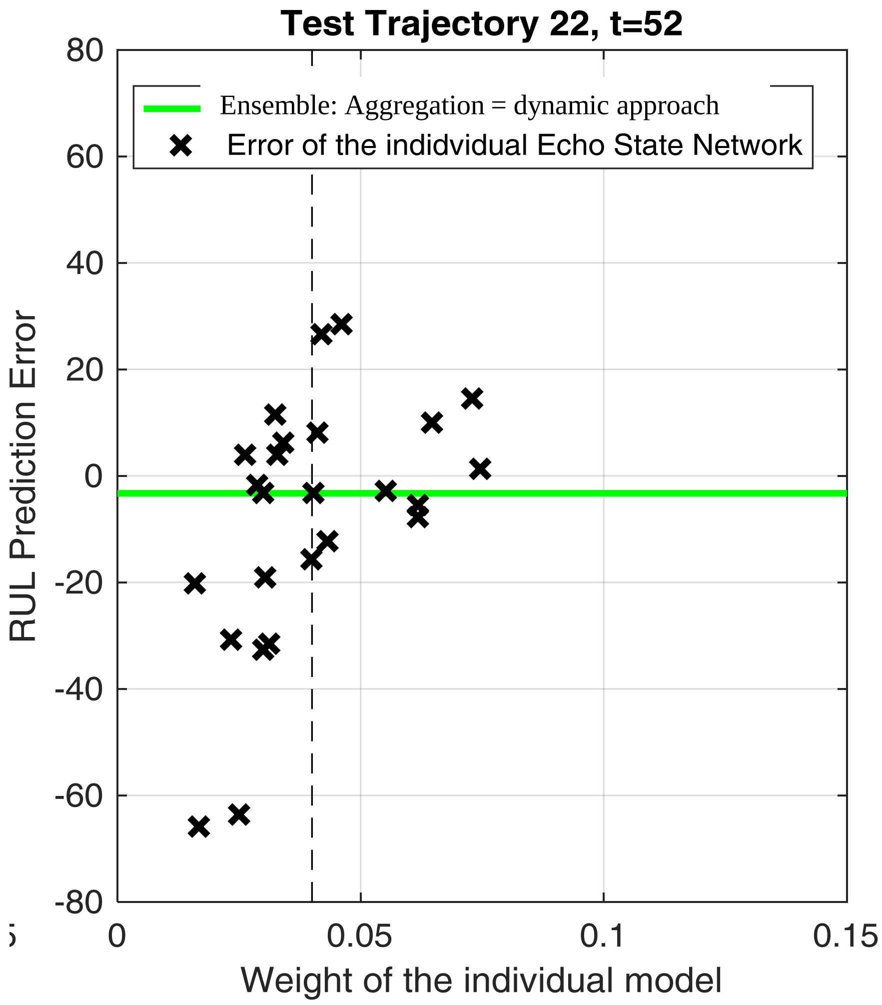
	Cumulative Relative Accuracy $RA_\lambda = \frac{\left\lvert R\hat{U}L_\lambda - RUL_\lambda \right\rvert}{RUL_\lambda}$ 	**Alpha-Lambda** $\boldsymbol{\alpha = 0.2}$ 	**Steadiness** $SI_t = \sqrt{\mathrm{var}(\overline{T}_{(t-\Delta t):t})},$ 
Individual Echo State Network	0.476 ± 0.041	0.390 ± 0.052	4.968 ± 1.213
Ensemble (aggregation = simple average)	0.417 ± 0.031	0.400 ± 0.049	4.444 ± 0.326
Ensemble (aggregation = dynamic approach)	0.369 ± 0.029	0.421 ± 0.028	4.414 ± 0.319

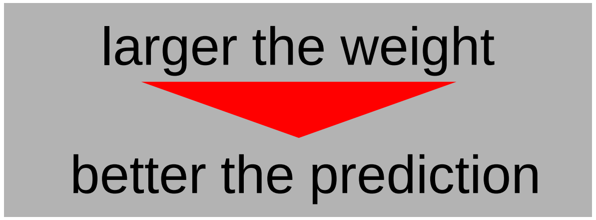
Application: Prognostics of Turbofan Engines

- Results – Comparison between different approaches for the aggregation of the individual model outputs

Application: Prognostics of Turbofan Engines

➢ Results – Analysis of the individual model weights





Tutorial Contents

- Ensembles of Models for Prognostics and Health Management (PHM):
  - Accuracy
  - Confidence Estimation
  - Incremental Learning

- **Confidence estimation:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
  - **Application: prognostics of turbine blades**

Confidence estimation: what is it?

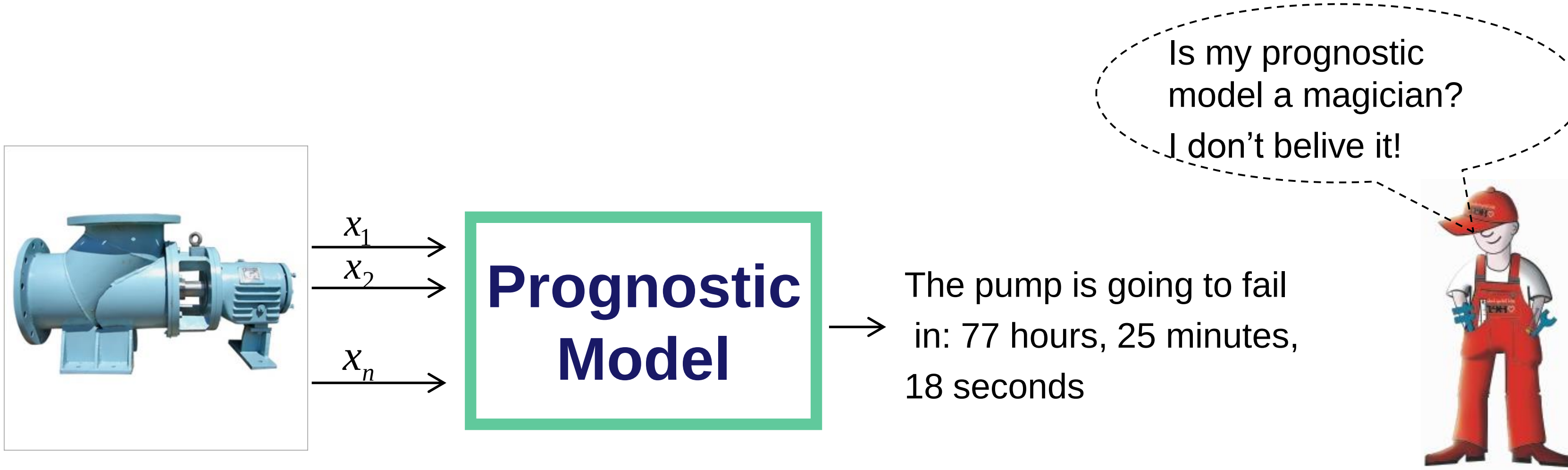
Is my prognostic model a magician?
I don't belive it!

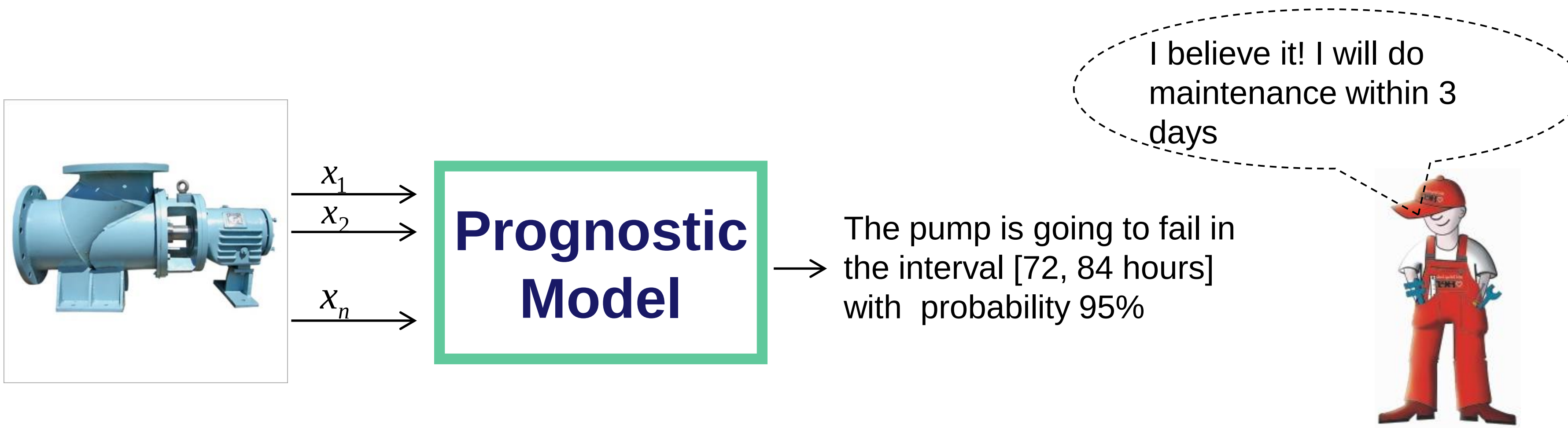
x_1
x_2
x_n

Prognostic Model

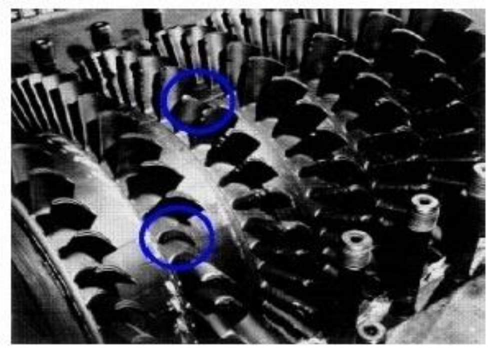
The pump is going to fail in: 77 hours, 25 minutes, 18 seconds

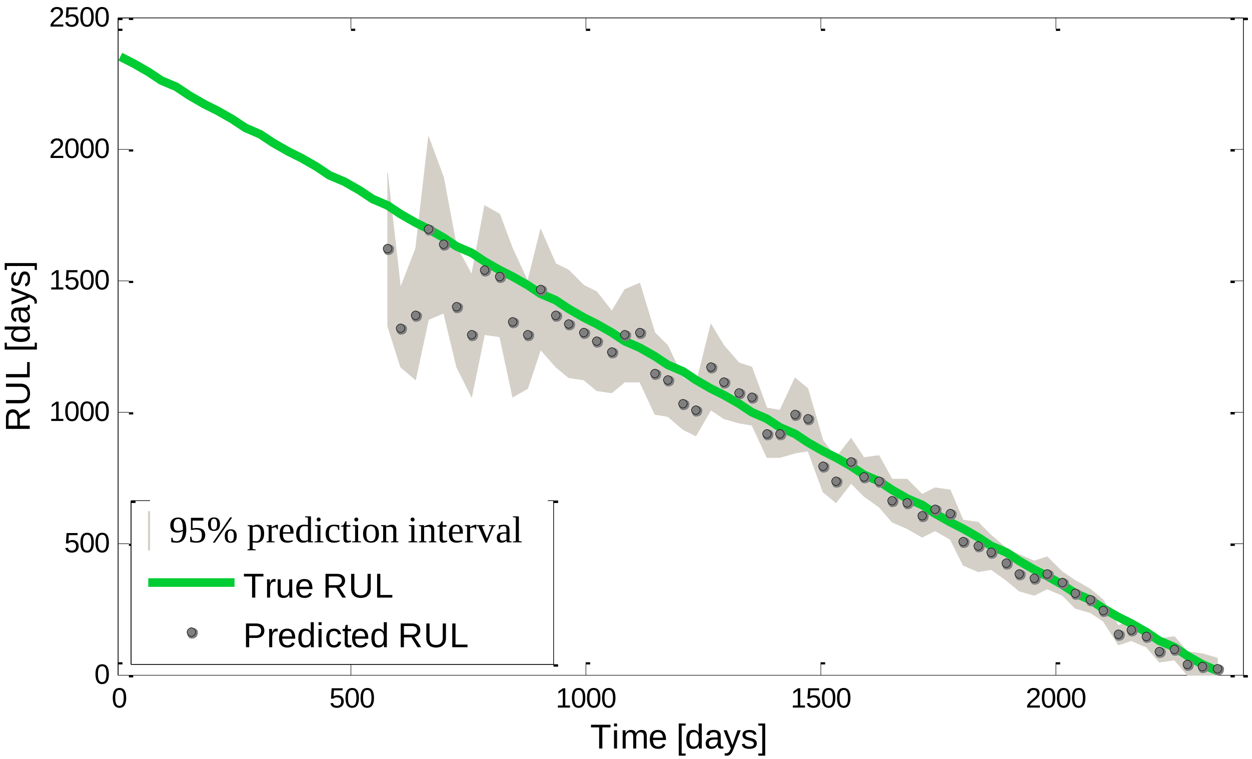
Confidence estimation: what is it?





Confidence estimation: what is it?

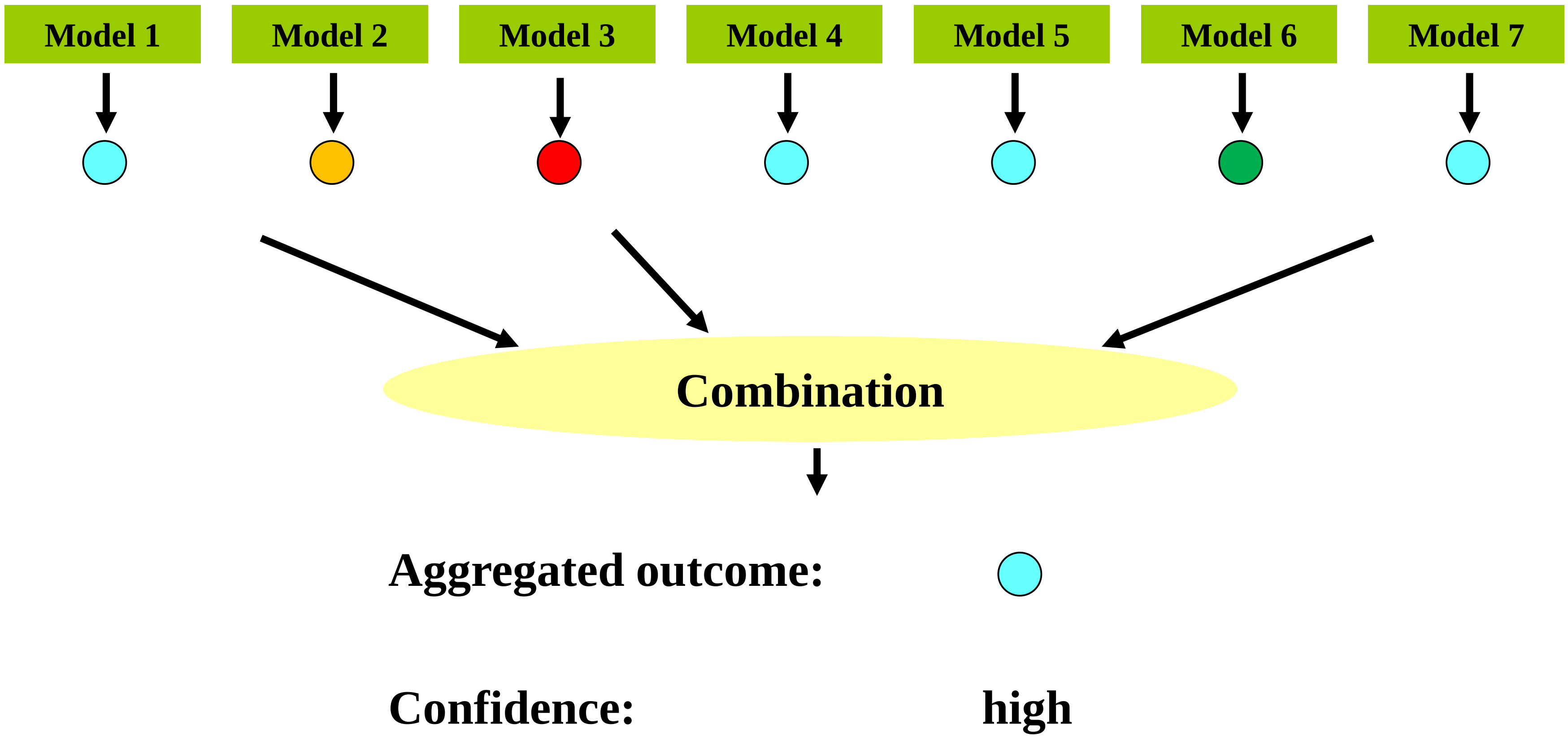




- **Confidence estimation:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
  - **Application: prognostics of turbine blades**

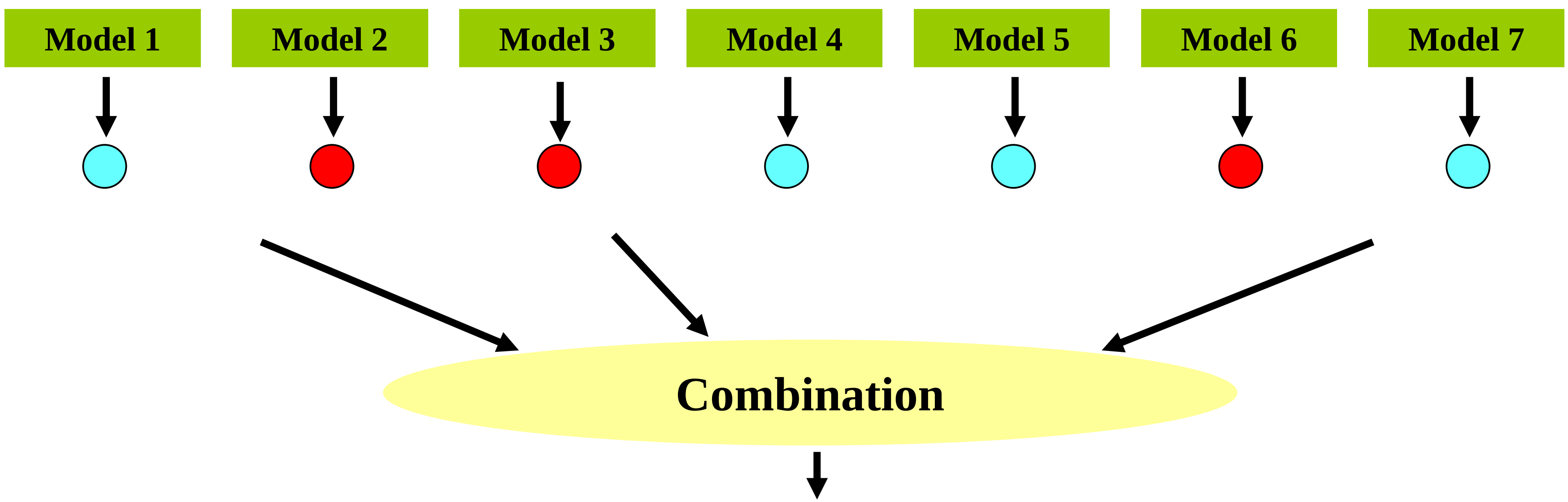
Why ensemble? [Diagnostics]

- Single model outcome distribution → information on the confidence



Why ensemble? [Diagnostics]

- Single model outcome distribution → information on the confidence



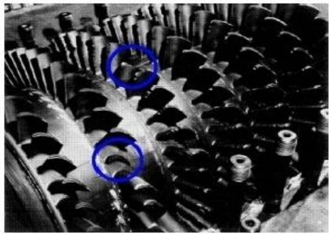
Aggregated outcome:

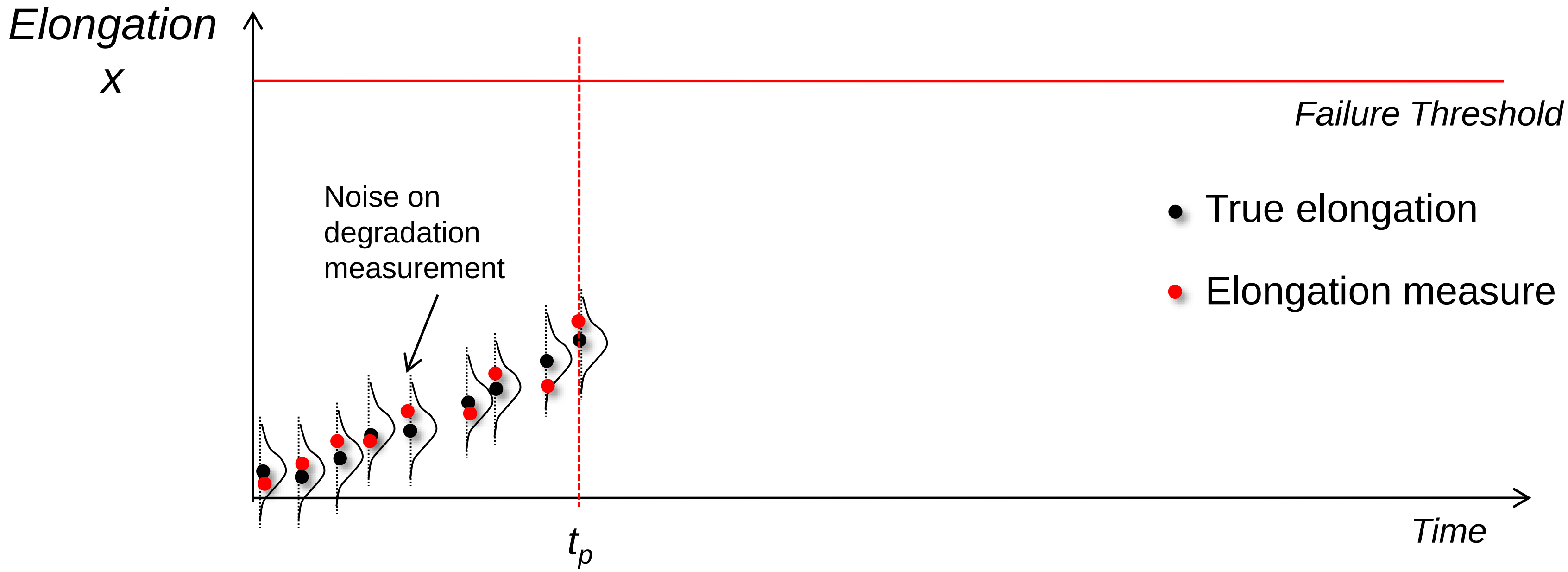
Confidence: Low (it could also be)

- **Confidence estimation:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
  - **Application: prognostics of turbine blades**

How? [prognostics]

- Sources of Uncertainty in Prognostics
  1) noise on the observations (measurements)



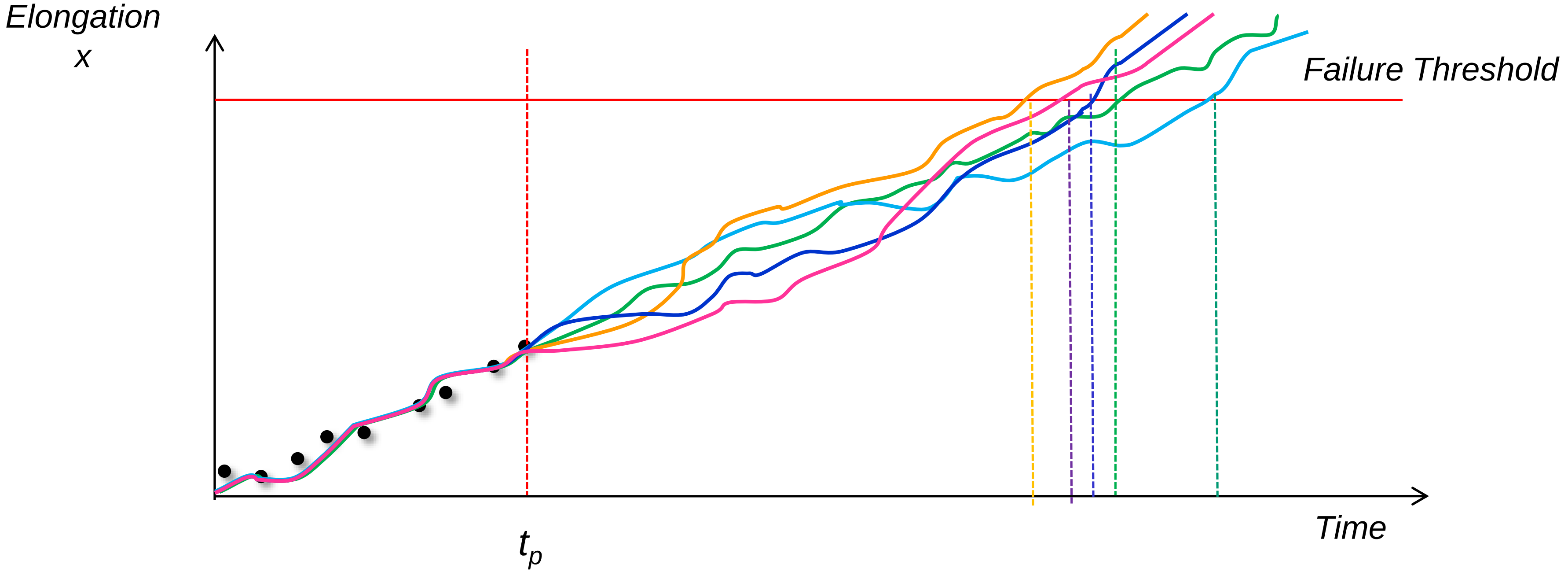


How? [prognostics]

- Sources of Uncertainty in Prognostics

  1) noise on the observations (measurements)

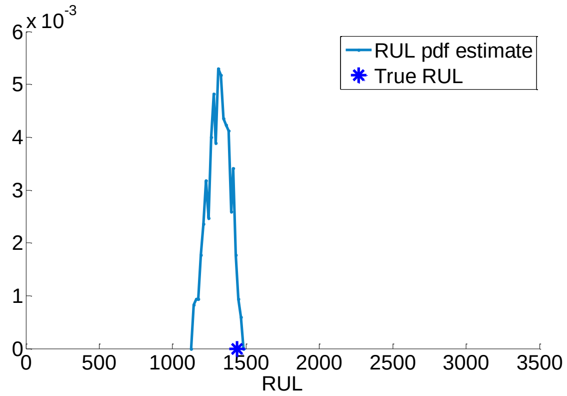
  2) Intrinsic stochasticity due to the degradation process and unknown future external/operational conditions (process)



How? [prognostics]

- Sources of Uncertainty in Prognostics
  1) noise on the observations (*measurements*)
  2) intrinsic stochasticity due to the degradation process and unknown future external/operational conditions (*process*)
  3) modeling errors, i.e. inaccuracy of the PHM model used to perform the prediction (*model*)

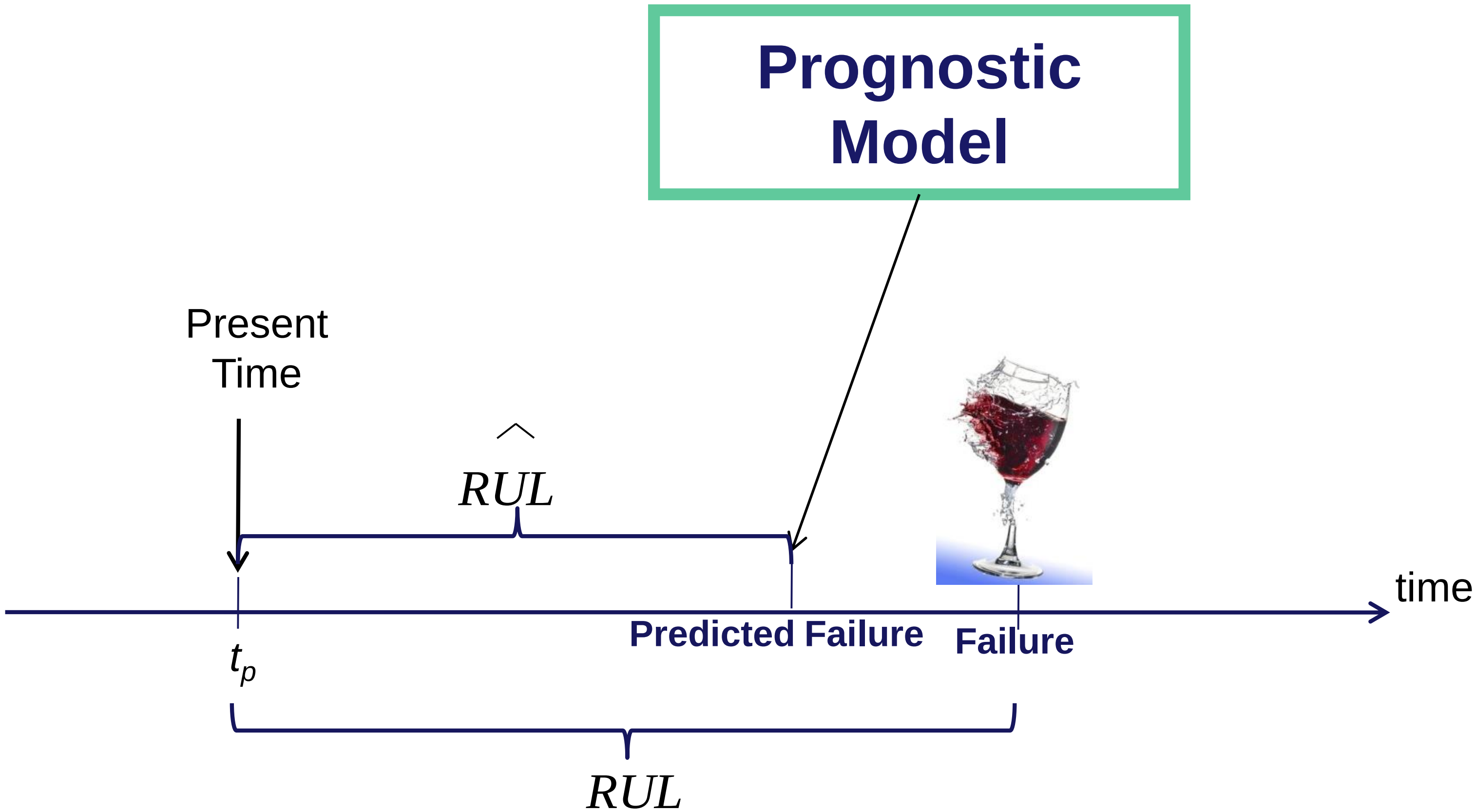
How to quantify uncertainty on the RUL prediction ???



How? [prognostics]

- Prediction error:

$$\varepsilon = rul - \hat{rul}$$



How? [prognostics]

- Prediction error:

$$\varepsilon = rul - r\hat{u}l$$

Random Variable!

- Assumptions:
  - $E[\varepsilon] = 0$ (the prognostic model is unbiased)
  - ε has a normal distribution

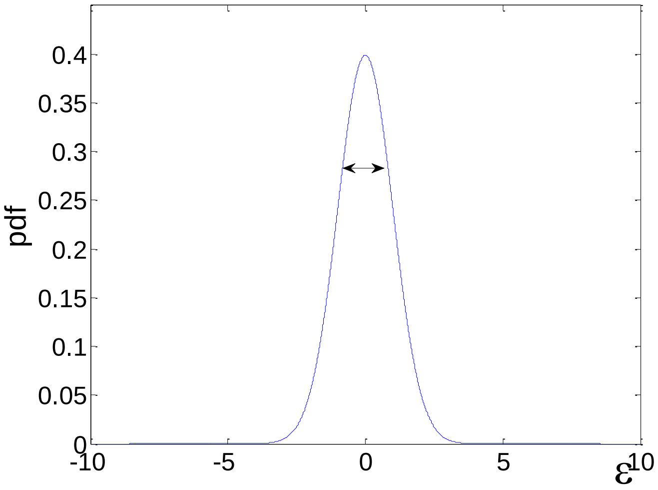
How? [prognostics]

- Prediction error:

$$\varepsilon = rul - r\hat{u}l$$

- Assumptions:
  - $E[\varepsilon] = 0$ (the prognostic model is unbiased)
  - ε has a normal distribution

- Objective: $\mathrm{V}ar[\varepsilon] = \sigma_\varepsilon^2 = E[(rul - r\hat{u}l\,)^2]$



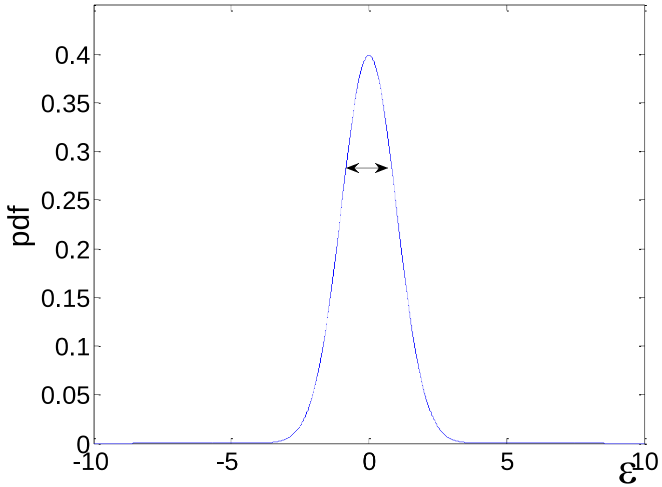
How? [prognostics]

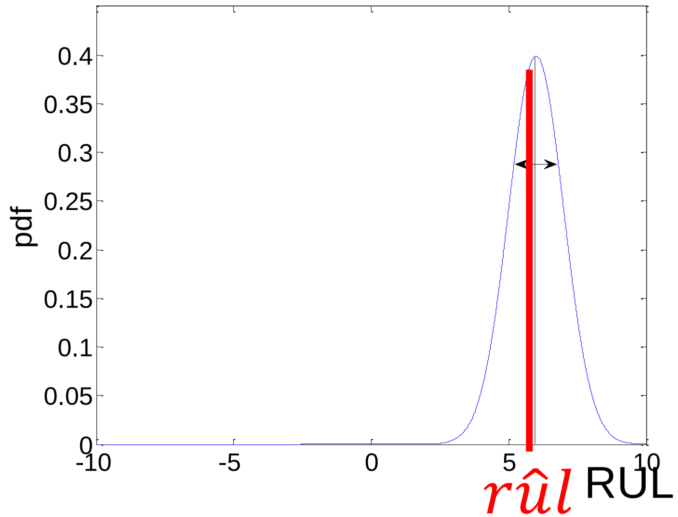
- Prediction error:

$$\varepsilon = rul - r\hat{u}l$$

- Assumptions:
  - $E[\varepsilon] = 0$ (the prognostic model is unbiased)
  - ε has a normal distribution

- Objective: $Var[\varepsilon] = \sigma_\varepsilon^2 = E[(rul - r\hat{u}l\ )^2]$





How? [prognostics]

- Prediction error:

$$\varepsilon = rul - r\hat{u}l$$

- Assumptions:
  - $E[\varepsilon] = 0$ (the prognostic model is unbiased)
  - ε has a normal distribution

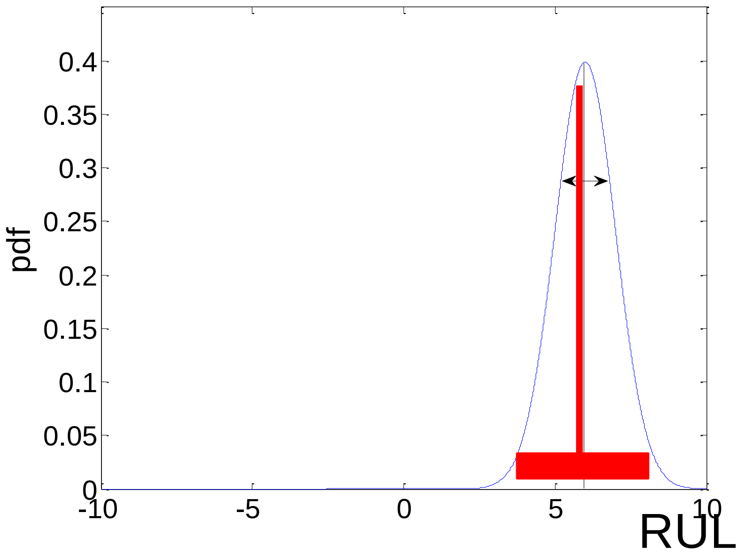
- Objective: $Var[\varepsilon] = \sigma_\varepsilon^2 = E[(rul - r\hat{u}l\,)^2]$

Prediction Interval:

$$P(rul_5 \leq rul \leq rul_{95}) = 90\%$$

with:

$$rul_\alpha = r\hat{u}l - c^\alpha \sigma_\varepsilon$$



How? [prognostics]

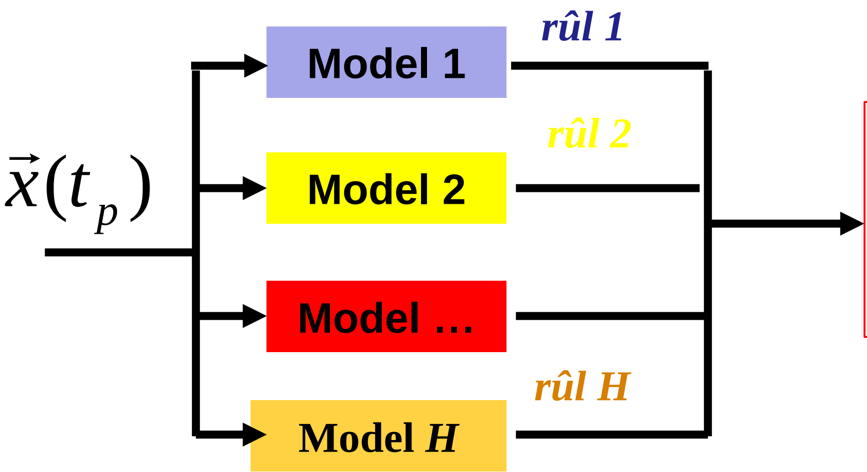
- Sources of uncertainty in prognostics
  1) Measurements → σ_{ME}^2
  2) Degradation Process → σ_{PR}^2
  3) Modeling Error → σ_{MO}^2

- Prediction error decomposition

$$\boxed{Var[\varepsilon] = \sigma_\varepsilon^2 = \sigma_{ME}^2 + \sigma_{PR}^2 + \sigma_{MO}^2}$$

How? [prognostics]

- Model uncertainty estimate: σ^2_{MO}



$$r\hat{u}l = \frac{\sum_{i=1}^{H} r\hat{u}l_i}{H}$$

Ensemble RUL prediction

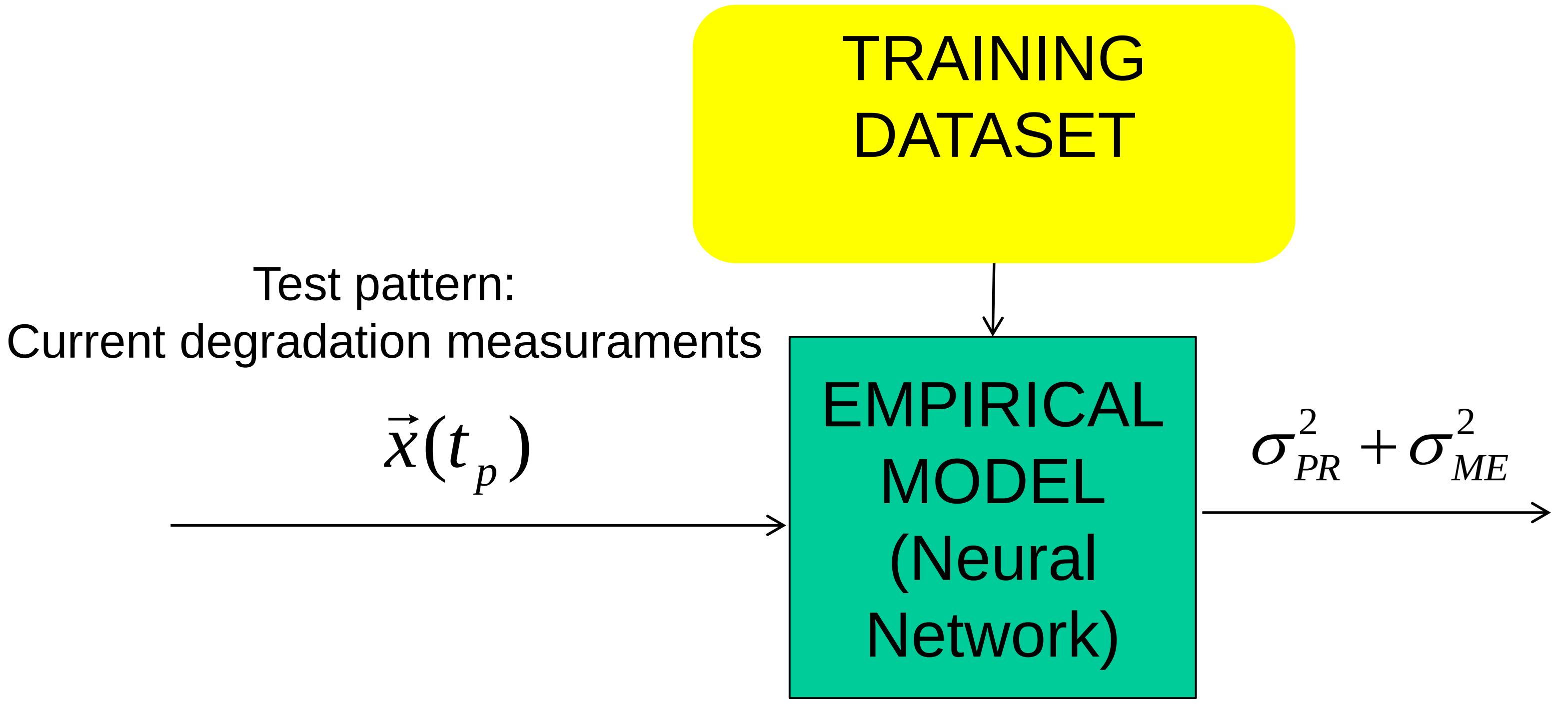
$$\sigma^2_{MO} = \frac{\sum_{i=1}^{H} (r\hat{u}l_i - r\hat{u}l)^2}{H-1}$$

Model uncertainty

(Heskes, 1997) T. Heskes. *Practical confidence and prediction intervals* in Advances Neural Information Processing Systems 9, M. Mozer, M. Jordan, and T. Heskes, Eds. Cambridge, MA: MIT Press, 1997, pp. 466–472.

How? [prognostics]

- Process and measurement uncertainty estimate: $\sigma_{PR}^2 + \sigma_{ME}^2$



How? [prognostics]

- Process and measurement uncertainty estimate: $\sigma_{PR}^2 + \sigma_{ME}^2$

TRAINING DATASET

$$\left(\vec{x}; \underbrace{(rul - r\hat{u}l)^2 - \sigma_{MO}^2}\right)$$

$$E[(rul - r\hat{u}l)^2] - \sigma_{MO}^2 =$$

$$= \sigma_{\varepsilon}^2 - \sigma_{MO}^2 =$$

$$= \sigma_{PR}^2 + \sigma_{ME}^2 + \sigma_{MO}^2 - \sigma_{MO}^2 =$$

Test pattern:
Current degradation measuraments

$$\vec{x}(t_p)$$

EMPIRICAL MODEL (Neural Network)

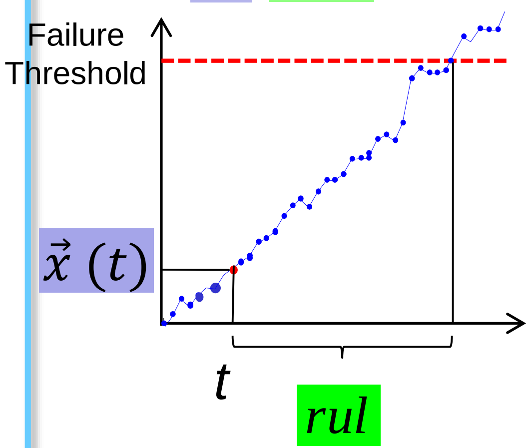
$$\sigma_{PR}^2 + \sigma_{ME}^2$$

How? [prognostics]

- Process and measurement uncertainty estimate: $\sigma_{PR}^2 + \sigma_{ME}^2$

Validation dataset

$(\vec{x}; rul)$



$\rightarrow rul \rightarrow$

TRAINING DATASET

$\left(\vec{x}; (rul - r\hat{u}l)^2 - \sigma_{MO}^2\right)$

Test pattern:
Current degradation measuraments

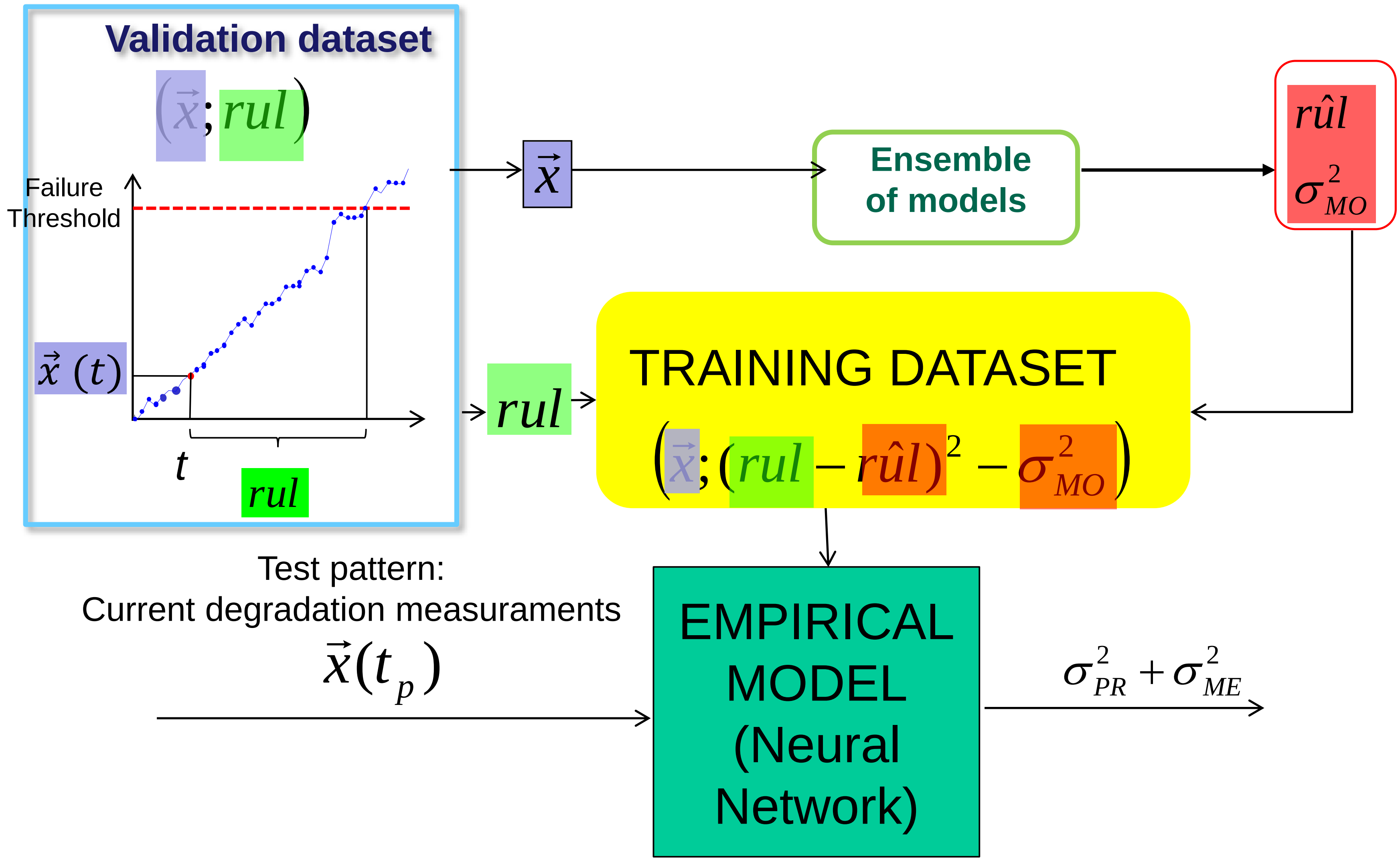
$\vec{x}(t_p)$

EMPIRICAL MODEL (Neural Network)

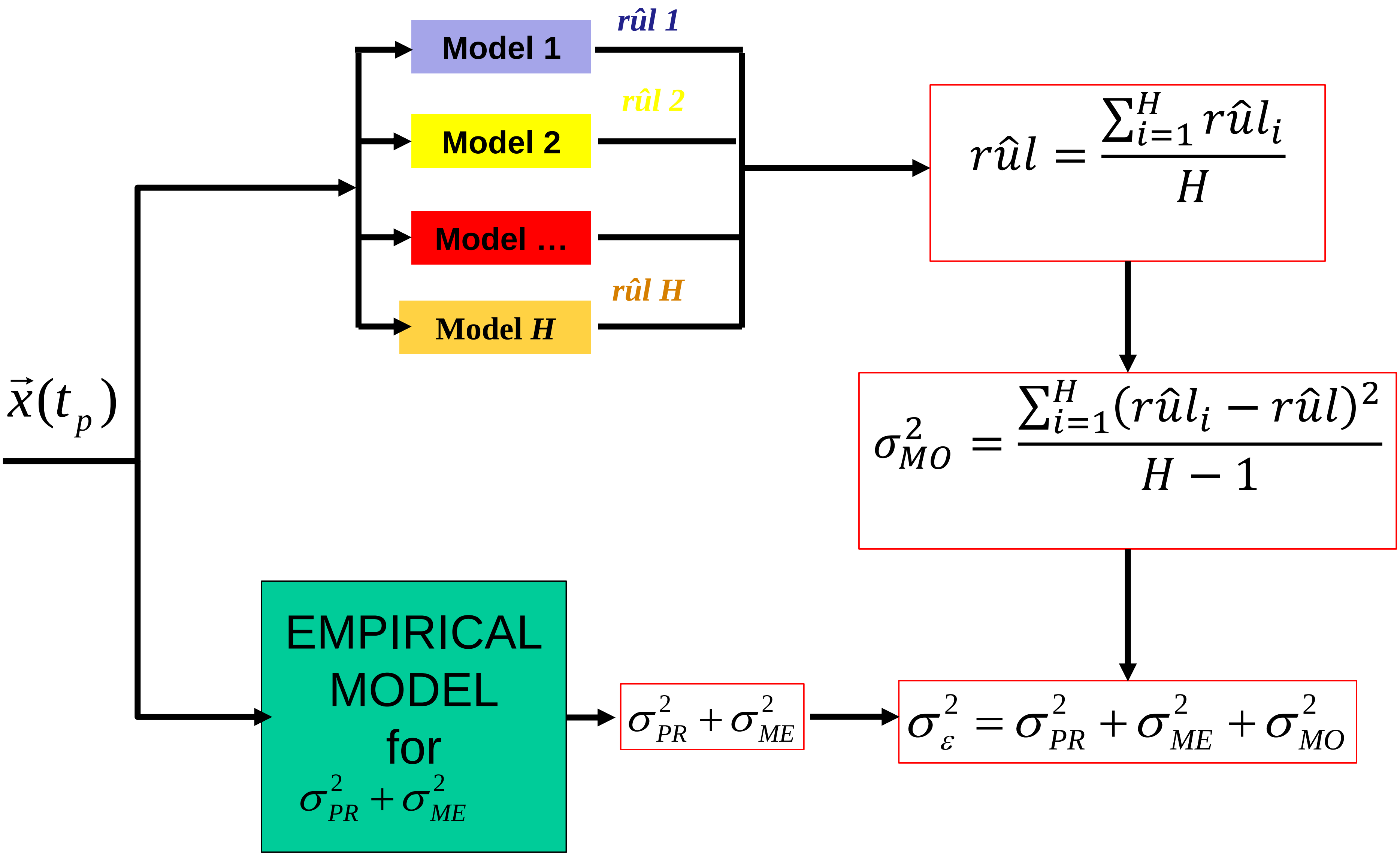
$\sigma_{PR}^2 + \sigma_{ME}^2$

How? [prognostics]

➢ Process and measurement uncertainty estimate: $\sigma^2_{PR} + \sigma^2_{ME}$



How? [prognostics]

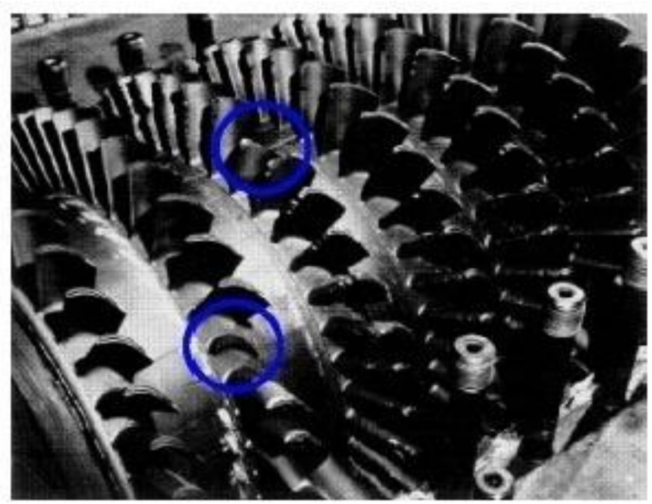


Confidence estimation:

- What is it?
- Why ensembles?
- How?
- Application: prognostics of turbine blades

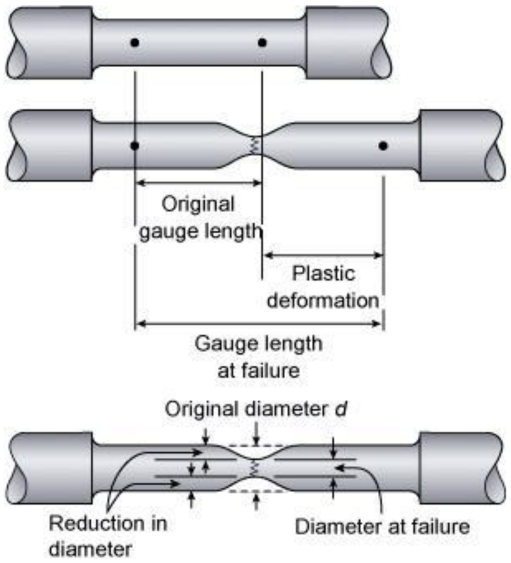
Application: prognostics of turbine blades

➢ Turbine Blade Creep Propagation



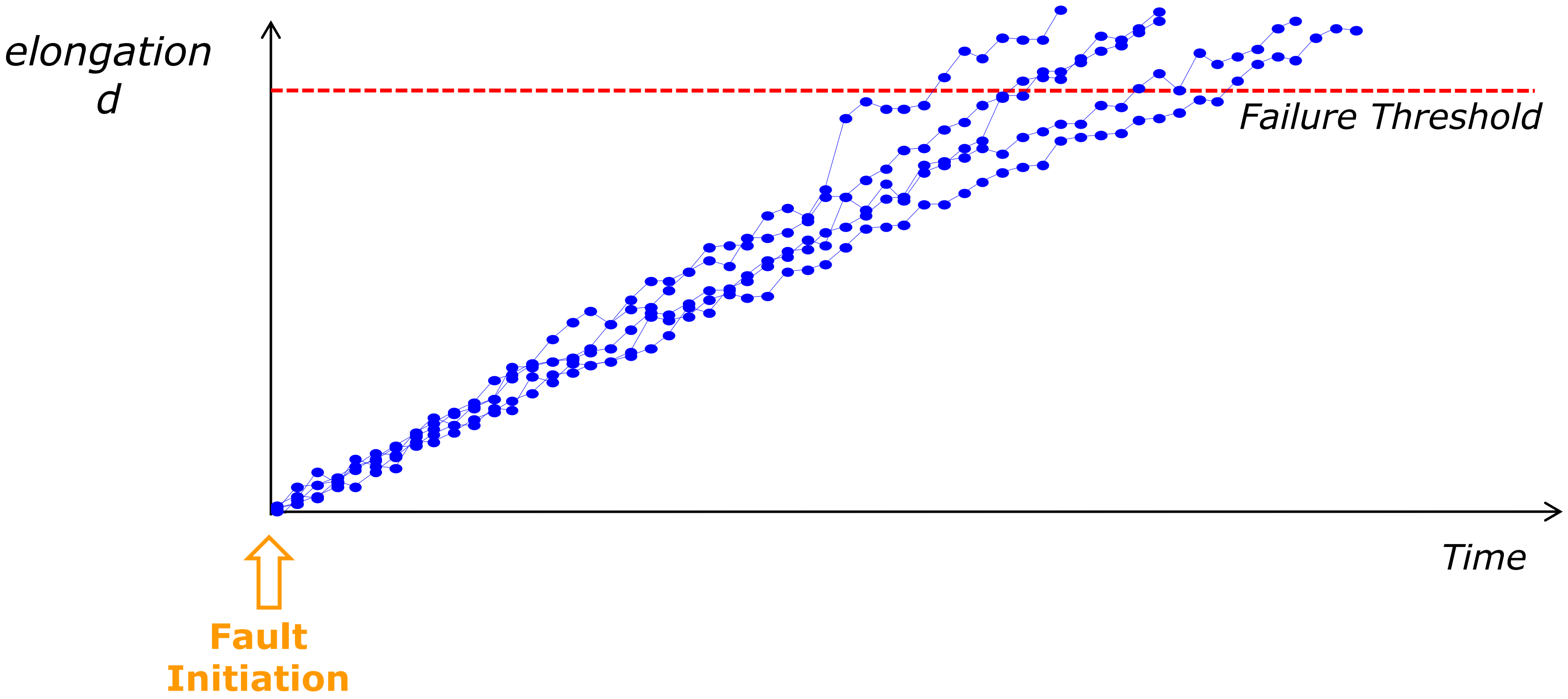
Component: turbine blade
Degradation mechanism: creep
Degradation indicator: blade elongation

$$x(t) = \frac{\text{Length}(t) - \text{initial length}}{\text{initial length}}$$



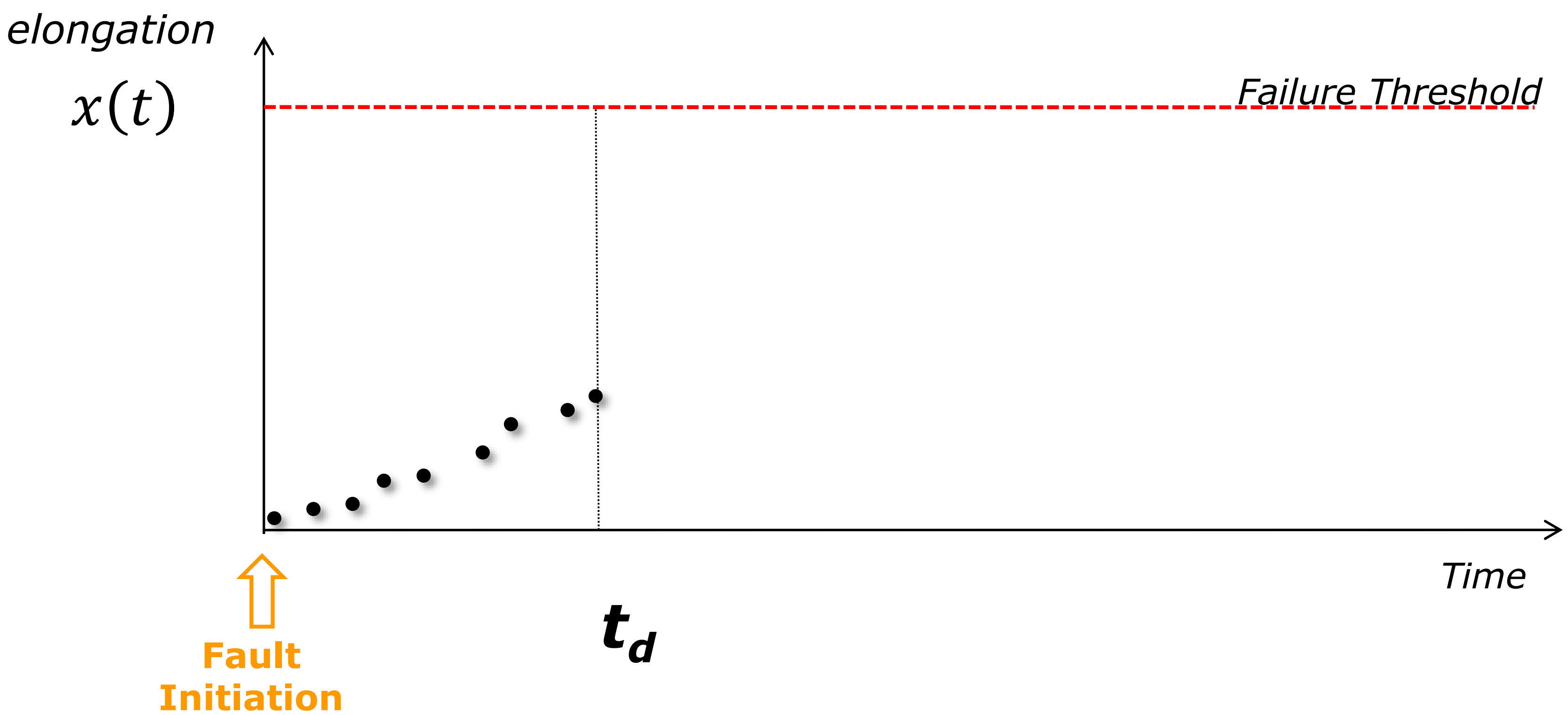
Application: prognostics of turbine blades

- Available Information
  - Failure threshold (1.5%)
  - Historical sequences of elongation measurements in turbine blades (13 run-to-failure trajectories)



Application: prognostics of turbine blades

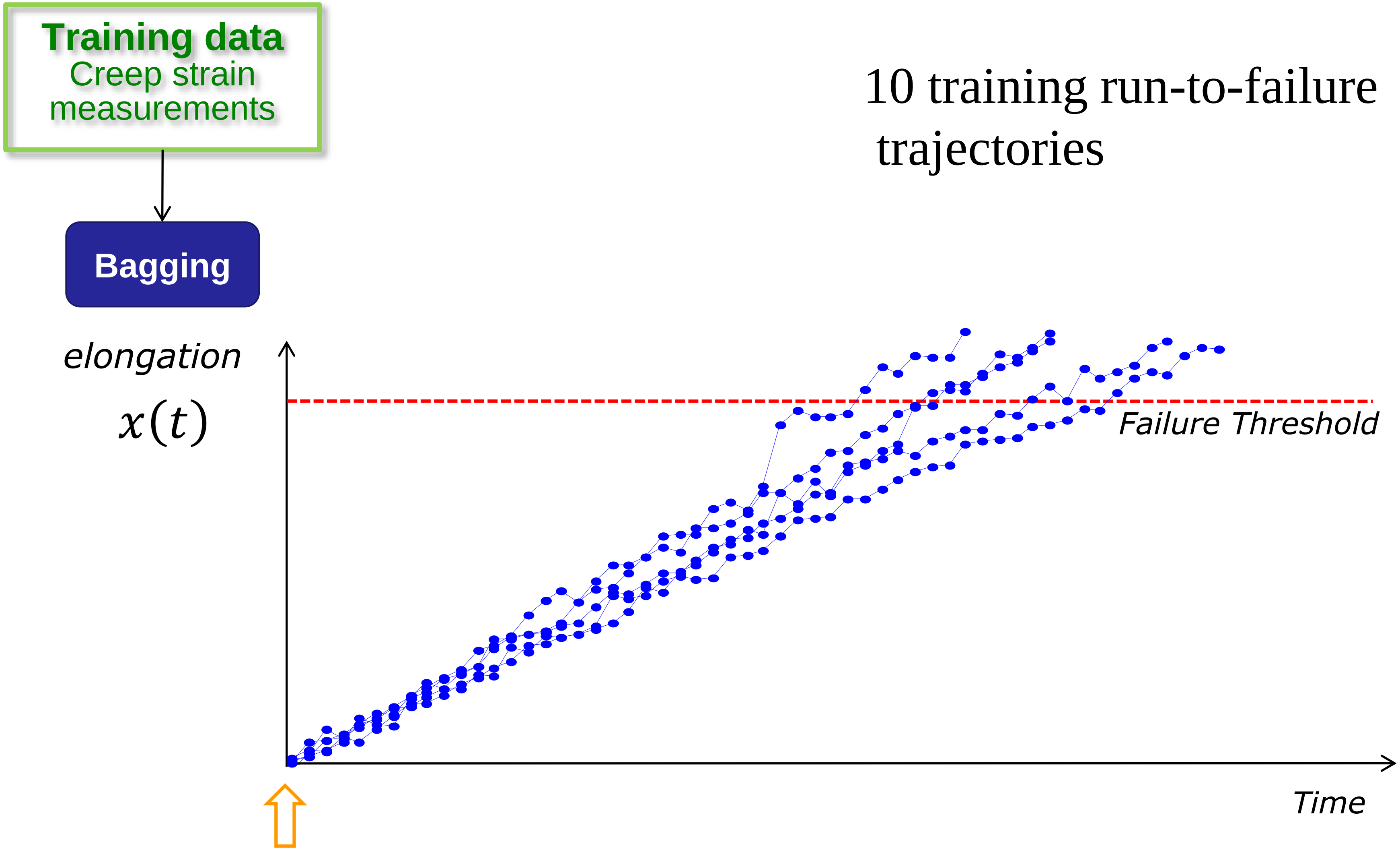
- Available Information
  - Failure threshold
  - Historical sequences of elongation measurements in turbine blades
  - Sequence of elongation measurements of the turbine blade under observation



Application: prognostics of turbine blades

- Ensemble of Bootstrapped Models

10 training run-to-failure trajectories



Application: prognostics of turbine blades

- Ensemble of Bootstrapped Models

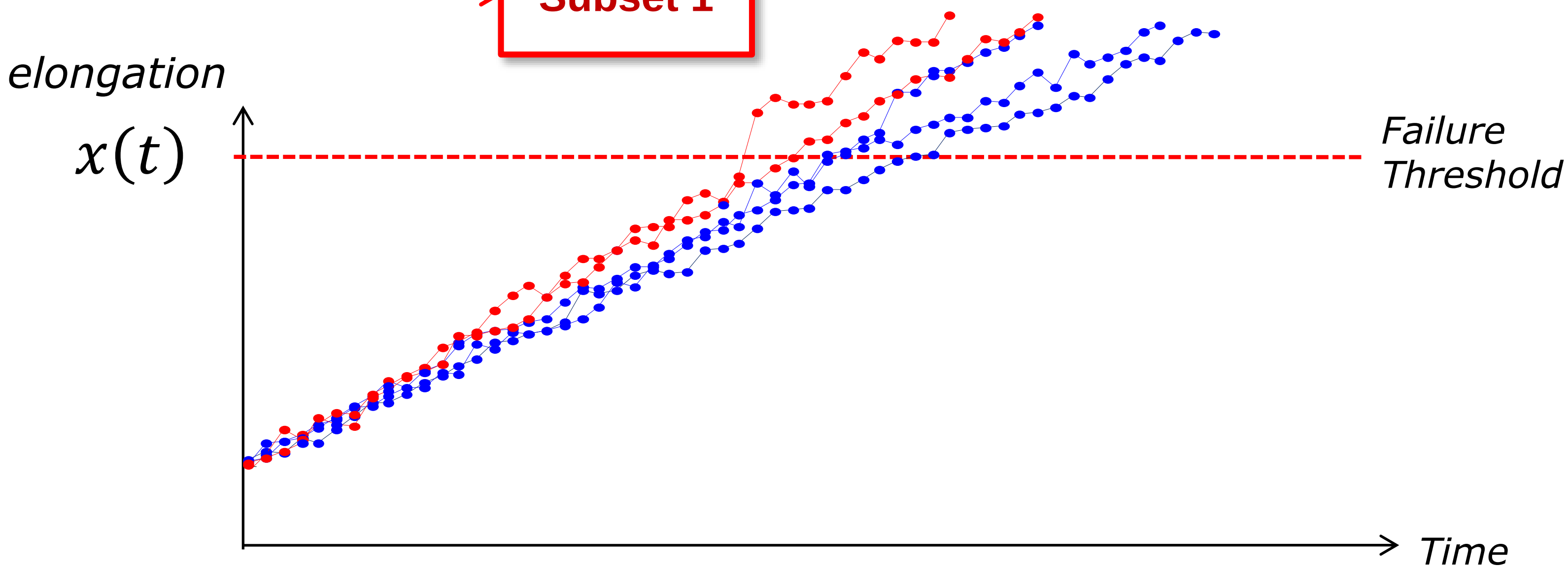
Training data
Creep strain measurements

↓

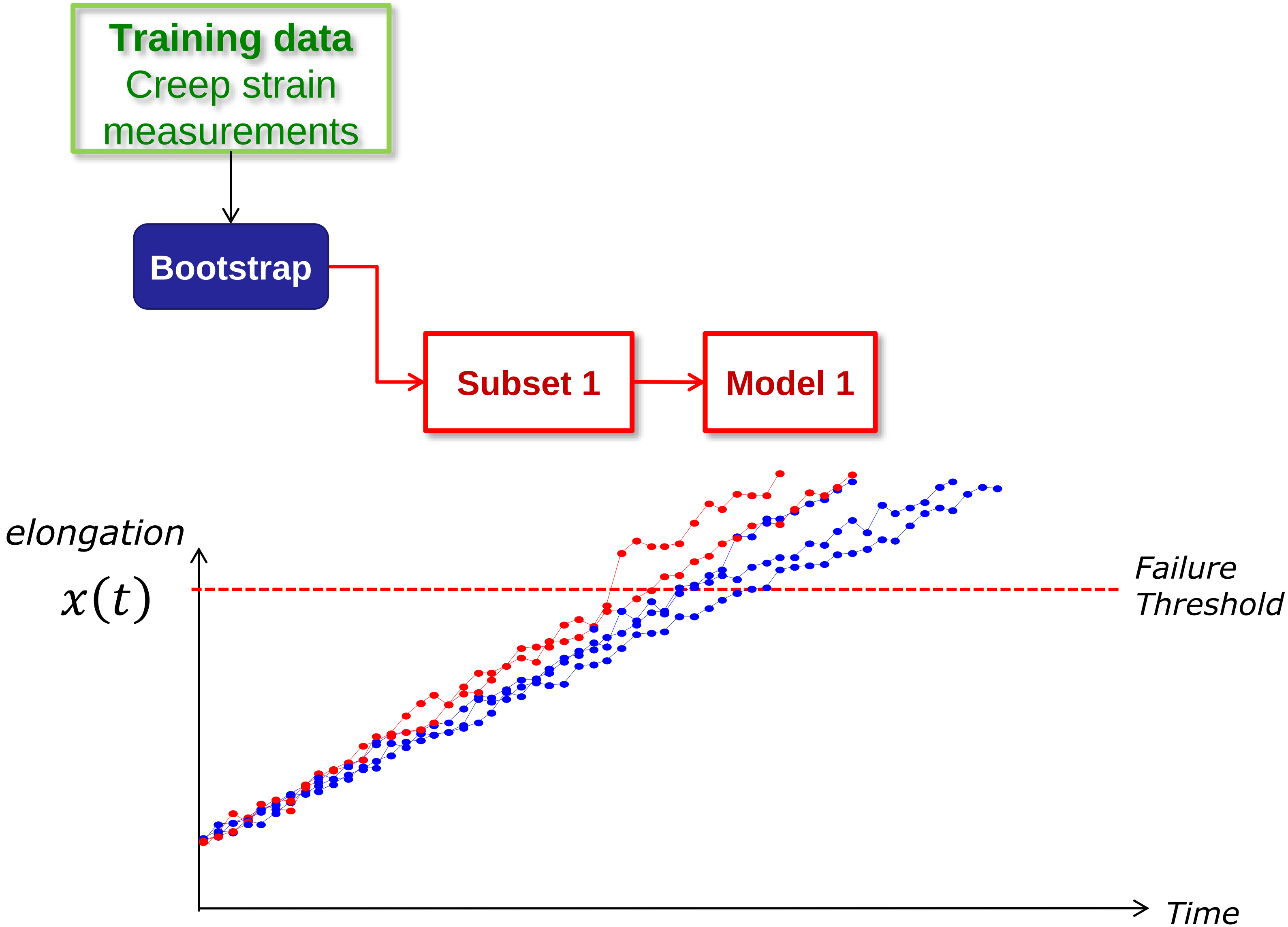
Bagging

→

Subset 1

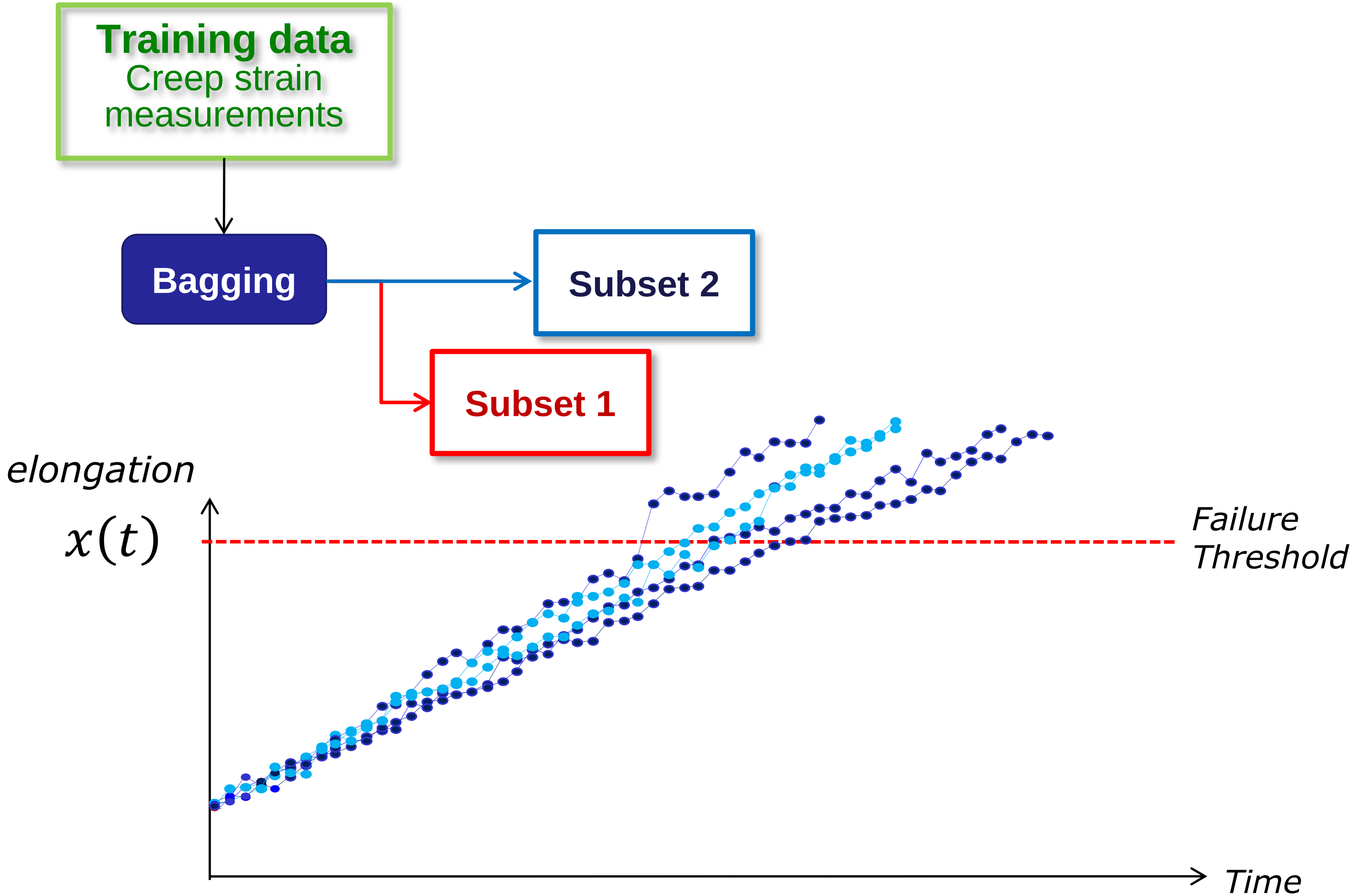


Application: prognostics of turbine blades

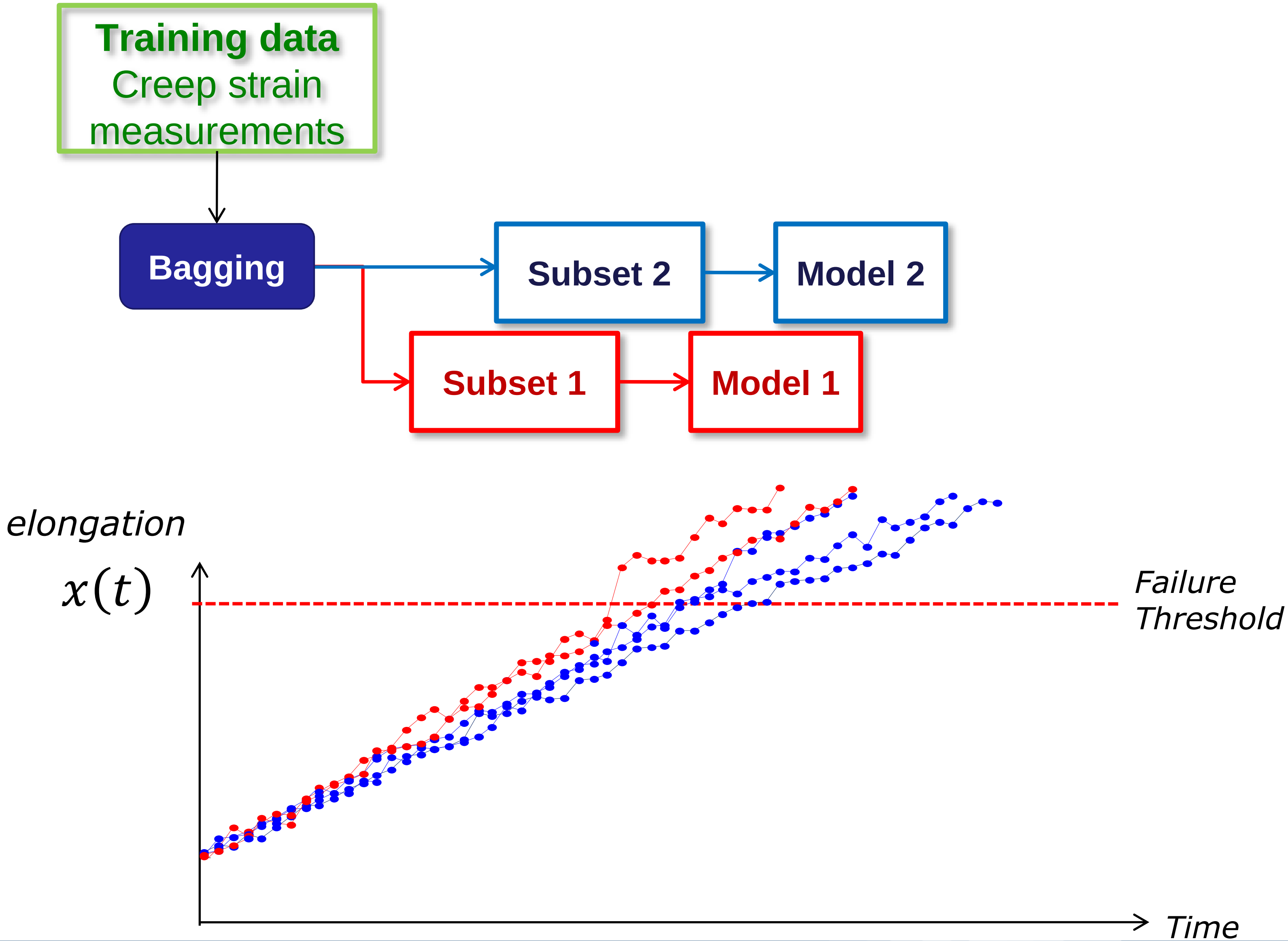


Application: prognostics of turbine blades

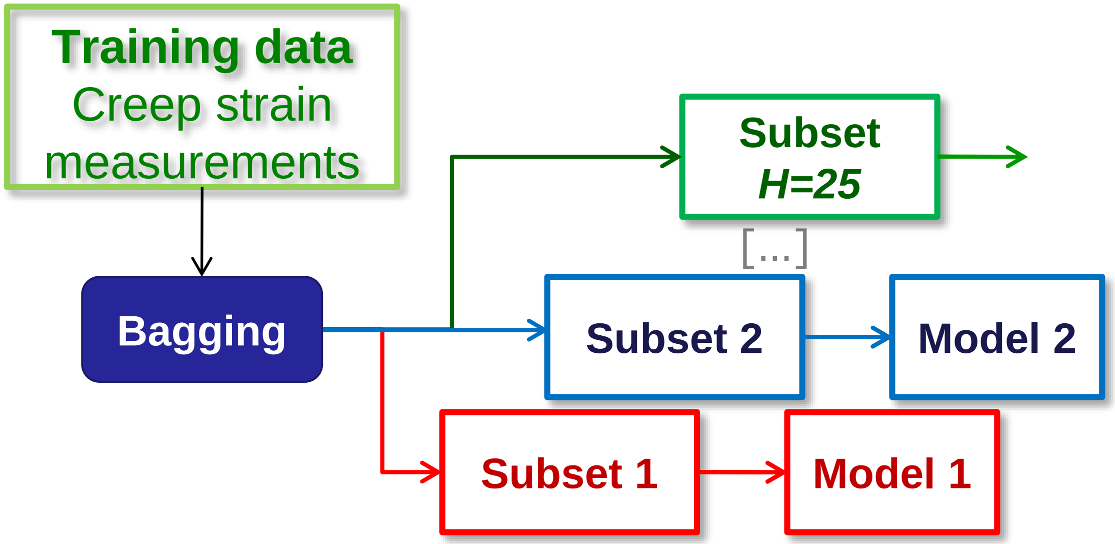
- Ensemble of Bootstrapped Models

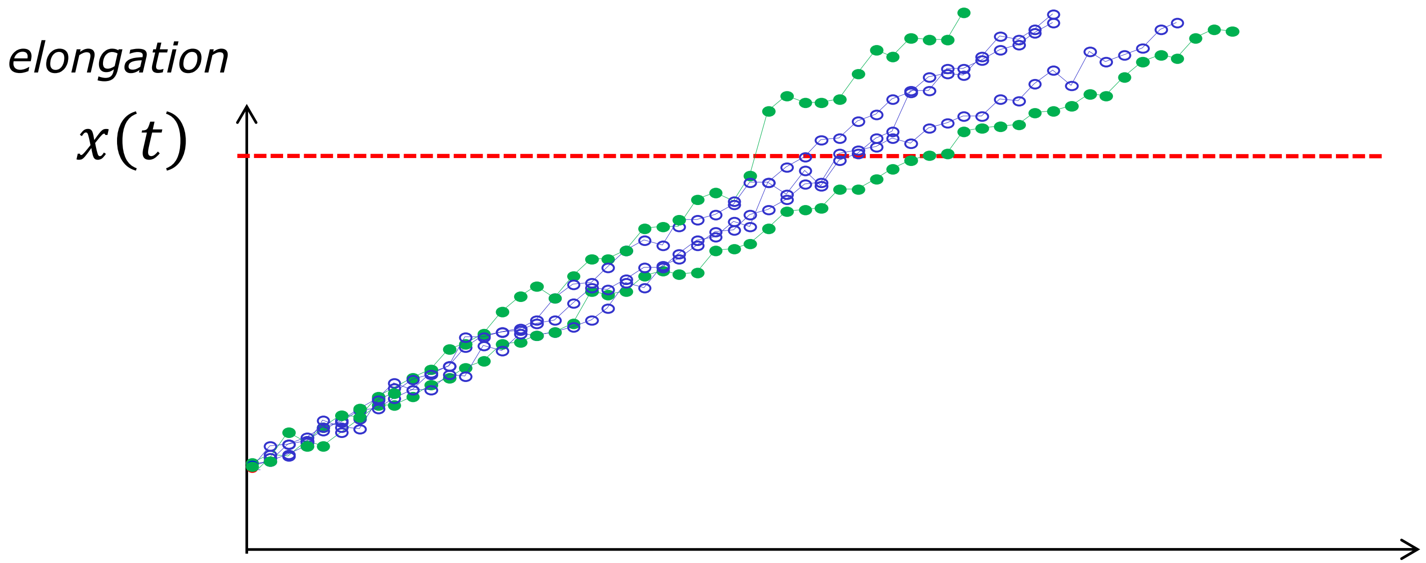


Application: prognostics of turbine blades

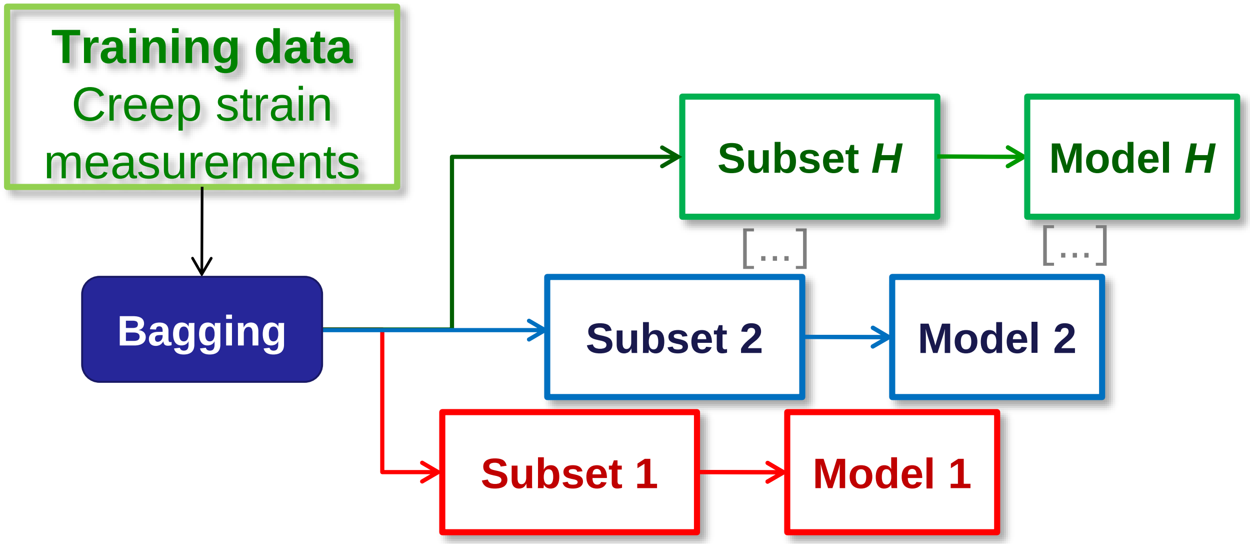


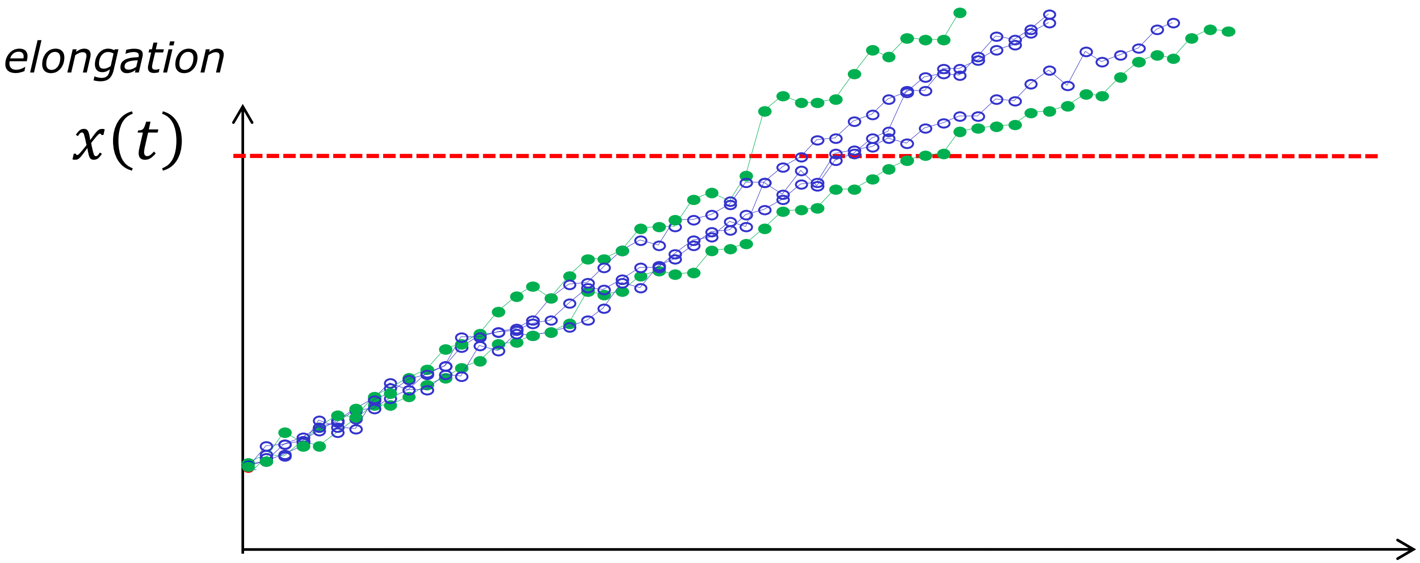
Application: prognostics of turbine blades





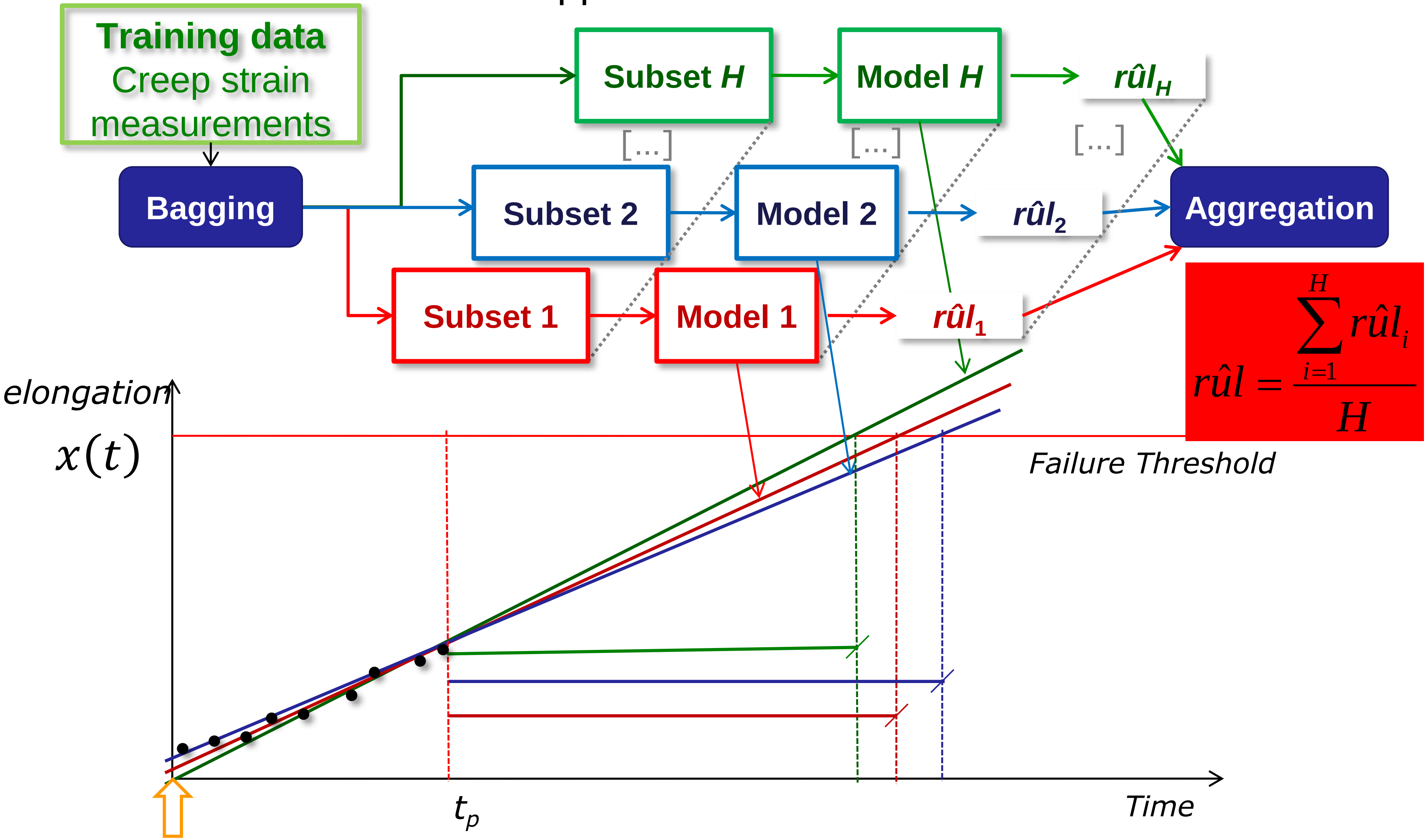
Application: prognostics of turbine blades





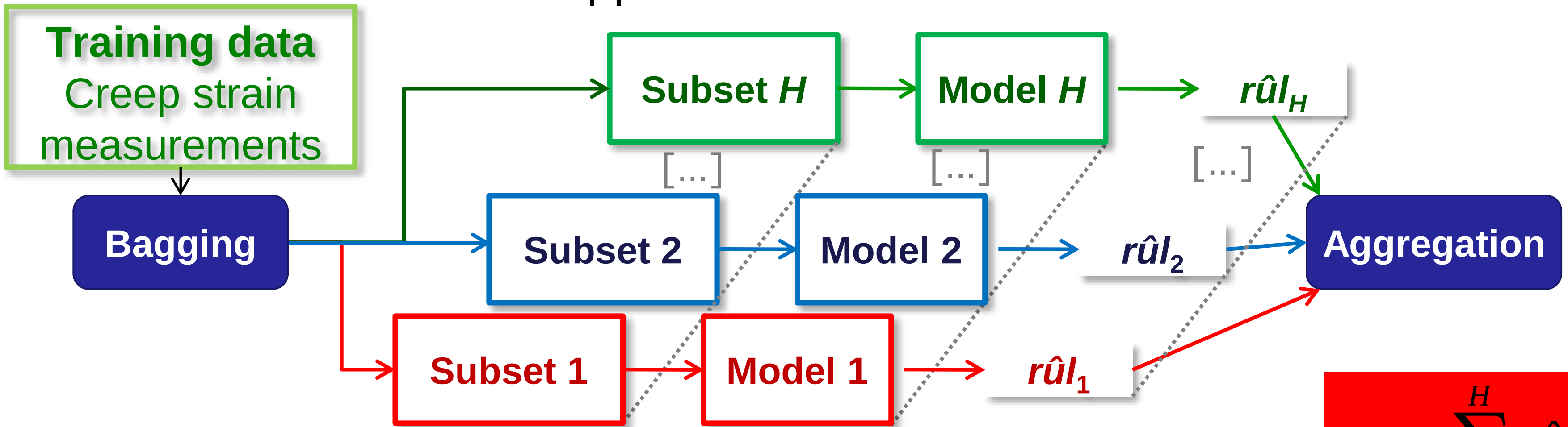
Application: prognostics of turbine blades

- Ensemble of Bootstrapped Models



Application: prognostics of turbine blades

➢ Ensemble of Bootstrapped Models



$$r\hat{u}l = \frac{\sum_{i=1}^{H} r\hat{u}l_i}{H}$$

$$\sigma_{MO}^2 = \frac{\sum_{i=1}^{H}(r\hat{u}l_i - r\hat{u}l)^2}{H-1}$$

Application: prognostics of turbine blades

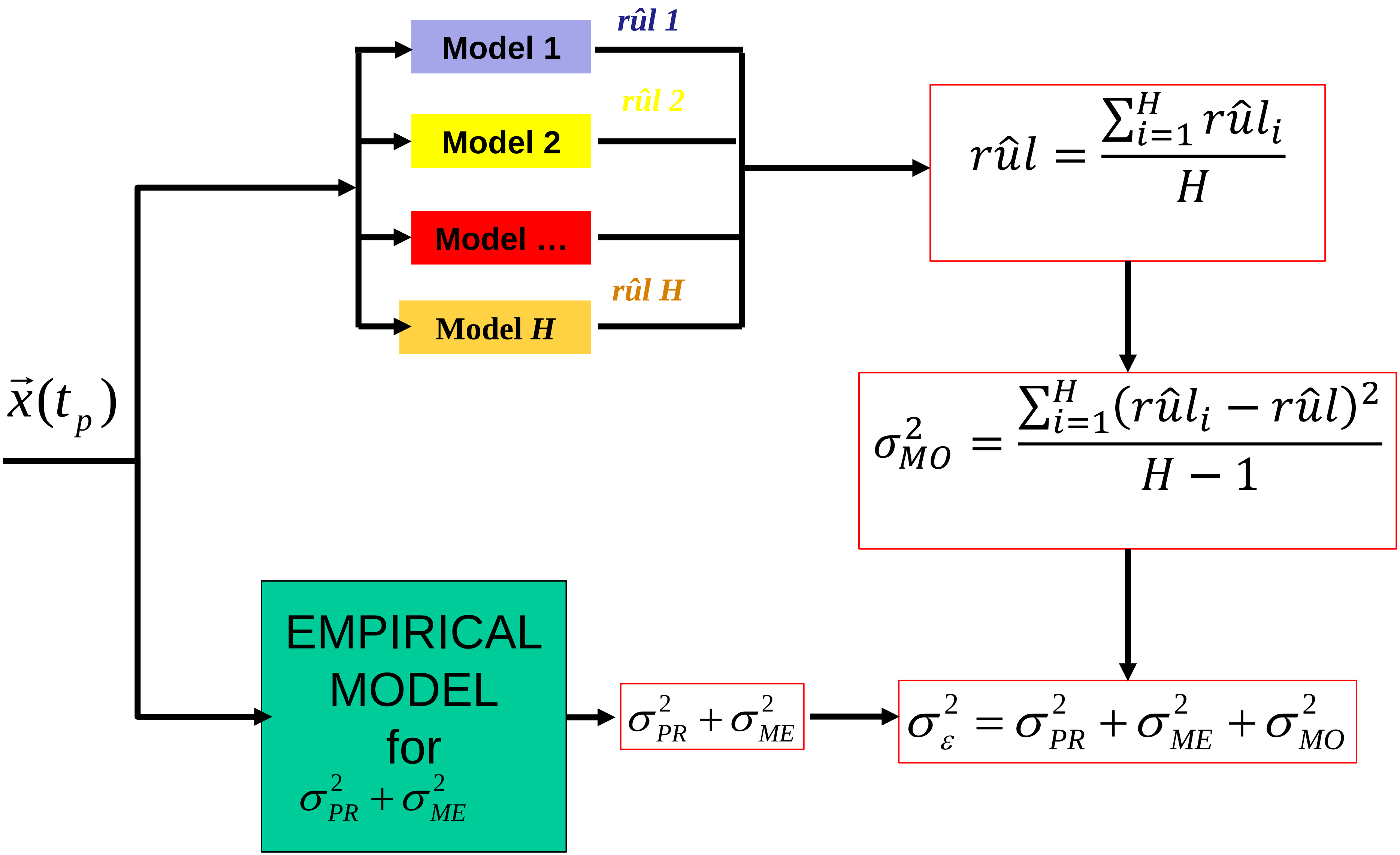
$\vec{x}(t_p)$

3 validation
run-to-failure trajectories

EMPIRICAL
MODEL
for
$\sigma_{PR}^2 + \sigma_{ME}^2$

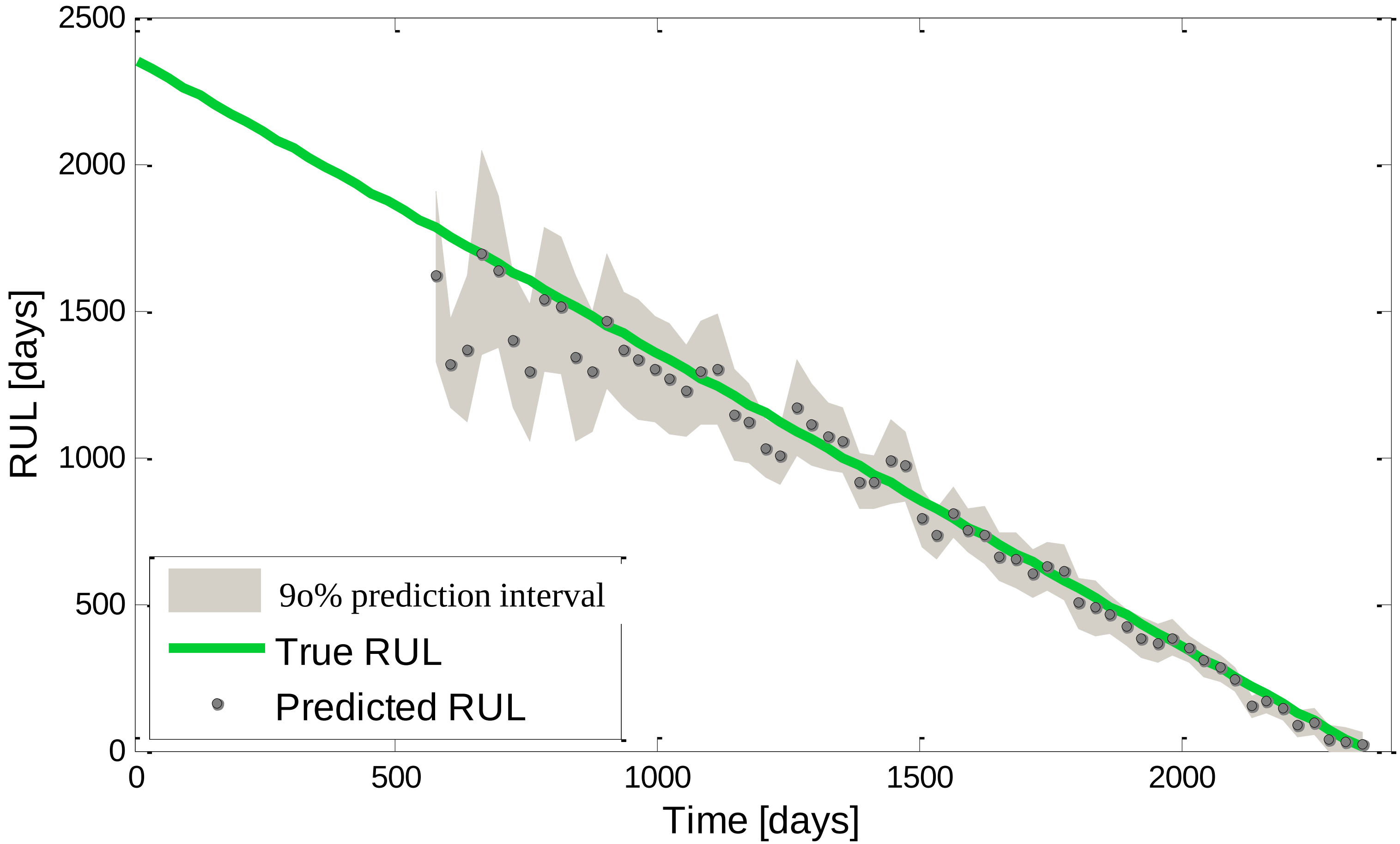
$\sigma_{PR}^2 + \sigma_{ME}^2$

Application: prognostics of turbine blades



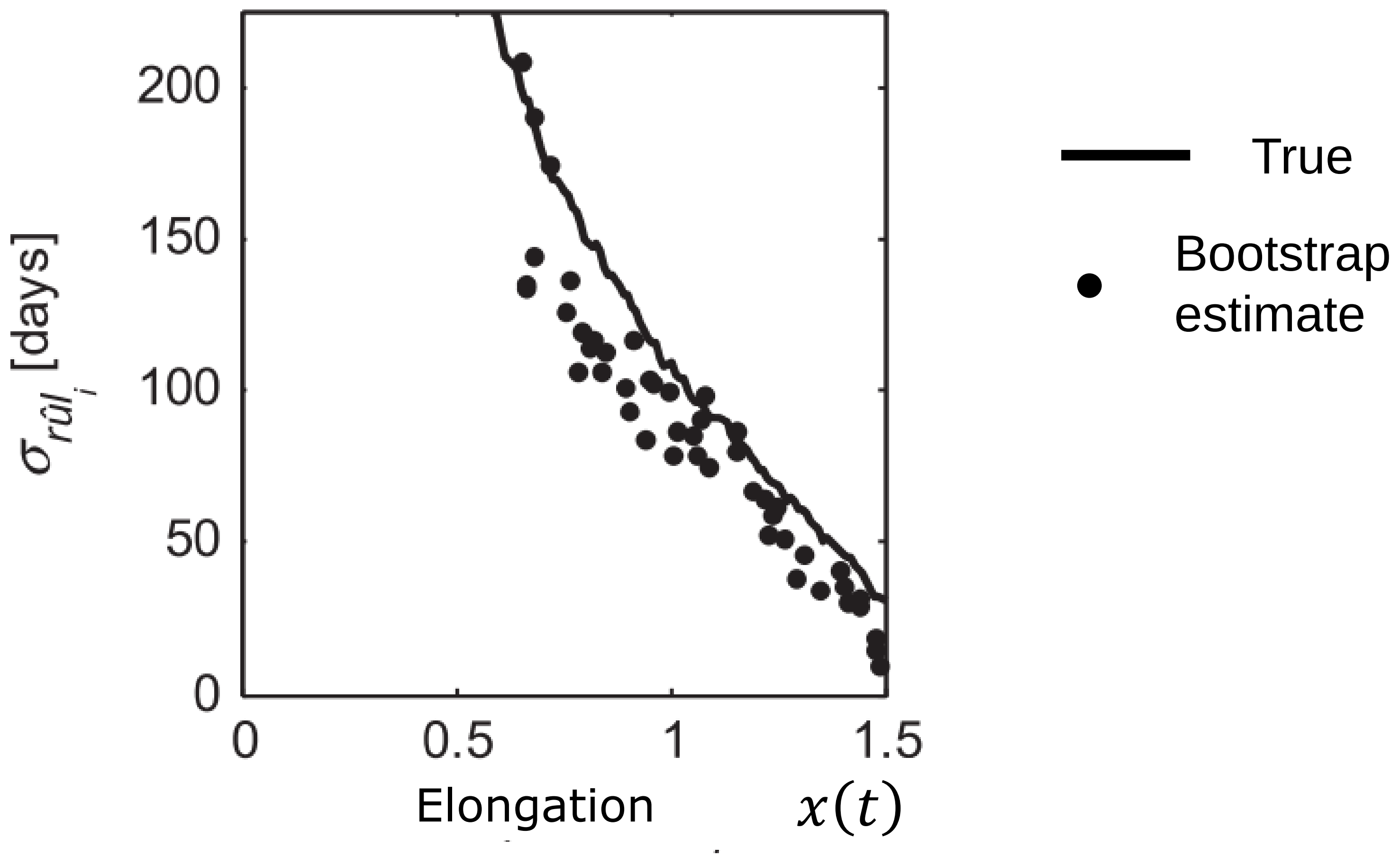
Application: prognostics of turbine blades

➢ RUL prediction with cprediction bounds (90% confidence interval)



Application: prognostics of turbine blades

- Uncertainty estimate



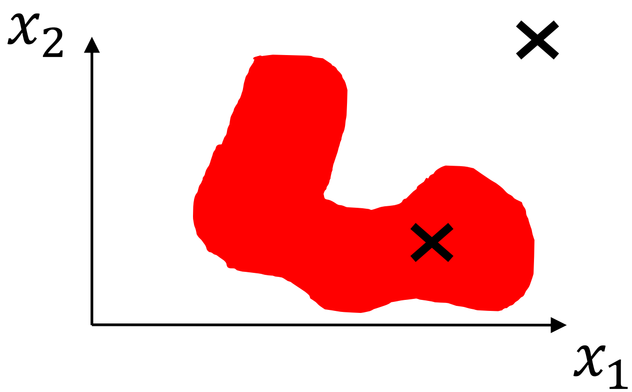
Tutorial Contents

- Ensembles of Models for Prognostics and Health Management (PHM):
  - Accuracy
  - Confidence Estimation
  - Incremental Learning

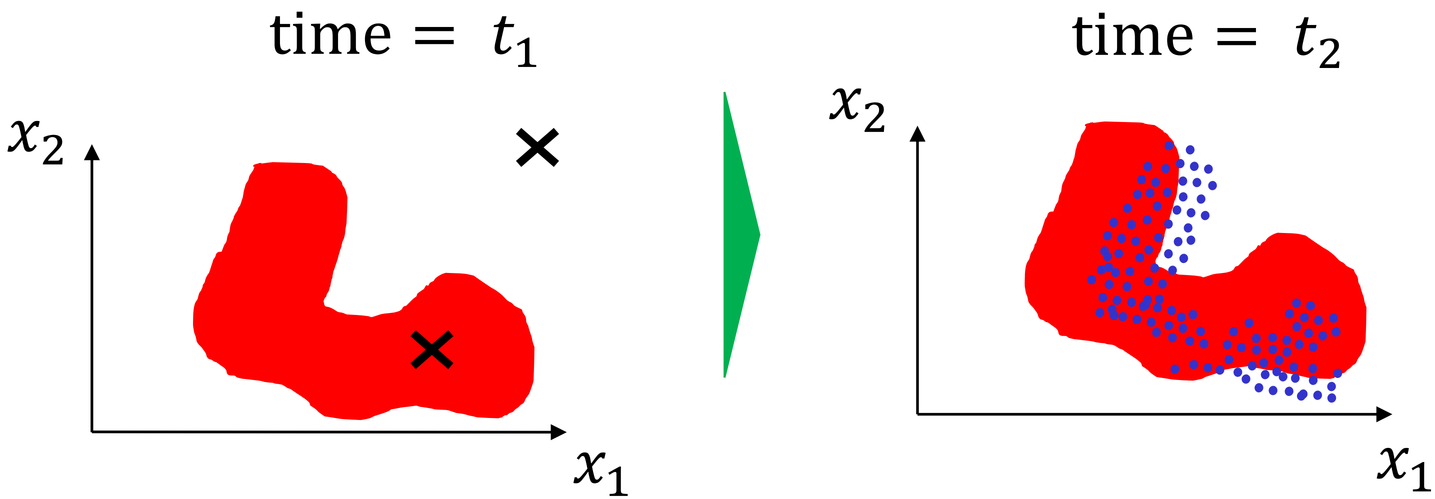
- **Incremental learning:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
  - **Application: fault diagnostics in a nuclear power plant system**

Incremental learning: What is it? [fault detection]

$\text{time} = t_1$



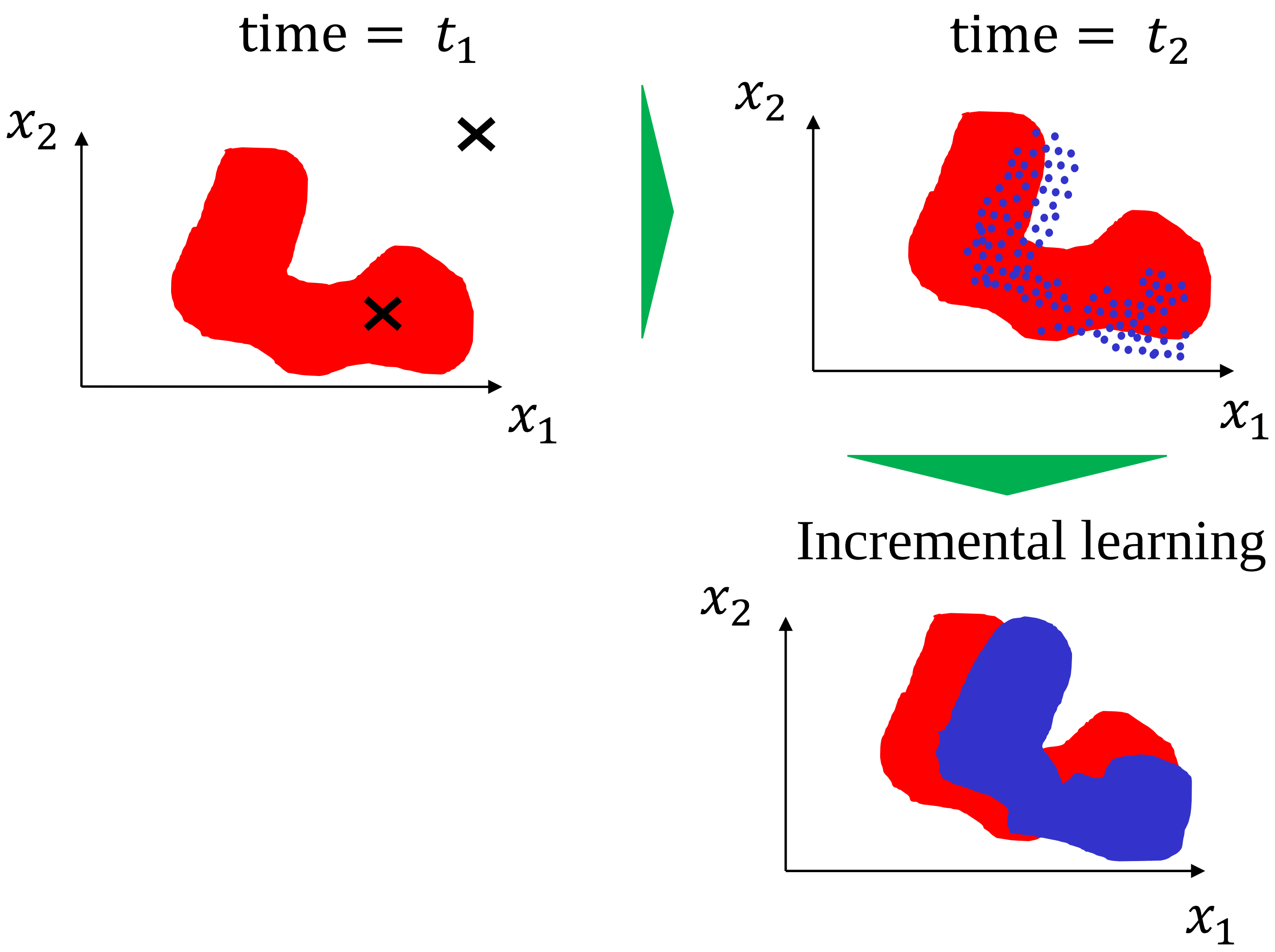
Incremental learning: What is it? [fault detection]



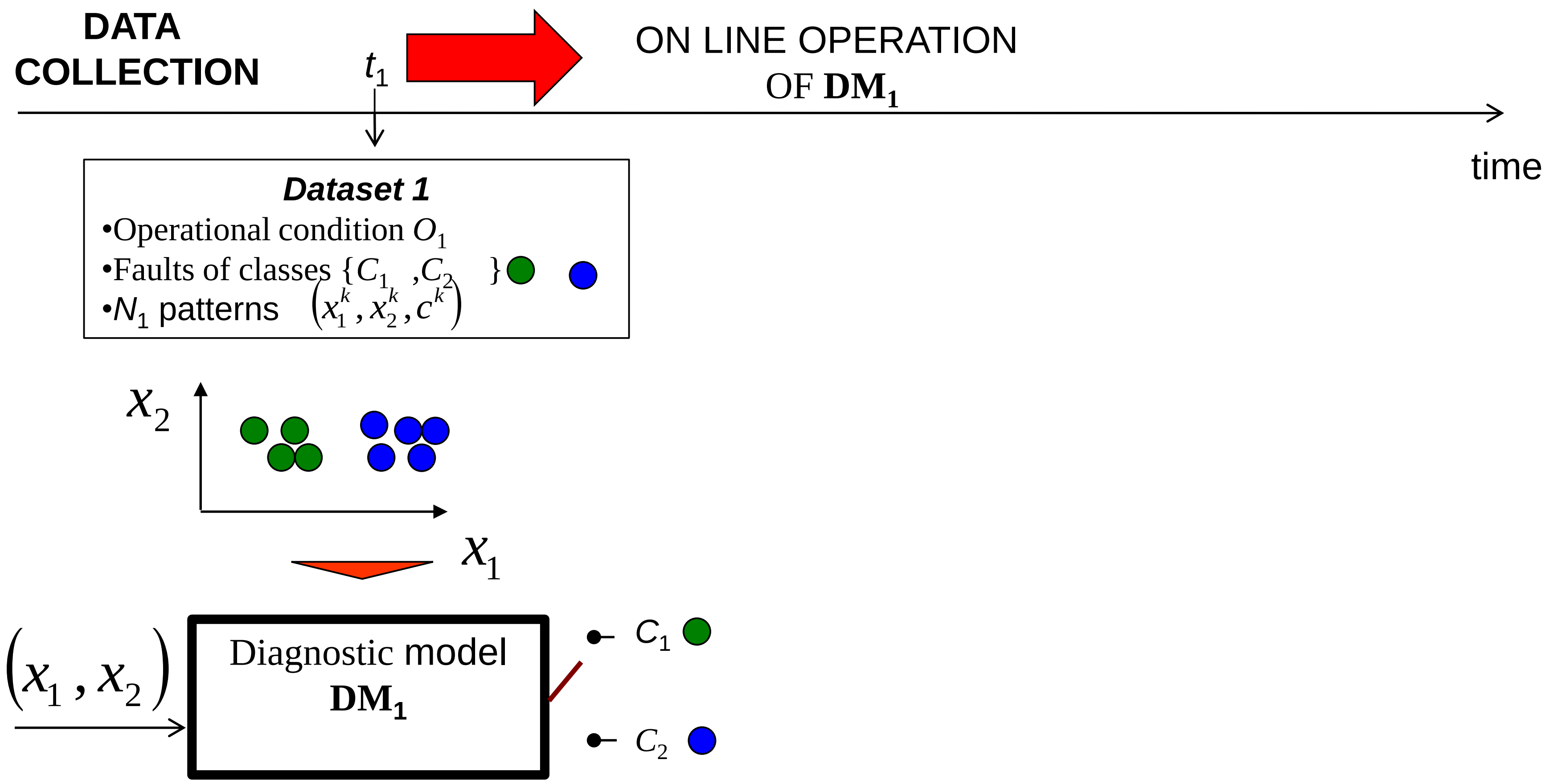
Evolving environment…
Context Change…

x_1

Incremental learning: What is it? [fault detection]



Incremental learning: What is it? [fault diagnostics]



Incremental learning: What is it? [fault diagnostics]

Diagnostics

Performance of DM_1: OK

- Operational condition: O_1
- Faults of classes $\{C_1, C_2\}$

DATA COLLECTION

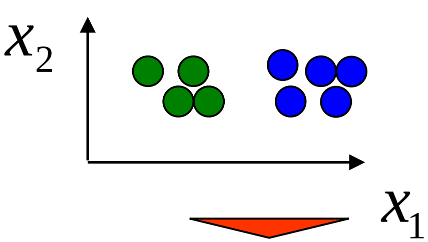
t_1

ON LINE OPERATION OF $\mathbf{DM_1}$

time

Dataset 1

- Operational condition O_1
- Faults of classes $\{C_1, C_2\}$
- N_1 patterns (x_1^k, x_2^k, c^k)



(x_1, x_2) → Diagnostic model $\mathbf{DM_1}$ → C_1 / C_2

Incremental learning: What is it? [fault diagnostics]

Performance of DM_1: OK

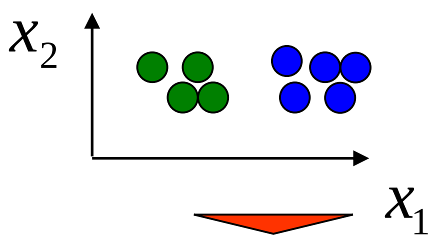
- Operational condition: O_1
- Faults of classes $\{C_1, C_2\}$

DATA COLLECTION

t_1 t_2 time

Dataset 1

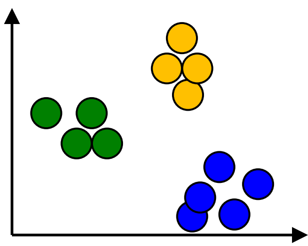
- Operational condition O_1
- Faults of classes $\{C_1, C_2\}$
- N_1 patterns $\left(x_1^k, x_2^k, c^k\right)$



$\left(x_1, x_2\right)$ → Diagnostic model **DM_1** → C_1, C_2

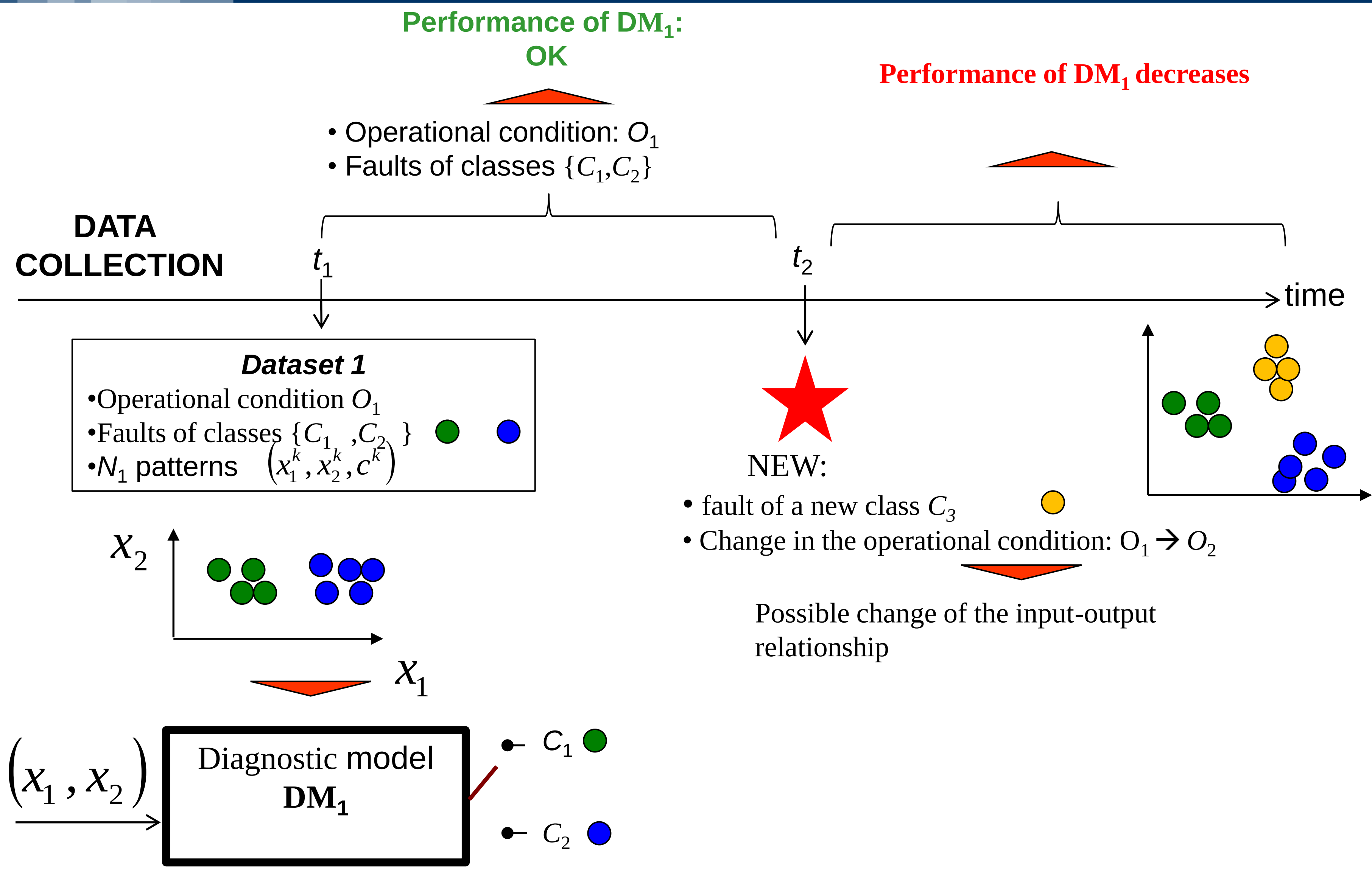
NEW:

- fault of a new class C_3
- Change in the operational condition: $O_1 \rightarrow O_2$

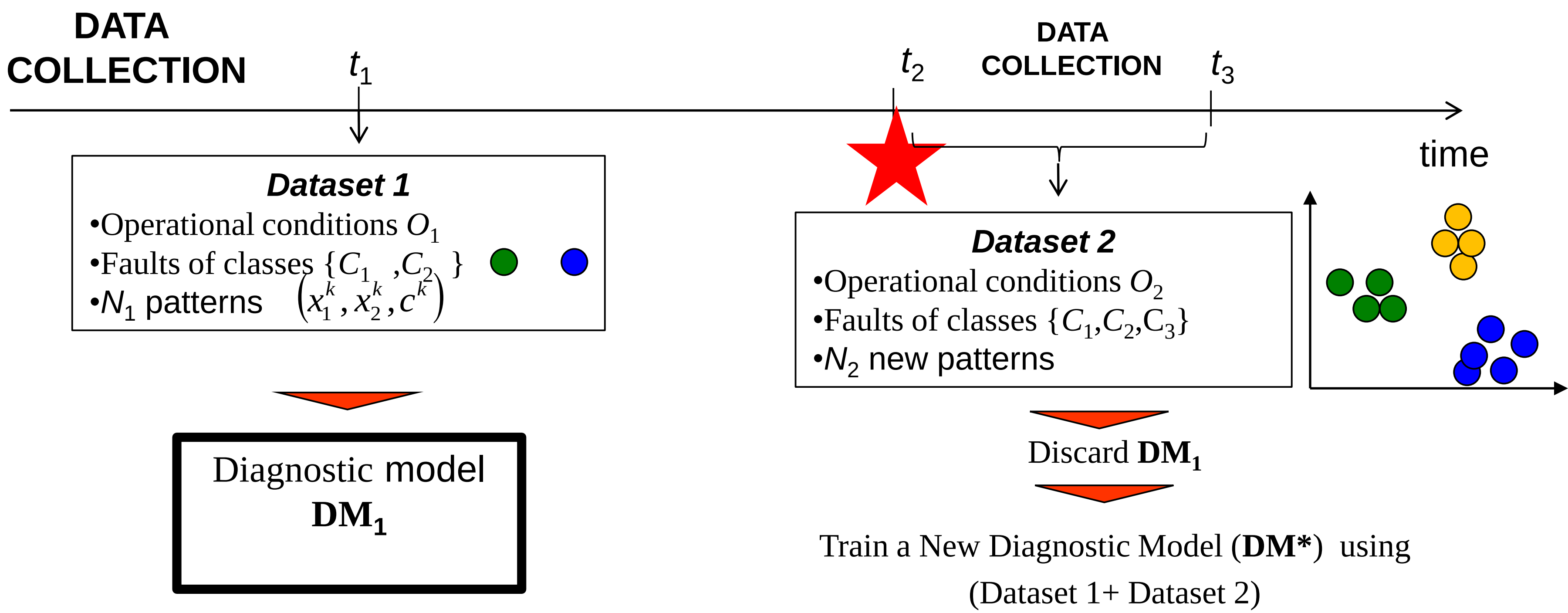


Possible change of the input-output relationship

Incremental learning: What is it? [fault diagnostics]



Incremental learning: What is it? [fault diagnostics]



catastrophic forgetting

- Loss of all previously acquired knowledge
- Computational and financial efforts necessary for each model retraining

Incremental learning: What is it? [fault diagnostics]

DATA COLLECTION t_1 t_2 **DATA COLLECTION** time

ON LINE OPERATION OF DM$_2$

Performance of DM$_2$: OK

Dataset 1
- Operational condition O_1
- Faults of classes $\{C_1, C_2\}$
- N_1 patterns $\left(x_1^k, x_2^k, c^k\right)$

Diagnostic model **DM$_1$**

Dataset 2
- Operational condition O_2
- Faults of classes $\{C_1, C_2, C_3\}$
- N_2 new patterns

Incremental learning
- Learn the novel information content of Dataset 2 without forgetting the previously acquired knowledge by **DM$_1$**
- Update the model without requiring access to the previously seen datasets

DM$_2$

- **Incremental learning:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
  - **Application: fault diagnostics in a nuclear power plant system**

Why ensembles? [Diagnostics]

➢ Ensemble of ensembles for novelty identifiability

t_1

ON LINE OPERATION OF **DM_1**

time

NEW:

• fault of a new class C_3,

Dataset 1

•N_1 patterns

•Faults of classes C_1,C_2

Performance DM1 decreases

Build **DM_1**:
Ensemble E
made of H_1 classifiers

Collect Dataset 2

•N_2 patterns

• Faults of classes C_1, C_2 ,C_3

Update DM_1:
add to ensemble *E*
H_2 new classifiers trained with **Dataset 2**

DM_2:
Ensemble
made of H_1 + H_2 classifiers

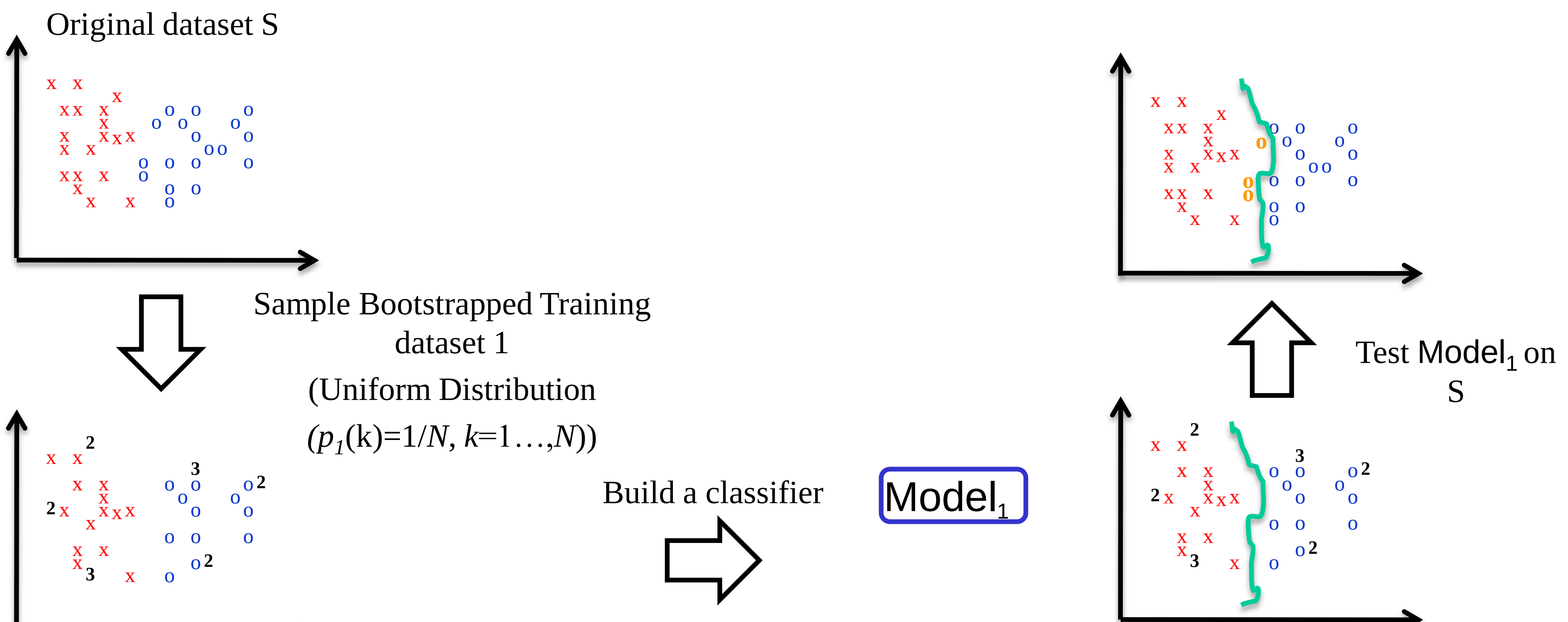
- **Incremental learning:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
  - **Application: fault diagnostics in a nuclear power plant system**

How?

1. AdaBoost
2. Ensemble of Ensembles
3. Consult and vote procedure

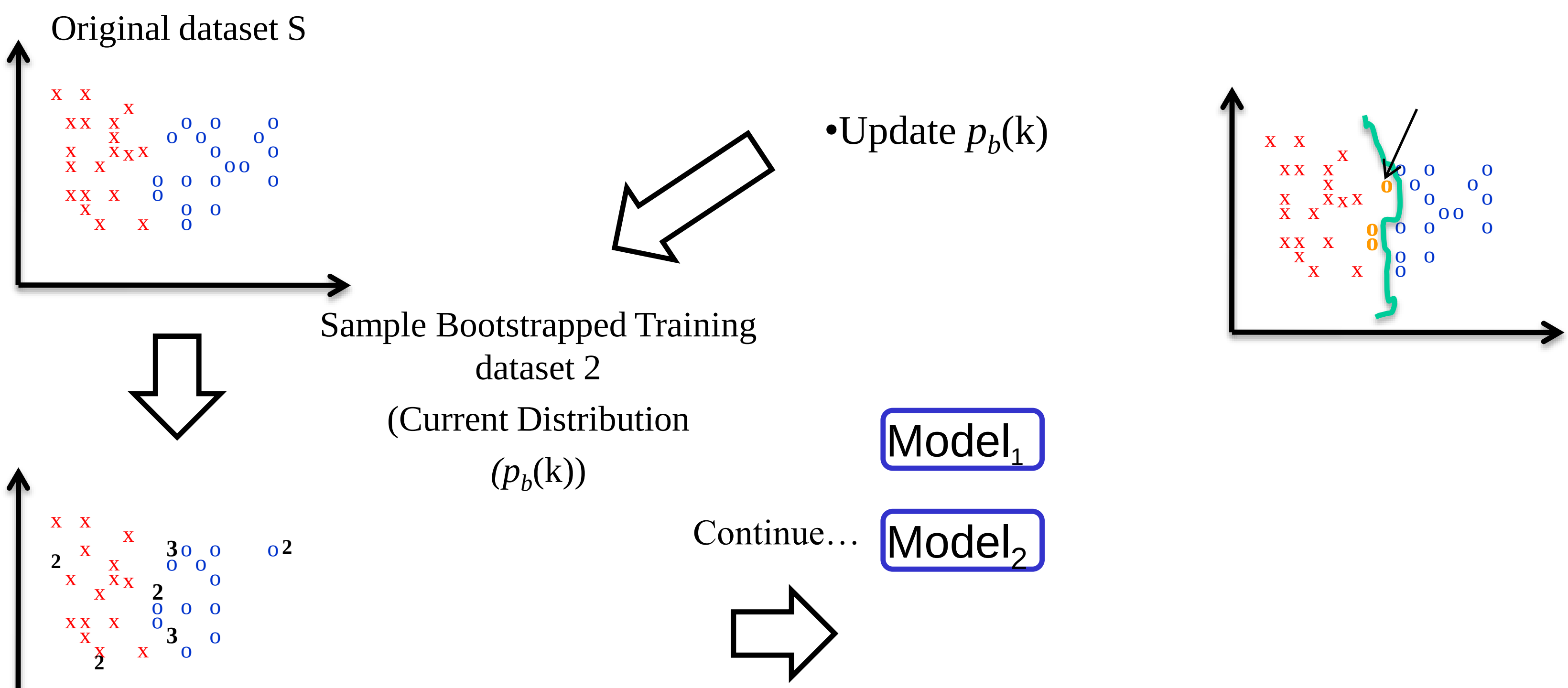
How?

1. Adaboost:
   - Sequential production of classifiers
   - Each classifier is dependent from the previous one
   - Patterns that are incorrectly classified by previous classifiers are chosen more often for the next classifier training



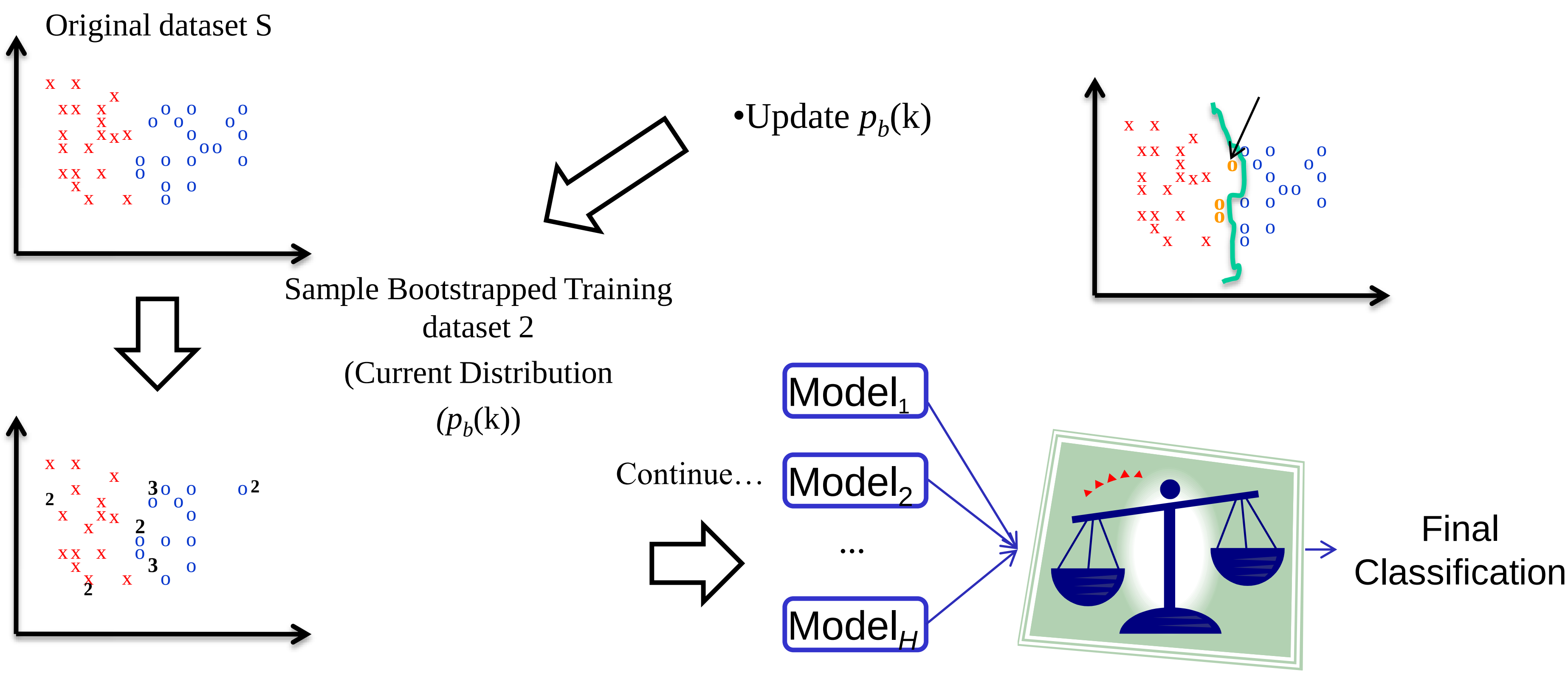
How?

1. Adaboost:
   - Sequential production of classifiers
   - Each classifier is dependent from the previous one
   - Examples that are incorrectly classified by previous classifiers are chosen more often for the next classifier training

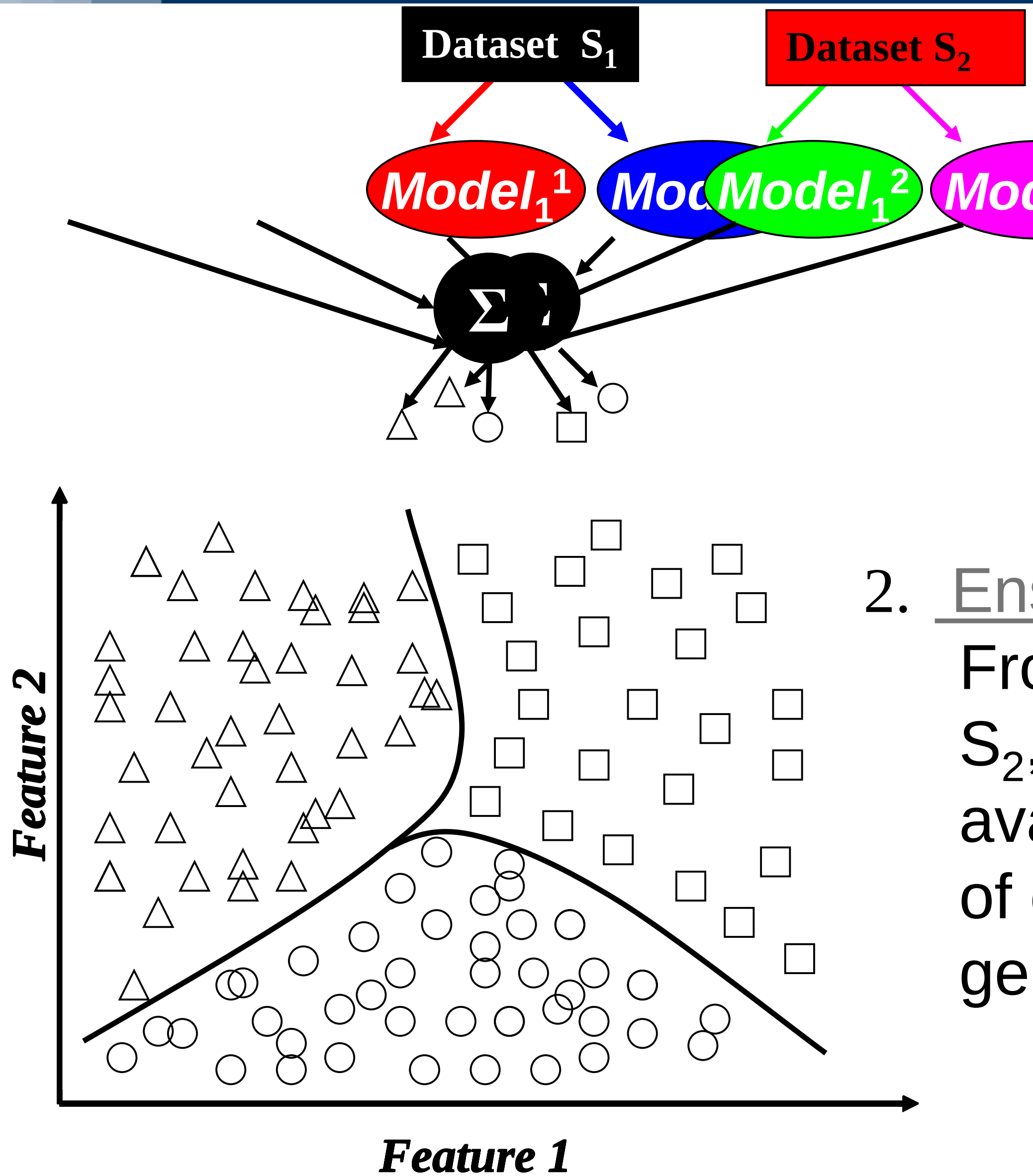


How?

1. AdaBoost:
   - Sequential production of classifiers
   - Each classifier is dependent from the previous one
   - Examples that are incorrectly classified by previous classifiers are chosen more often for the next classifier training



How? Ensemble of Ensembles



2. Ensemble of Ensembles
   From each dataset S_1, S_2, … which becomes available, an ensemble of classifiers is generated

How? Consult and vote

- A vote is associated to each classifier
- Classifiers consult each other to determine their voting weight

Testing with new, class – 4 data

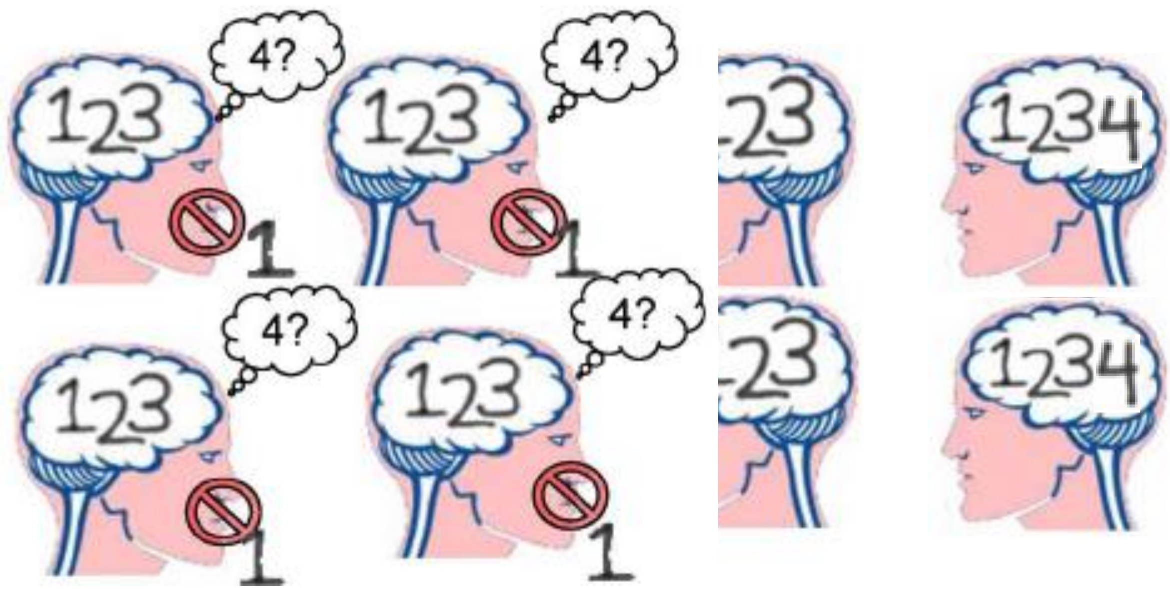
Ensemble 1 classifiers then realize that Ensemble 2 dominantly choose a new class on which they are not trained. Hence, Ensemble 1 classifiers decide to withdraw their decision

Ensemble 1 inevitably misclassifies

Ensemble 1 learns classes 1,2 and 3

Ensemble 2 learns classes 1,2,3 and 4

Ensemble 2 correctly classifies new data

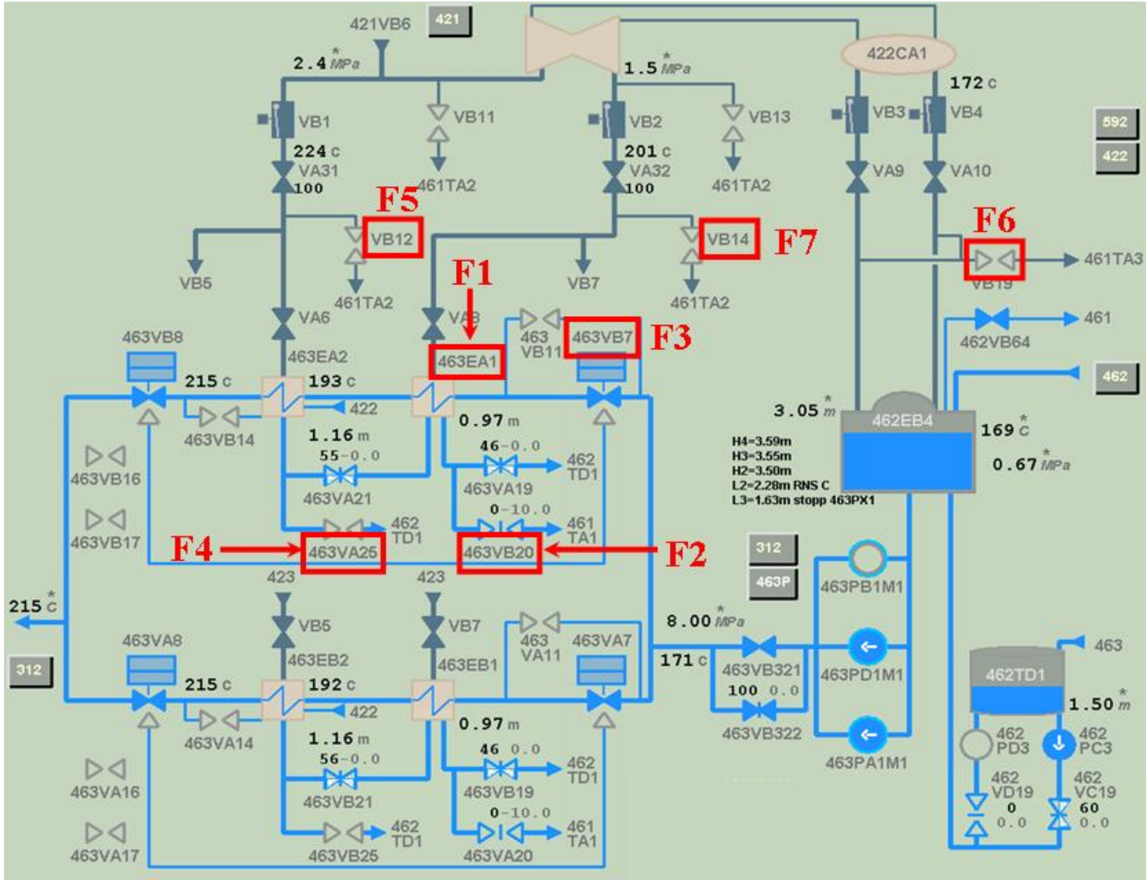


Adapted from: M. Muhlbaier, A. Topalis, and R. Polikar, "Learn++.nc: Combining ensemble of classifiers with dynamically weighted consult-and-vote for efficient incremental learning of new classes." *IEEE Transactions on Neural Networks*, vol. 20(1), pp. 152–168, 2009

- **Incremental learning:**
  - **What is it?**
  - **Why ensembles?**
  - **How?**
  - **Application: fault diagnostics in a nuclear power plant system**

Application: fault diagnostics in a nuclear power plant system

- Forsmark 3 BWR Nuclear Power Plant (Sweden)



Data provided by the Institute for Energy - Halden Reactor Project (Norway)

Application: fault diagnostics in a nuclear power plant system

- System faults:
  - C_1 - Leakage in the first high pressure preheater to the drain tank
  - C_2 - Leakage through the second high-pressure preheater
  - C_3 - Leakage through the first high pressure preheater Drain back-up valve to the condenser
  - C_4 - Leakage through the high-pressure preheaters bypass valve
  - C_5 - Leakage through the second high-pressure preheater drain back-up valve to the feedwater tank

Application: fault diagnostics in a nuclear power plant system

- Signals Selected for the classification:
  - Temperature in drain 4 before VB3 [°C]
  - Water level in tank TD1 [m]
  - Feedwater Temperature after EA2 in train A [°C]
  - Feedwater Temperature after EB2 in train B [°C]

Application: fault diagnostics in a nuclear power plant system

- Dataset 1

Number of patterns						
Training Dataset	Normal	Class 1	Class 2	Class 3	Class 4	Class 5
S_1	90	60	60	60	60	-

Number of classifiers: 10

Ensemble base classifiers: neural network

Diagnostic Model DM_1

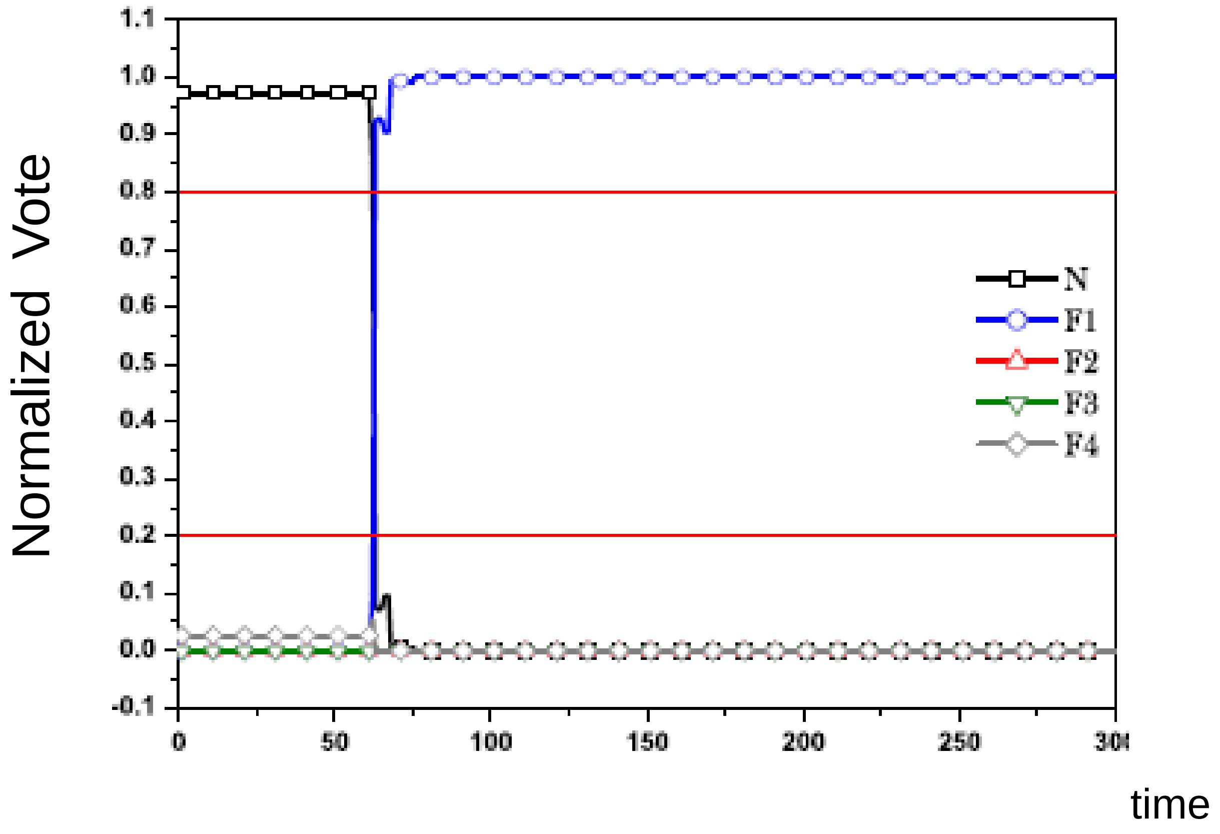
All patterns of class 5 are misclassified!

F

Testset	classes	Performance
S_1^{Test}	1,2,3,4	0.981
S_2^{Test}	1,2,3,4,5	0.830

Application: fault diagnostics in a nuclear power plant system

➢ Classification of a Transient of Class 1

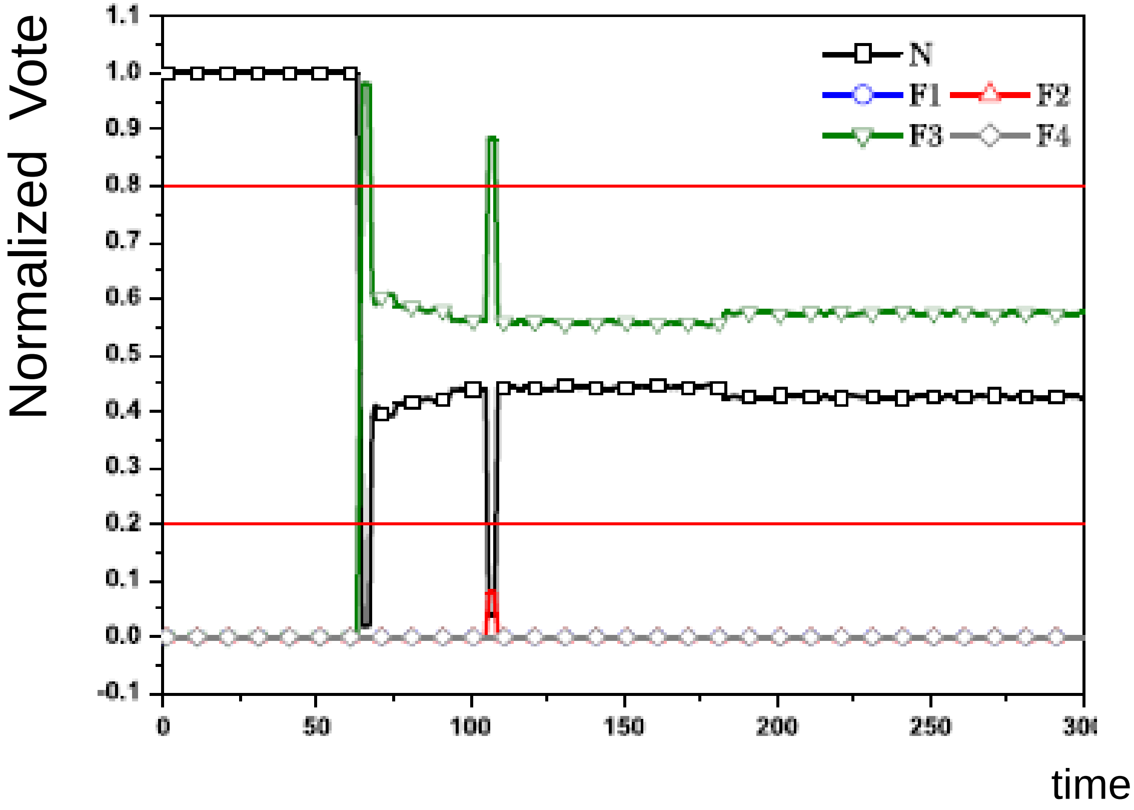


High Vote

High Confidence

Application: fault diagnostics in a nuclear power plant system

➢ Classification of a Transient of Class 5



Low Vote

Low Confidence

Application: fault diagnostics in a nuclear power plant system

➢ Dataset *2*

Number of patterns						
Training Dataset	Normal	Class 1	Class 2	Class 3	Class 4	Class 5
S_1	90	60	60	60	60	-
S_2	90	60	60	60	90	60

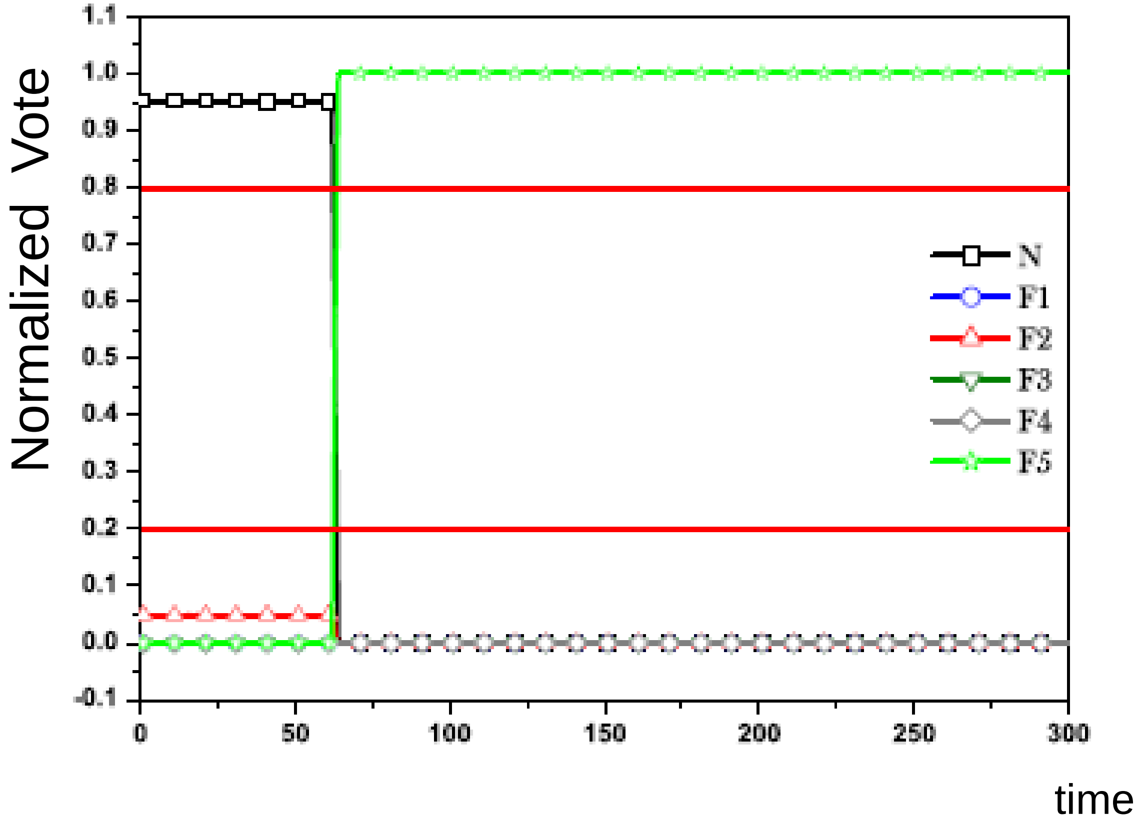
Update of diagnostic model DM_1 with 10 new classifiers

Diagnostic Model DM2

Testset	classes	Performance
S_1^{Test}	1,2,3,4	0.984
S_2^{Test}	1,2,3,4,5	0.987

Application: fault diagnostics in a nuclear power plant system

➢ Classification of a Transient of Class 5



Conclusions

- Ensembles of Models for PHM:
  - Accuracy
  - Confidence Estimation
  - Incremental Learning

Acknowledgments

Dr. Sameer Al-Dahidi

Allegra Alessi

Francesco Cannarile

Dr. Michele Compare

Dr. Giulio Gola

Dr. Francesca Mangili

Dr. Roozbeh Razavi

Marco Rigamonti

Sergio Sauco

References

S. Al-Dahidi, F. Di Maio, P. Baraldi, E. Zio, R. Seraoui, “*A novel ensemble clustering for operational transients classification with application to a nuclear power plant turbine*”, International Journal of Prognostics and Health Management, 6 (SP3), pp. 1-21, 2015.

P. Baraldi, E. Zio, F. Mangili, G. Gola, B. H. Nystad, “*Ensemble of Kernel Regression Models for Assessing the Health State of Choke Valves in Offshore Oil Platforms*”, International Journal of Computational Intelligence Systems, Vol. 7(2), pp. 225-241, 2014.

P. Baraldi, F. Mangili, G. Gola, B. H. Nystad, E. Zio “*A Hybrid Ensemble-Based Approach for Process Parameter Estimation and Degradation Assessment in Offshore Oil Platforms*”, International Journal of Performability Engineering, Vol. 10(5), pp. 497-509, 2014.

P. Baraldi, F. Mangili, E. Zio, “*Investigation of uncertainty treatment capability of model-based and data-driven prognostic methods using simulated data*” Reliability Engineering and System Safety, Vol. 112, pp. 94-108, 2013

P. Baraldi, F. Cadini, F. Mangili, E. Zio, “*Model-based and data-driven prognostics under different available information*”, Probabilistic Engineering Mechanics, Vol. 32, pp. 66-79, 2013.

P. Baraldi, F. Mangili and E. Zio, Ensemble of Bootstrapped Models for the prediction of the Remaining Useful Life of a Creeping Turbine Blade, 2012 IEEE International Conference on Prognostics and Health Management, 2012.

P. Baraldi, R. Razavi-Far, E. Zio, "*Classifier-Ensemble Incremental-Learning Procedure for Nuclear Transient Identification at Different Operational Conditions*", Reliability Engineering and System Safety, Vol. 96, pp. 480–488, 2011.

P. Baraldi, M. Compare, E. Zio, "*Identification of contradictory patterns in experimental datasets for the development of models for electrical cables diagnostics*". special issue: Data Mining in Reliability, Availability, Maintainability, and Risk Management of International Journal of Performability Engineering, Vol. 7 (1), pp. 43-60, 2011.

P. Baraldi, R. Razavi-Far, E. Zio, "*Bagged Ensemble of FCM Classifiers for Nuclear Transient Identification*", Annals of Nuclear Energy; Vol. 38 (5), pp 1161-1171, 2011.

P. Baraldi, E. Zio, G. Gola, D. Roverso, M. Hoffmann, "*Signal reconstruction by a GA-optimized ensemble of PCA models*", Nuclear Engineering and Design, Vol. 241 (1), pp. 301-309, 2011.

P. Baraldi, E. Zio, G. Gola, D. Roverso, M. Hoffmann, "*Two novel procedures for aggregating randomized model ensemble outcomes for robust signal reconstruction in nuclear power plants monitoring systems*", Annals of Nuclear Energy, Vol. 38 (2-3), pp. 212-220, 2011.

References

P. Baraldi, G. Gola, E. Zio, D. Roverso, M. Hoffmann, "*A randomized model ensemble approach for reconstructing signals from faulty sensors*". Expert Systems With Application, Vol. 38 (8), pp. 9211-9224, 2011.

P. Baraldi, A. Cammi, F. Mangili, E. Zio, "*Local Fusion of an Ensemble of Models for the Reconstruction of Faulty Signals*" - IEEE Transactions on Nuclear Science, Vol. 57 (2), pp.793-806, 2010.

P. Baraldi, A. Cammi, F. Mangili, E. Zio, "*An ensemble approach to sensor fault detection and signal reconstruction for nuclear system control*" - Annals of Nuclear Energy, Vol. 37 (6), pp. 778-790, 2010.

P. Baraldi, E. Zio, G. Gola, D. Roverso, M. Hoffmann, "*Robust nuclear signal reconstruction by a novel ensemble model aggregation procedure*", International Journal of Nuclear Knowledge Management, Vol 4 (1), pp. 34-41, 2010.

Breiman, L., 1999. Combining predictors, in Combining Artificial Neural Nets. A. J. C. Sharkey (ed.), Springer.

Breiman, L., 1996. Bagging predictors. Machine Learning. 24 (2), pp. 123–140.

F. Di Maio, J. Hu, P. Tse, K. Tsui, E. Zio, M. Pecht, “Ensemble-approaches for clustering health status of oil sand pumps”, Expert Systems with Applications, 2011.

L. Kuncheva, *Combining pattern classifiers, methods and algorithms. Hoboken, NJ: Wiley Interscience, 2004.*

M. Muhlbaier, A. Topalis, and R. Polikar, “Learn++.nc: Combining ensemble of classifiers with dynamically weighted consult-and-vote for efficient incremental learning of new classes.” *IEEE Transactions on Neural Networks,* vol. 20(1), pp. 152–168, 2009

Polikar, R., 2006. Ensemble Based Systems in Decision Making. IEEE Circuits and Systems Magazine, 6 (3), pp. 21-45.

Polikar, R., Udpa, L., Udpa, S.S., Honavar, V., 2001. Learn++: An incremental learning algorithm for supervised neural networks. IEEE transactions on systems, man, and cybernetics-part C: Applications and reviews, Vol. 31, No. 4.

R. Razavi-Far, P. Baraldi, E. Zio, “*Dynamic Weighting Ensembles for Incremental Learning and Diagnosing New Concept Class Faults in Nuclear Power Systems*”, IEEE Transactions On Nuclear Science, Vol. 59 (5), pp. 2520-2530, 2012.

F. Xue, P. Bonissone, A. Varma, W. Yan, N. Eklund, K. Goebel (2008), An Instance-Based Method for Remaining Useful Life Estimation for Aircraft Engines, *Journal of Failure Analysis and Prevention, (8)-2:199-206, April 2008*

References

K. Woods, WP. Kegelmeyer, and K. Bowyer (1997), Combination of multiple classifiers using local accuracy estimates, *IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(4):405-410*

E. Zio, P. Baraldi, G. Gola, “Feature-based classifier ensembles for diagnosing multiple faults in rotating machinery”. Applied Soft Computing, Vol. 8 (4), pp. 1365–1380, 2008.

E. Zio, P. Baraldi, G. Gola, D. Roverso, M. Hoffmann “Diagnosing faults in nuclear components by an ensemble of feature-diverse fuzzy classifiers”, International Journal of Nuclear Knowledge Management (IJNKM), Vol. 2 (3), pp 224-238, 2007.

E. Zio, P. Baraldi and G. Gola, “Ensemble feature selection for diagnosing multiple faults in rotating machinery”, Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, Vol. 221 (O1), pp. 29-41, 2007.