Labex Action
Integrated smart systems

femto-st SCIENCES & TECHNOLOGIES

ensmm

Prognostics in presence of imprecise health indicator based on computational geometry and similarity-based approach: a case study on C-MAPSS datasets

Keywords: *Computational geometry, Prognostics, Health Indicator estimation, Imprecise health indicator, Similarity-based approach, CMAPSS datasets*

Emmanuel RAMASSO

Seminar GDR MACS / GT S3
June 25, 2014, Paris, France

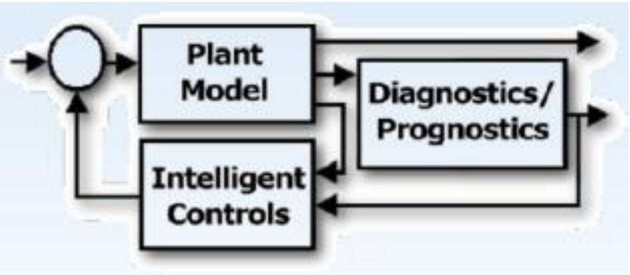
Context & Motivations

Prognostics and Health Management (PHM)

LabeXction
Integrated smart systems

Through the **interpretation** of sensor measurements (D. Simon, 2012):

- **Health assessment** – Early signs of wear
  - Accurately assess system / component's performance deterioration over the lifetime
  - Accurately detect and isolate any engine system and/or instrumentation malfunctions
- **Prognostics** – Remaining Useful Life (**RUL**)
  - Predict the state of health in future use
  - Provide a degree of confidence in predictions



From S. Garg and D. Simon, Controls and Dynamics Branch, NASA Glenn Research Center

↘ Costs

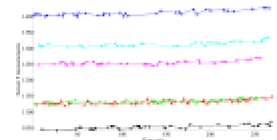
↗ Security

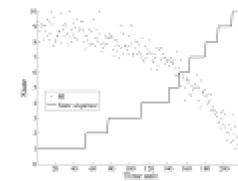
↗ Availability

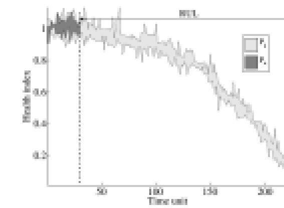
Taxonomy of PHM approaches

- Category 1 – **Data driven** approaches
- Category 2 – **Physics-based** approaches

Context & Motivations
PHM taxonomy





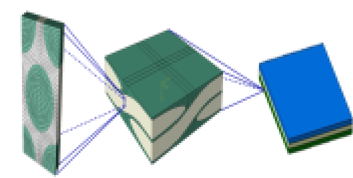


Data driven approaches:

- Use **sensor data** (reflecting the system's behavior)
- Use **pattern recognition** / machine learning / data mining to:
  - Detect a « change » in the system: Mapping data → health state
  - Predict the RUL: Mapping « data → RUL » or « past data → future data »

Model/Physics-based approaches:

- Use **physical models** of the system for RUL estimation
- Modeling physics relies on **defining relationships** between
  - Degradation of components / subsystems and operational conditions
  - Multiple components (inputs, state variables, measurements)

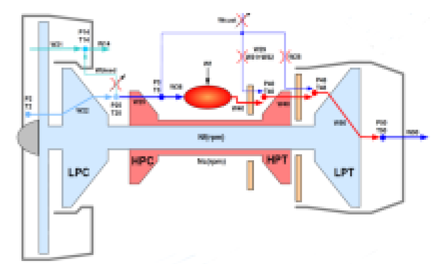


http://www.ifam.rwth-aachen.de/

In both cases, the « model » is a **simplified representation** of the system's behavior: **Uncertainty management** required aiming at compensating modeling errors

http://airchive.com/

≠



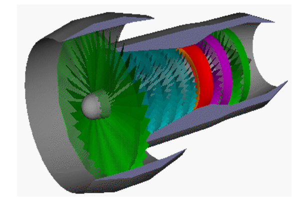
http://www.grc.nasa.gov/

Outline

Data-driven models for PHM under uncertainty

- **Part 1: Turbofan engine**
  - Presentation of turbofan engine and CMAPSS
  - Datasets description (complexity illustrated)

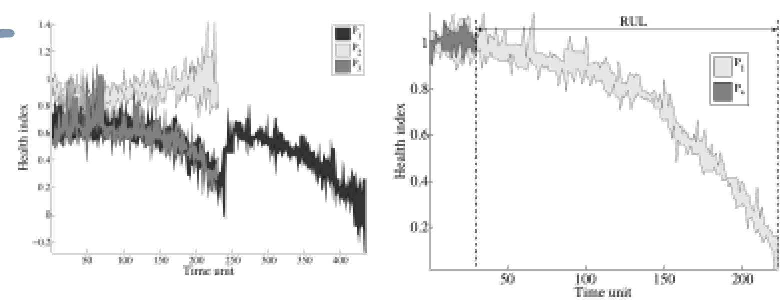
http://airchive.com/



http://www.grc.nasa.gov/

- **Part 2: RUL estimation**
  - Sensor data have high variability
  - No knowledge about the data generating process, complex noise

***RUL-CLIPPER: R**emaining **U**seful **L**ife estimation based on impre**C**ise hea**L**th **I**ndex modeled by sim**P**le **P**olygons and similarity-bas**E**d **R**easoning*

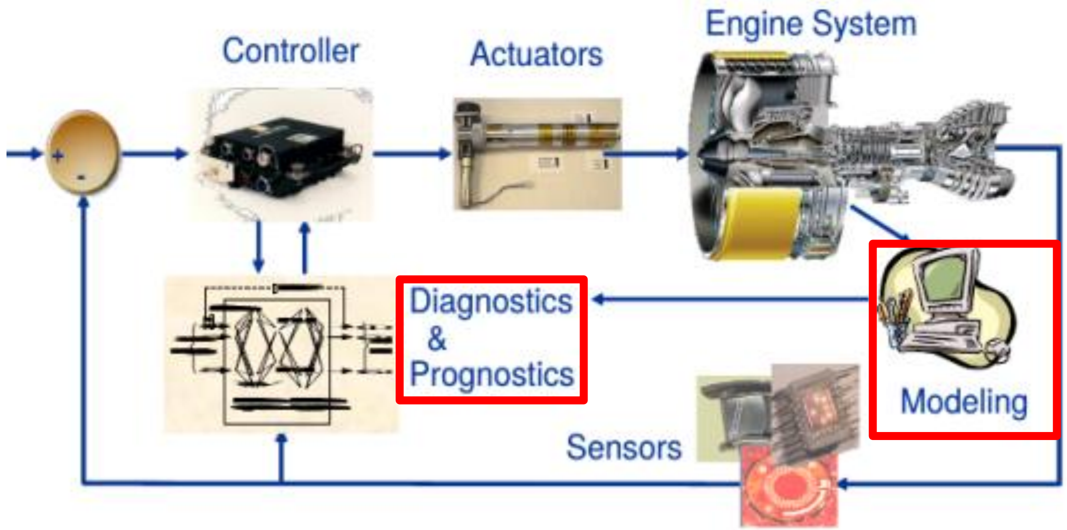


Part 1: Turbofan engine

Turbofan engine PHM

PHM in the loop

http://airchive.com/



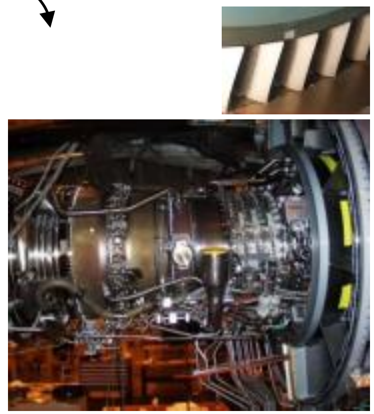
- Engine control using an **engine model** to provide guaranteed **performance** throughout the life of the engine, but engine models are **imperfect**
- Model used to generate datasets for enhanced on-board turbofan PHM:
  - No two engines are the same, sensors not modeled correctly, model inaccuracy during transients and at off-design operating conditions, models not updated once engine into production (design changes not always)…
  - **Hybrid** modeling (analytical + empirical) techniques hold promise for capturing engine-model mismatch

From Don Simon, Controls and Dynamics Branch, NASA Glenn Research Center

Modeling physics of a turbofan engine

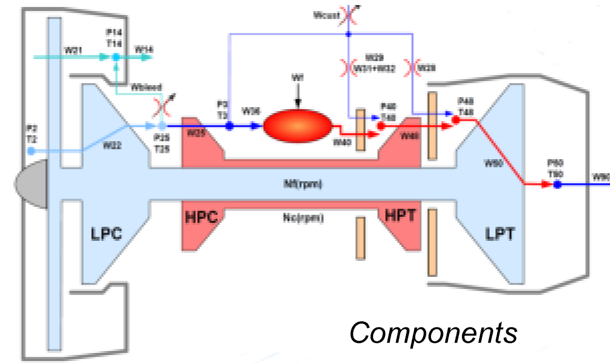
LabeXction
Integrated smart systems

http://airchive.com/



http://www.grc.nasa.gov/

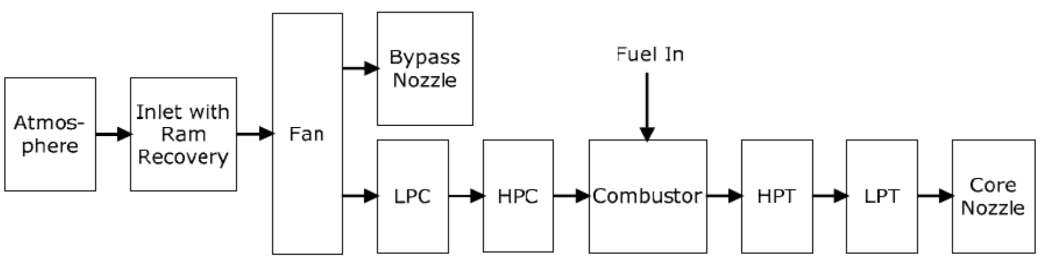
(S. Garg and D. Simon, 2012)



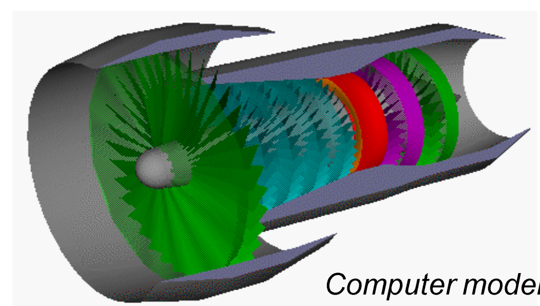
Components

http://www.grc.nasa.gov/

Commercial Modular Aero-Propulsion System Simulation (CMAPSS)
(Frederick, D., DeCastro, J., & Litt, J., 2007)



Simulink model

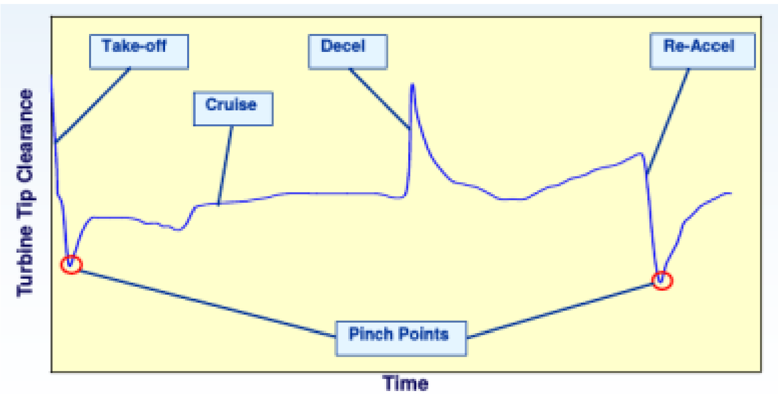


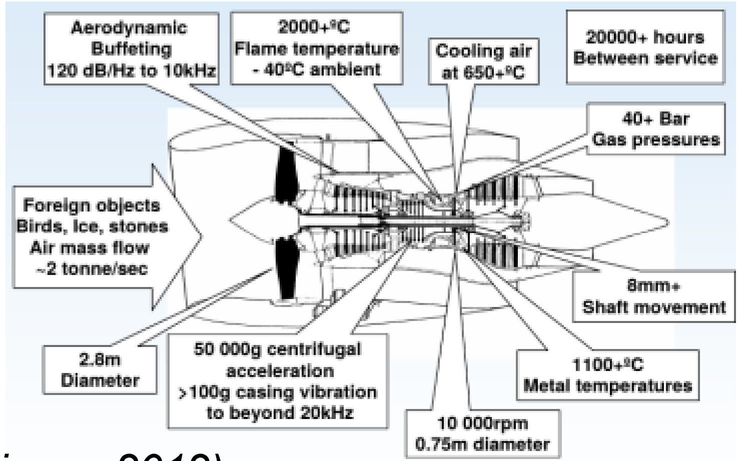
Computer model

http://www.grc.nasa.gov/

Turbofan engine PHM
Simulated datasets on CMAPSS

- CMAPSS (Commercial Modular Aero-Propulsion System Simulation) datasets (Saxena, A., Goebel, K., Simon, D., & Eklund, N., 2008)
- Datasets were generated for prognostics development: Simulating of various operational conditions + faults injection + varying wear degree





(S. Garg and D. Simon, 2012)

Turbofan engine PHM
Simulated datasets on CMAPSS

- CMAPSS (Commercial Modular Aero-Propulsion System Simulation) datasets (Saxena, A., Goebel, K., Simon, D., & Eklund, N., 2008)
- Datasets were generated for prognostics development: Simulating of various operational conditions + faults injection + varying wear degree

Sensor data

IDX	Symbol	Meaning	Unit
6	T2	Tot. temp. at fan inlet	°R
7	T24	Tot. temp. at LPC outlet	°R
8	T30	Tot. temp. at HPC outlet	°R
9	T50	Tot. temp. at LPT outlet	°R
10	P2	Pressure at fan inlet	psia
11	P15	Tot. pressure in bypass-duct	psia
12	P30	Tot. pressure at HPC outlet	psia
13	Nf	Physical fan speed	rpm
14	Nc	Physical core speed	rpm
15	epr	Engine pressure ratio (P50/P2)	–
16	Ps30	Static pressure at HPC outlet	psia
17	phi	Ratio of fuel flow to Ps30	pps/psi
18	NRf	Corrected fan speed	rpm
19	NRc	Corrected core speed	rpm
20	BPR	Bypass Ratio	–
21	farB	Burner fuel-air ratio	–
22	htBleed	Bleed Enthalpy	–
23	Nf_d	Demanded fan speed	rpm
24	PCNfR_d	Demanded corrected fan speed	rpm
25	W31	HPT coolant bleed	lbm/s
26	W32	LPT coolant bleed	lbm/s

Operating conditions

Symbol	Meaning	Unit	Range
TRA	Throttle Resolver Angle	%	[20, 100]
M	Mach number	Mach	[0, 0.84]
ALT	Altitude	Kft	[0, 42]

TRA: pilot power request

Turbofan engine PHM
Simulated datasets on CMAPSS

Labex Action
Integrated smart systems

❖ CMAPSS (Commercial Modular Aero-Propulsion System Simulation) datasets (Saxena, A., Goebel, K., Simon, D., & Eklund, N., 2008)

Datasets characteristics

Datasets	#1	#2	#3	#4	#5T	#5V
Nb. fault modes	1	1	2	2	1	1
Nb. op. conditions	1	6	1	6	6	6
Nb. training units	100	260	100	249	218	218
Nb. testing units	100	259	100	248	218	435
Minimum RUL	7	6	6	6	10	6
Maximum RUL	145	194	145	195	150	190

~1360 simulated turbofan run-to-failure testing data (trajectories / instances)

Datasets available since the 2008's Int. Conference on PHM

- #1, #2, #3, #4: « Turbofan engine datasets »
- #5, #6: « Data challenge » or « PHM'08 datasets »

Scoring function (official performance measure called **timeliness, to be minimized)**

$$S = \sum_{n=1}^{N} S_n \qquad S_n = \begin{cases} e^{-d_n/13} - 1, d_n \leq 0 \\ e^{d_n/10} - 1, d_n > 0 \end{cases}, n = 1 \ldots N$$

$$d_n = \text{estimated RUL} - \text{true RUL}$$

During 2008-2014 period:

- ~7100 downloads of datasets #1-#4, ~2000 of datasets #5-#6
- 70 papers using those datasets (Google Scholar)
- 5 papers focused on prognostics using full training/testing datasets as originally provided by the organisers

Prognostics on CMAPSS

From sensor measurements to a health indicator

- Operating conditions can be clustered (grouped) into 6 main operating conditions (OC), (Wang 2010, Richter 2012)
- For the *i*-th training trajectory, sensor data are grouped in each OC
- **Hyp. 1:** The sensor measurements do not vary too much between two consecutive samples in the same regime → The evolution is **locally linear**
- **Hyp. 2:** The health state is **monotonically** decreasing → An **exponential** model can be used to represent the global evolution of the health index (theoretical output)

$$\hat{\mathrm{HI}}_i(\mathbf{x}_t, \boldsymbol{\theta}^p) \equiv 1 - \exp\left(\frac{\log(0.05)}{0.95 \cdot T_i} \cdot t\right) \;, t \in [\sigma_1, \sigma_2].$$

- Deduce a regression model in each regime

$$\mathrm{HI}_i(\mathbf{x}_t, \boldsymbol{\theta}_i^p) = \theta_{i,0}^p + \sum_{n=1}^{q} \theta_{i,n}^p \cdot x_{t,n}$$

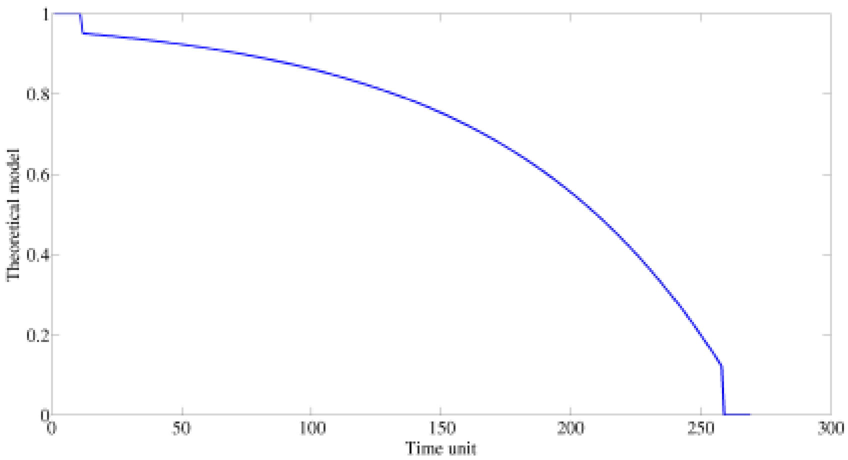
This approach was used on all (six) CMAPSS datasets

Presented in [E. Ramasso, Investigating computational geometry for failure prognostics, Int. Journal on PHM and in PHM-E conference, accepted, 2014].

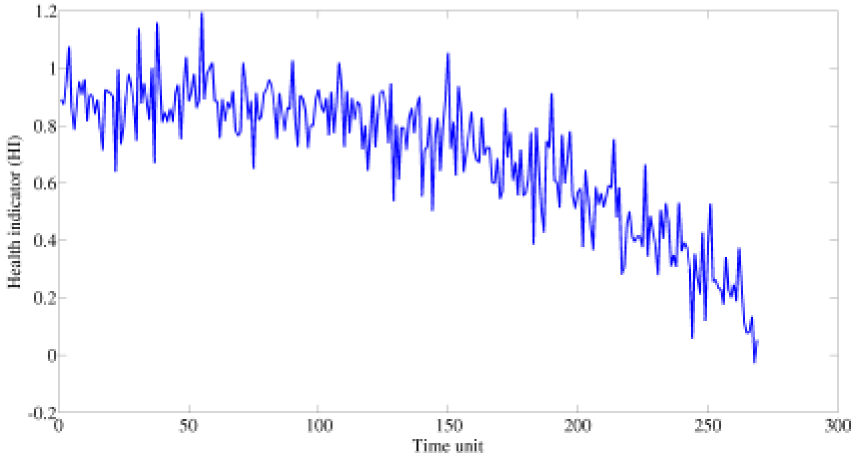
Prognostics on CMAPSS

From sensor measurements to a health indicator

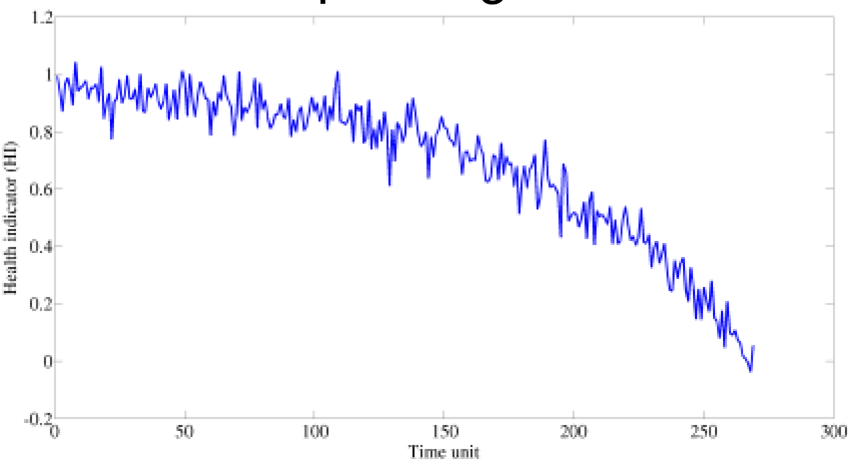
Theoretical model (Hyp. 2)



HI **without** operating conditions



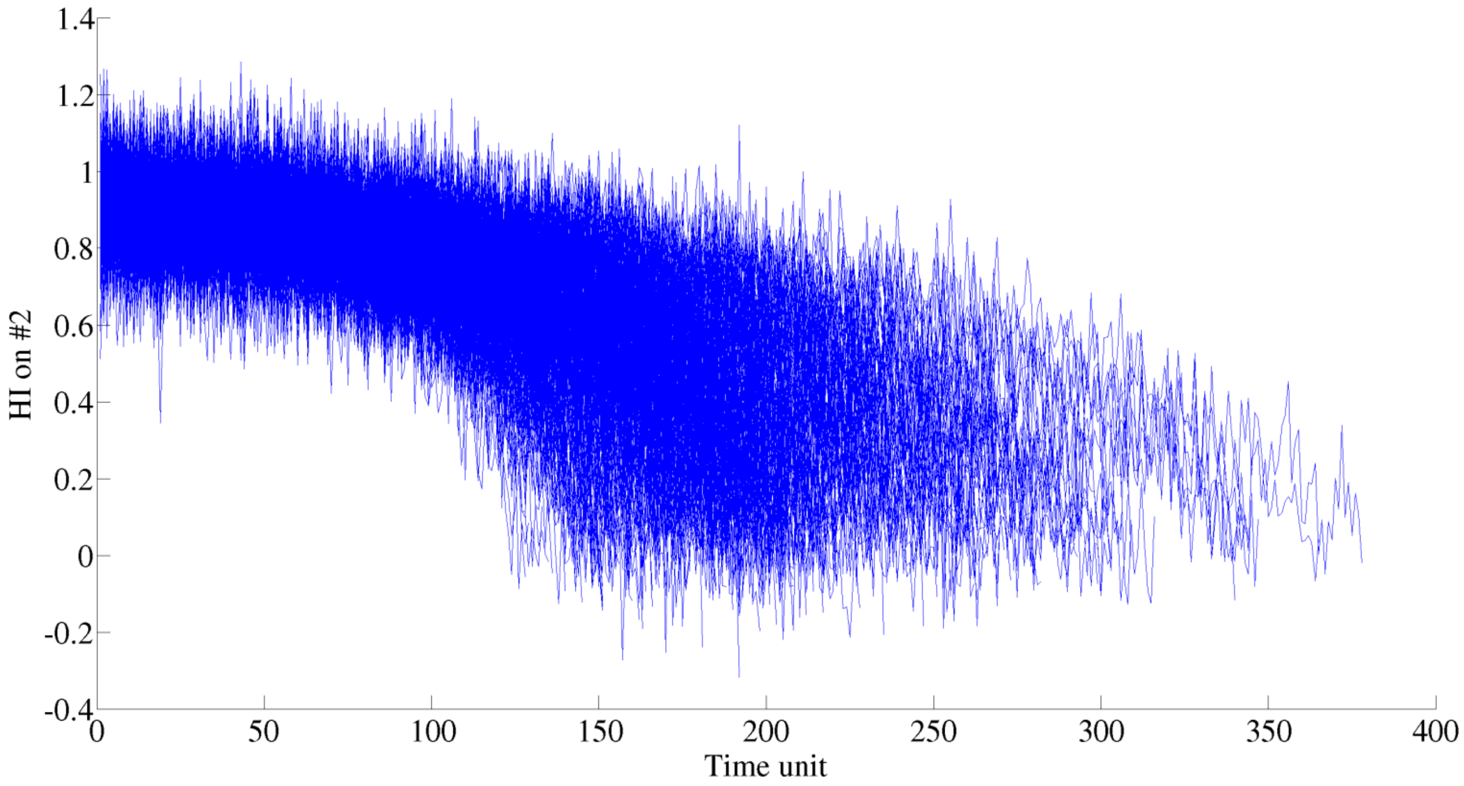
HI **with** operating conditions



Prognostics on CMAPSS

From sensor measurements to a health indicator

All HI computed on dataset #2 (6 OC, 1 Fault)



LabeX Action
Integrated smart systems

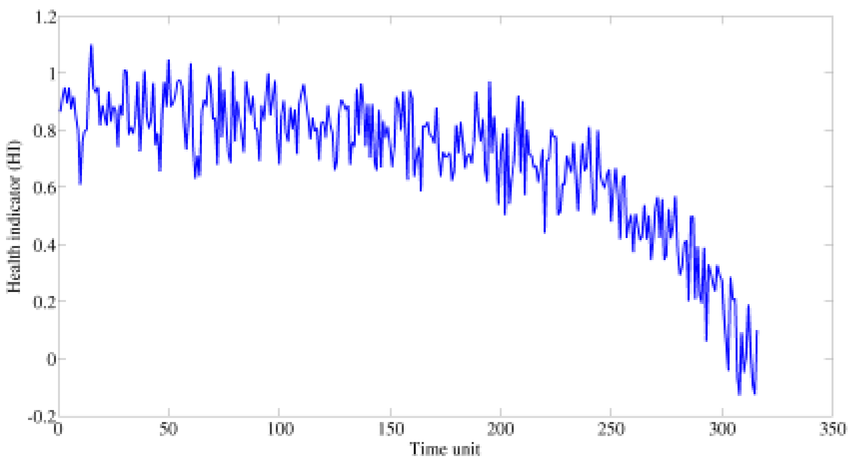
Part 2: RUL estimation (RULCLIPPER algorithm)

RULCLIPPER Algorithm

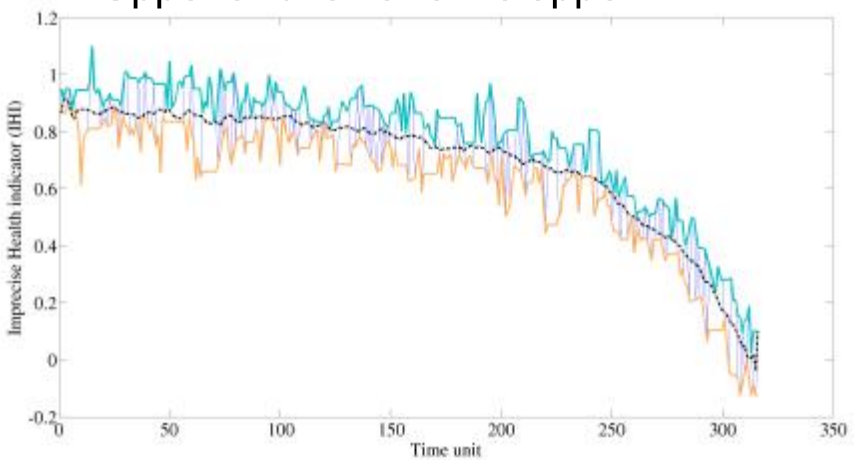
Imprecise Health Indicators

- Based on HI, usual approaches generally use a **functional form** (approx.) of the HI: locally linear, exponential, quadratic, neural networks…
- These approximations are then used for **extrapolation** for RUL estimation
- Problem: **variability of measurements is high**, so that noise models / variance-based parameters (embedded in methods) do not generalise well
- A rather unusual approach is proposed: a HI is interpreted as a « geometric figure », more exactly a **planar polygon** ⇨ Imprecise Health Indicator (IHI)

Health indicator

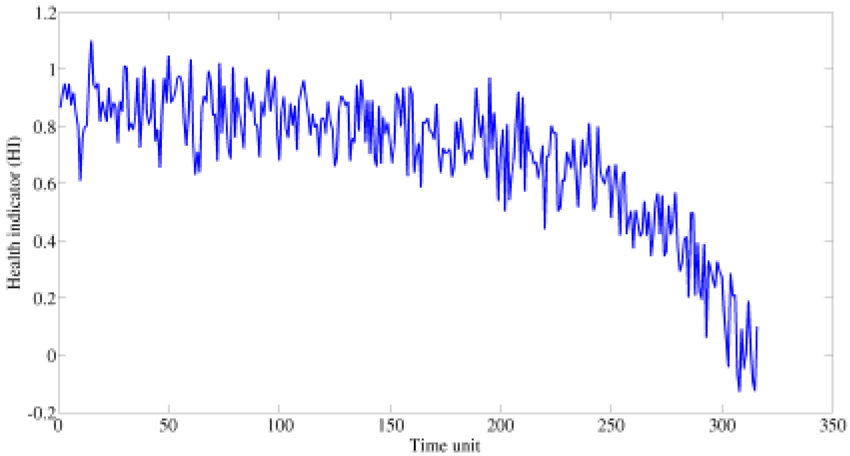


Upper and lower enveloppe: IHI

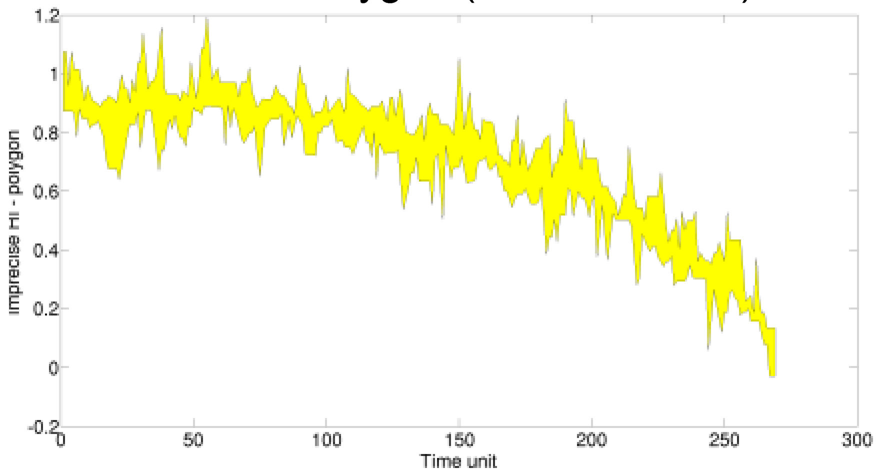


- Based on HI, usual approaches generally use a **functional form** (approx.) of the HI: locally linear, exponential, quadratic, neural networks…
- These approximations are then used for **extrapolation** for RUL estimation
- Problem: **variability of measurements is high**, so that noise models / variance-based parameters (embedded in methods) do not generalise well
- A rather unusual approach is proposed: a HI is interpreted as a « geometric figure », more exactly a **planar polygon** ⇨ Imprecise Health Indicator (IHI)

Health indicator

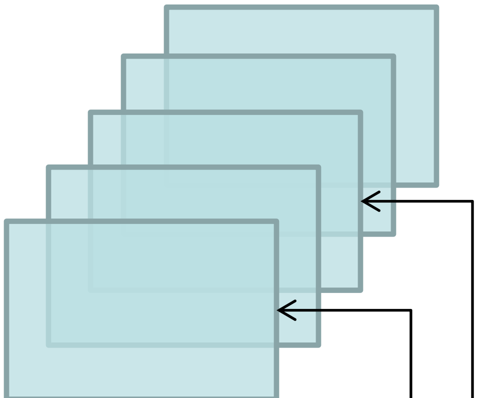


Model: Polygon (set of vertices)

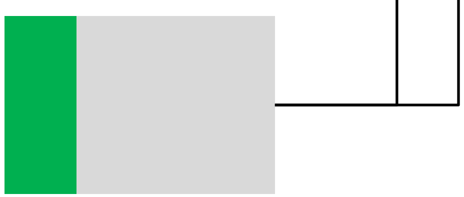


RULCLIPPER Algorithm
Library of IHI

- From each training data, an IHI is built
- A library of IHI (Imprecise Health Indicator) is obtained for each dataset with known RULs



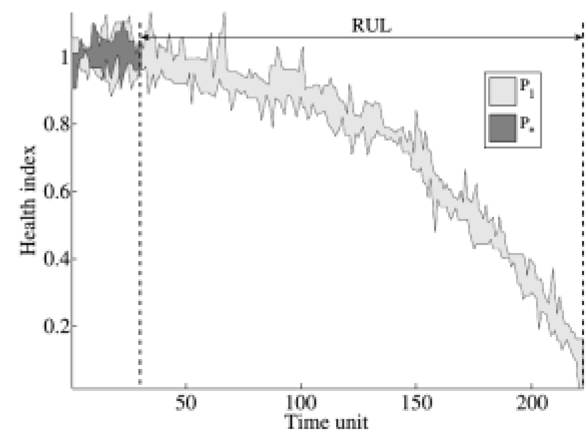
Library of training IHI:
Run-to-failure data



Testing IHI: Only an earlier portion of the run is known

Sort the training instances with respect to the **similarity**

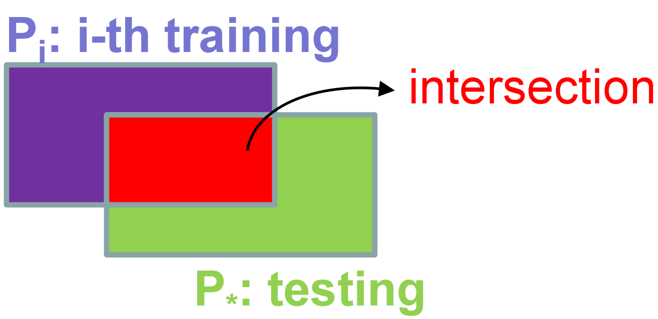
Combine the « closest » training runs (RUL fusion), to guess the RUL



RULCLIPPER Algorithm

Similarity-based matching of IHI

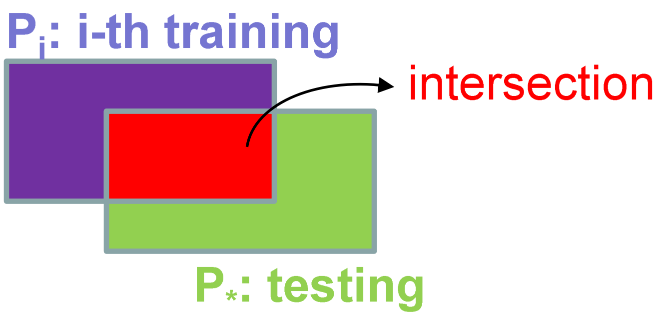
- « Closest » ⇨ **similarity measure**, that has to work between two IHI
- Solution: Area of the intersection of polygons (between testing/training data)
- Polygon intersection makes use of **computational geometry** tools, such as **generalised polygon clipping**: **Vatti's algorithm**, working on convex and non-convex polygons
- The area of intersection is then **converted** into a similarity measure:



$$\left.\begin{aligned} \mathcal{A}_\cap &= \mathrm{Area}\left(\mathcal{P}_i \cap \mathcal{P}_*\right) \\ R &= \frac{\mathcal{A}_\cap}{\mathcal{A}_i} \quad \text{(Recall)} \\ P &= \frac{\mathcal{A}_\cap}{\mathcal{A}_*} \quad \text{(Precision)} \end{aligned}\right\} \quad F_{1,i} = 2\frac{R \cdot P}{R + P}$$

Called F_1-measure, bounded in [0,1]

- Illustration of recall (R) and precision (P) measures



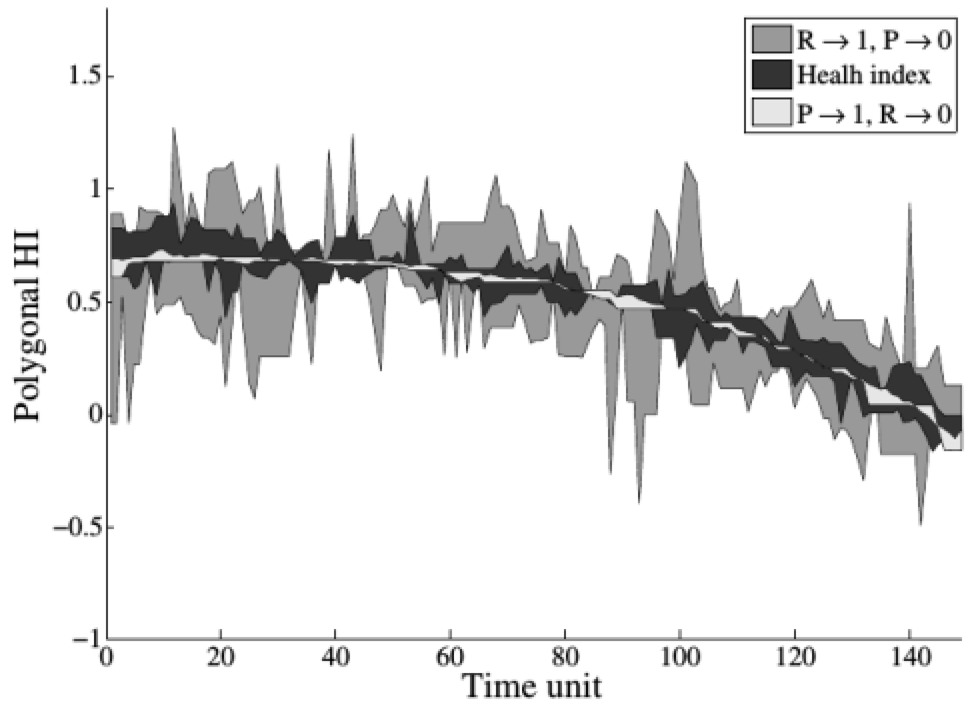
$$\mathcal{A}_\cap = \text{Area}\left(\mathcal{P}_i \cap \mathcal{P}_*\right)$$

$$R = \frac{\mathcal{A}_\cap}{\mathcal{A}_i} \quad \text{(Recall)}$$

$$P = \frac{\mathcal{A}_\cap}{\mathcal{A}_*} \quad \text{(Precision)}$$

$$F_{1,i} = 2\frac{R \cdot P}{R + P}$$

Called F_1-measure, bounded in [0,1]



RULCLIPPER Algorithm

In a nutshell

LabeXction
Integrated smart systems

- Inputs:
  - Training run-to-failure data (N-dimensional) with known RUL
  - Testing data
- Outputs:
  - RUL of testing data
  - Confidence degree
- Algorithm
  - Transform each training data *i* into HI_i
  - Transform HI_i into IHI_i (polygon) ⇨ library
  - Convert the testing data into IHI using the i-th model
  - Compute the degree of intersection with each IHI_i in the library (using Vatti’s algorithm + estimate similarity)

  RUL = weighted sum of the RULs of the K closest training data

  Confidence = Average degree of intersection

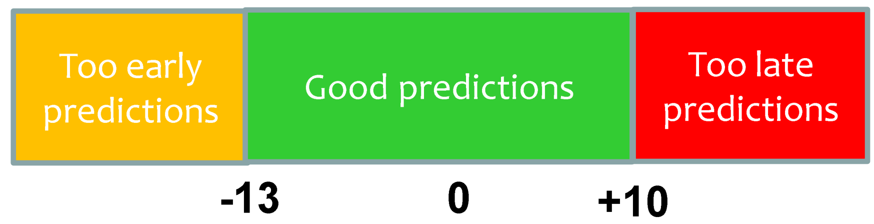
RULCLIPPER Algorithm

Evaluation of results on CMAPSS datasets

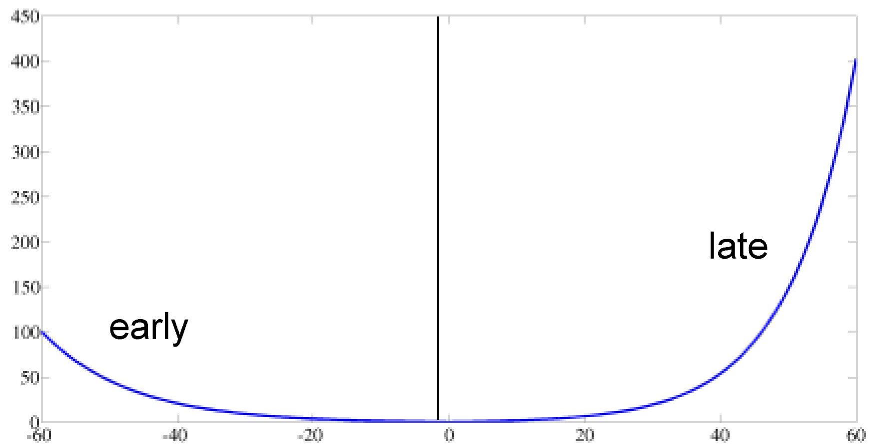
Labex Action
Integrated smart systems

Metrics for evaluation

Accuracy measure: number of predictions with an interval, **to be maximised**



Scoring function: penalises more the late predictions than early ones, **to be minimised**

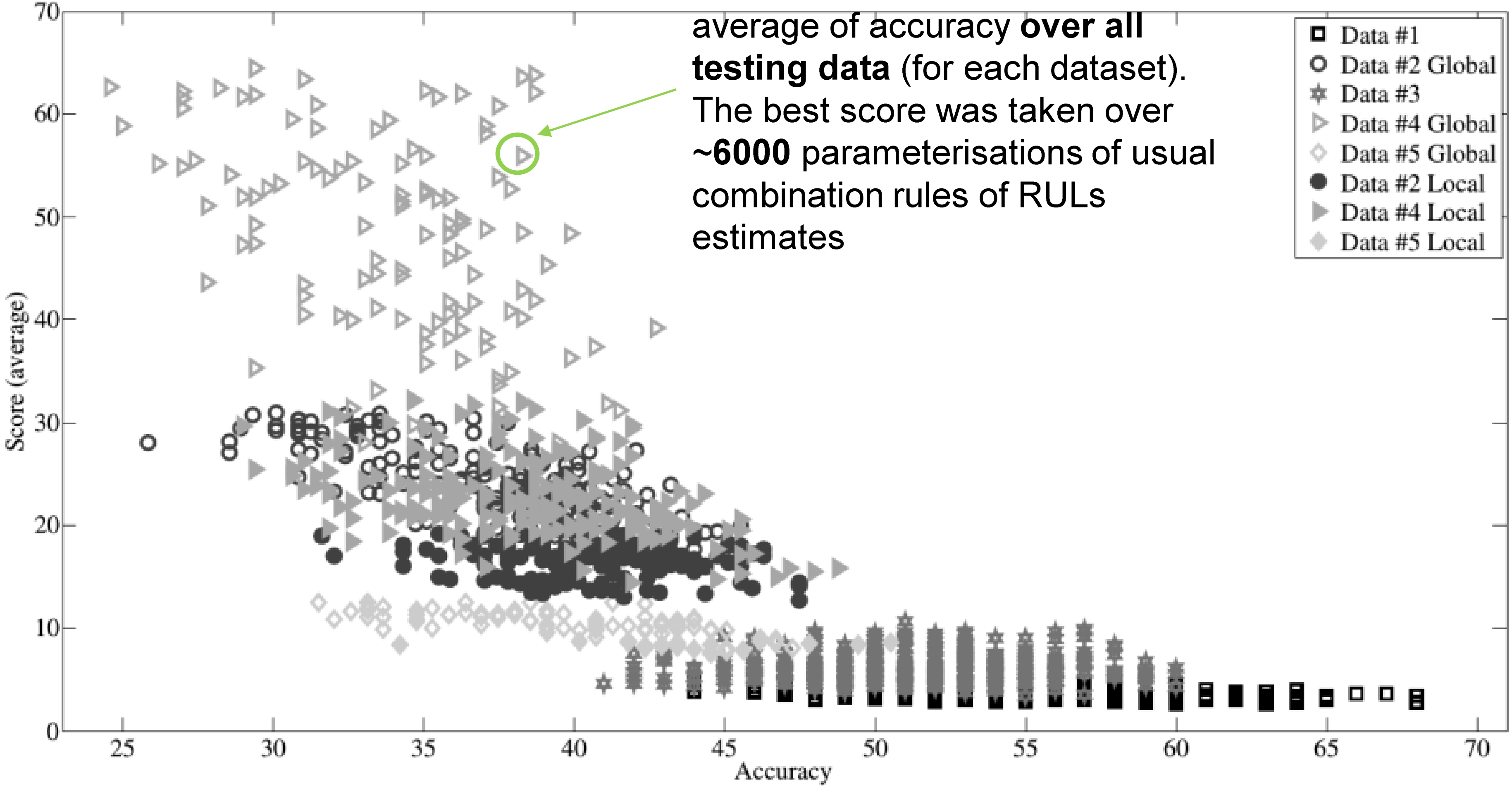


Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., & Schwabacher, W. (2008). Metrics for evaluating performance of prognostic techniques. In Int. conf. on prognostics and health management.

RULCLIPPER Algorithm

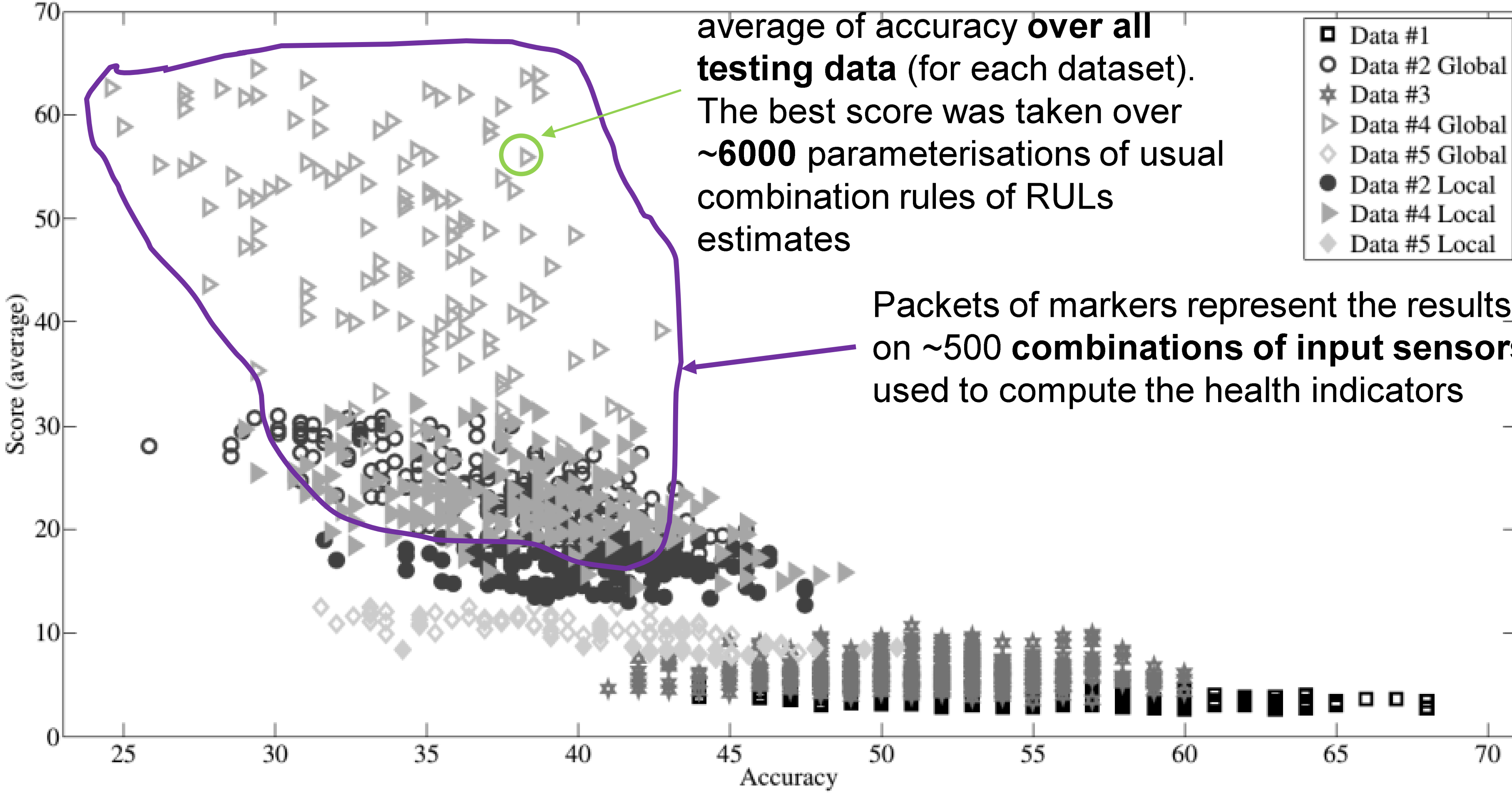
Results

Each marker has two coordinates: the sum of scores and the average of accuracy **over all testing data** (for each dataset). The best score was taken over **~6000** parameterisations of usual combination rules of RULs estimates



RULCLIPPER Algorithm

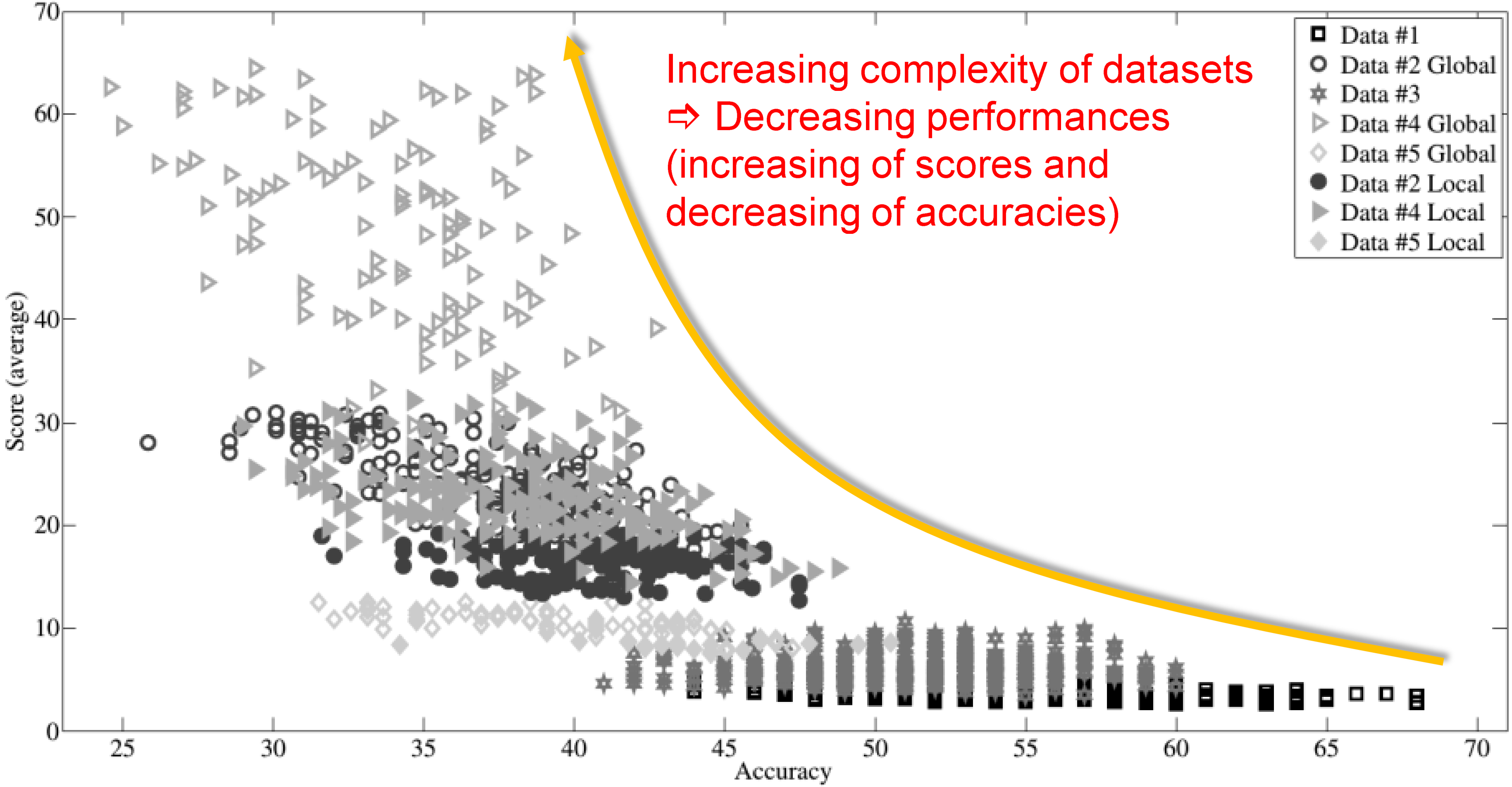
Results



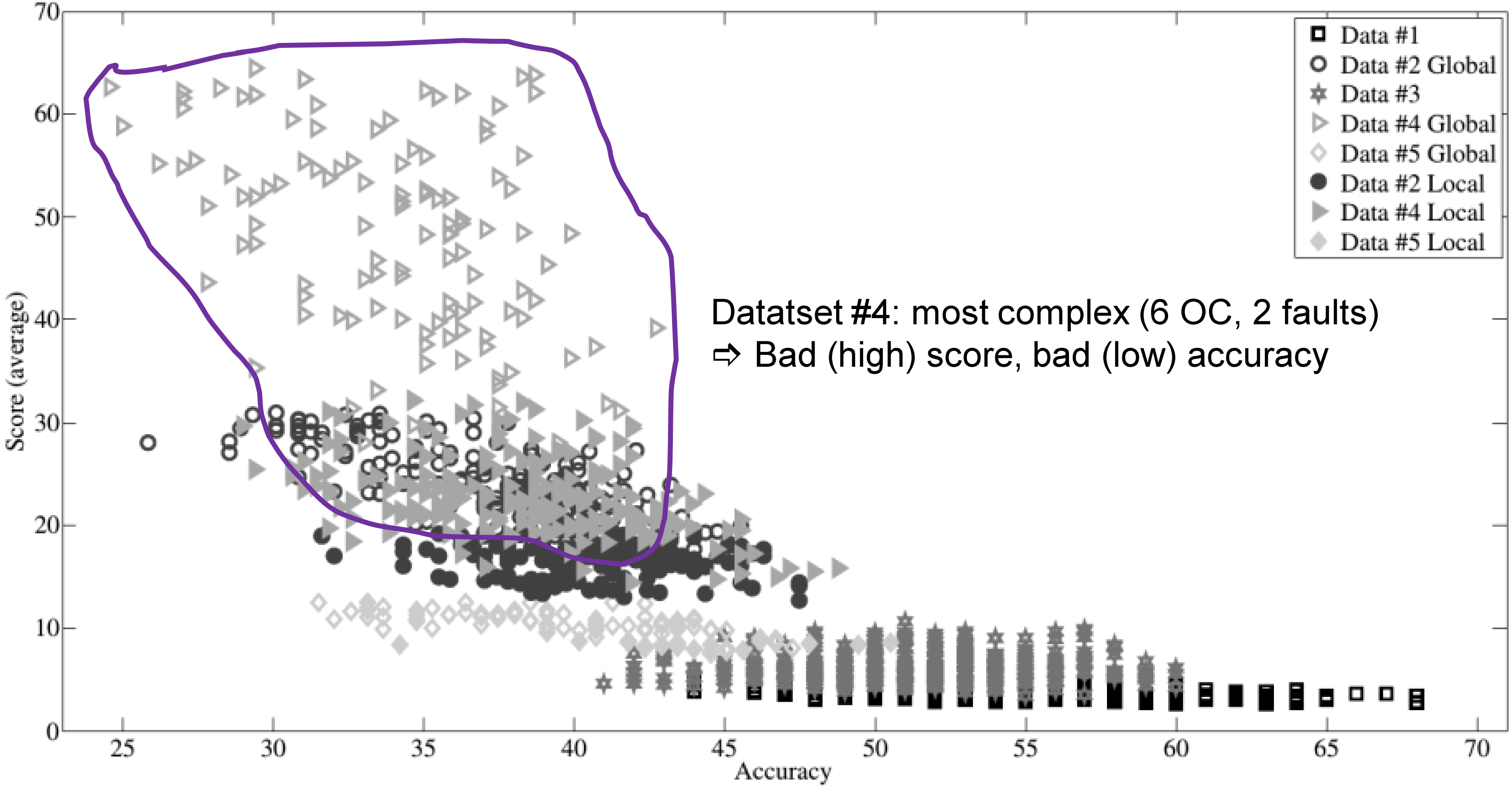
Each marker has two coordinates: the sum of scores and the average of accuracy **over all testing data** (for each dataset). The best score was taken over **~6000** parameterisations of usual combination rules of RULs estimates

Packets of markers represent the results on ~500 **combinations of input sensors** used to compute the health indicators

RULCLIPPER Algorithm
Results



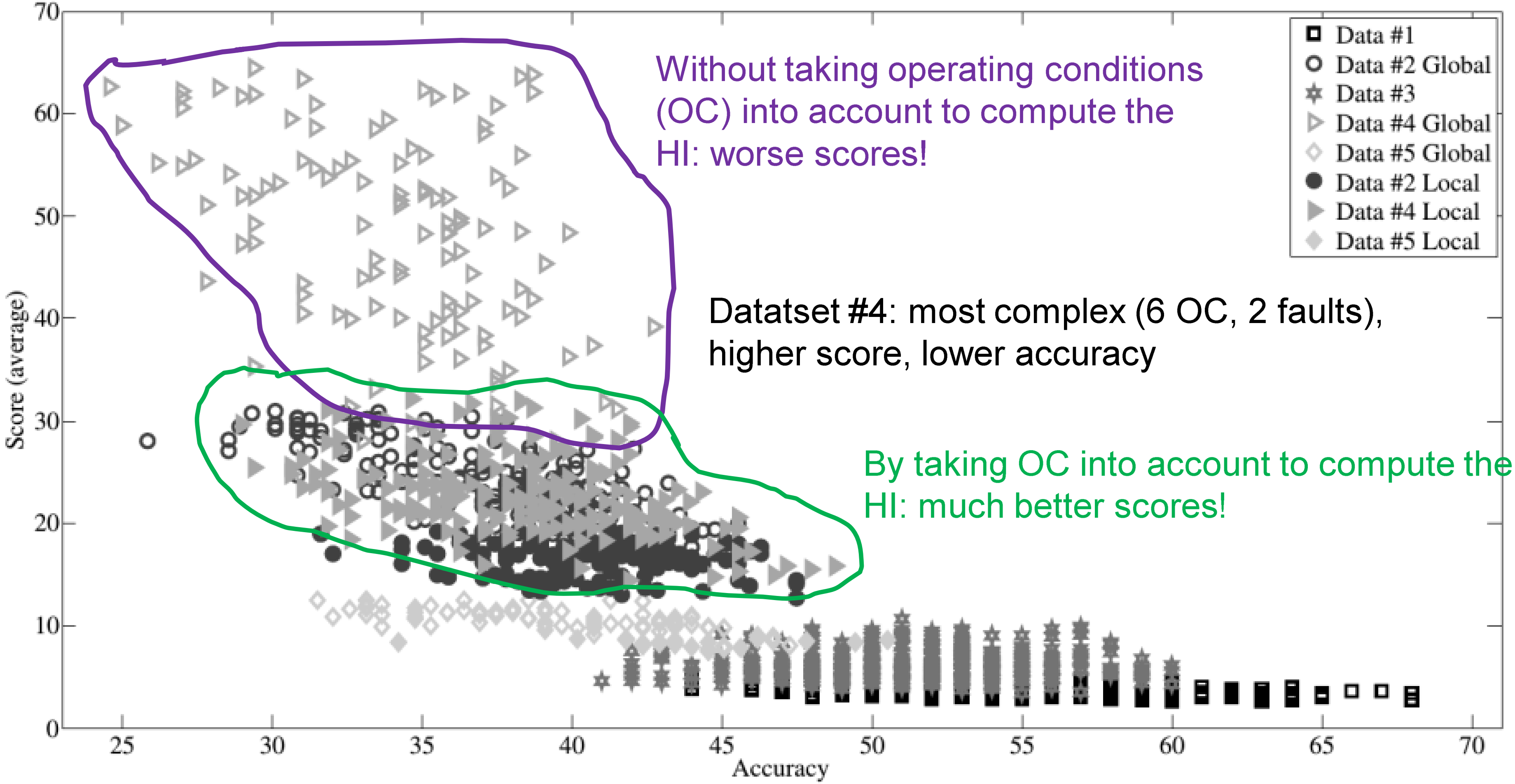
RULCLIPPER Algorithm
Results



Datatset #4: most complex (6 OC, 2 faults)
⇨ Bad (high) score, bad (low) accuracy

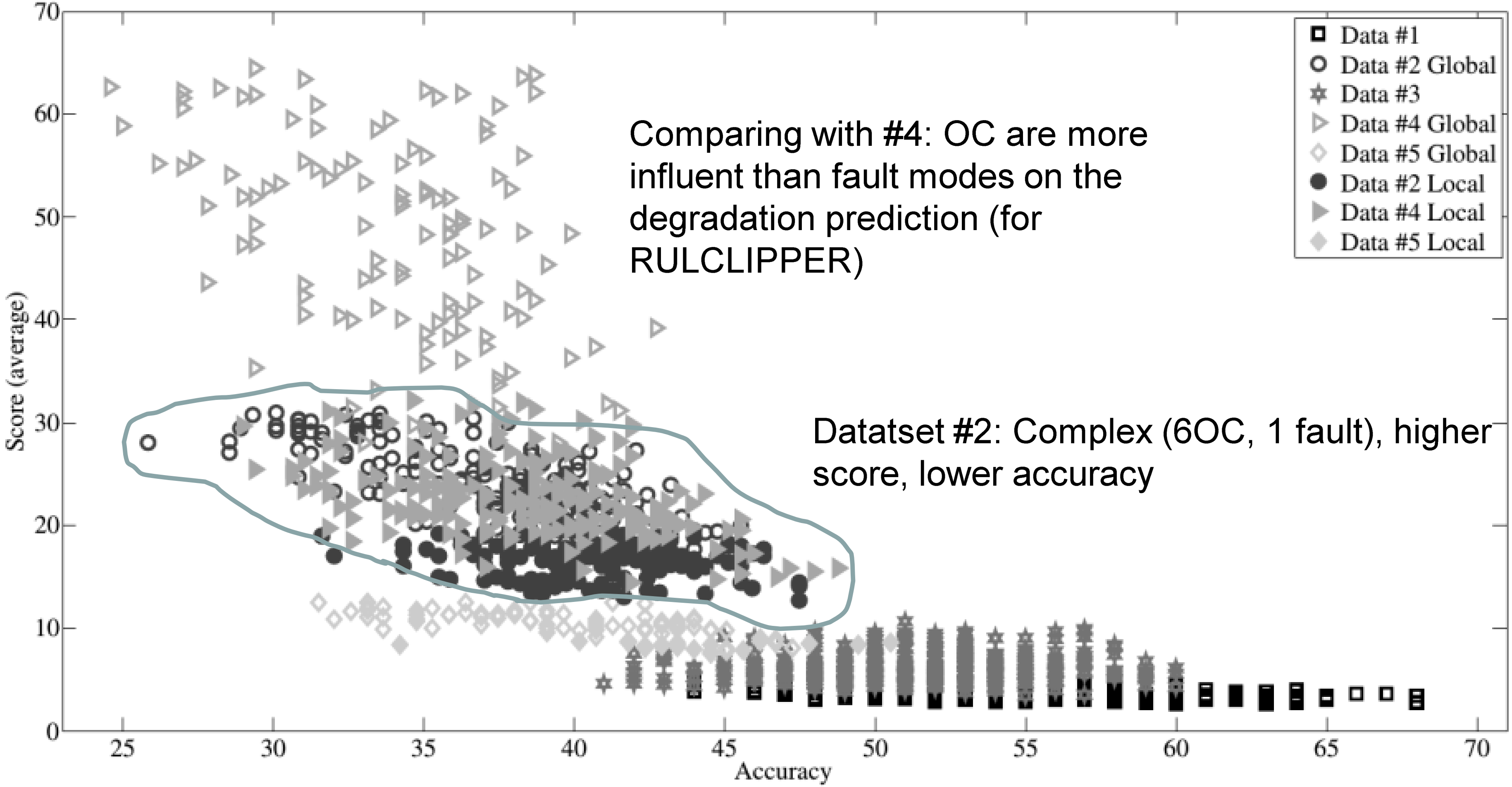
RULCLIPPER Algorithm

Results



RULCLIPPER Algorithm

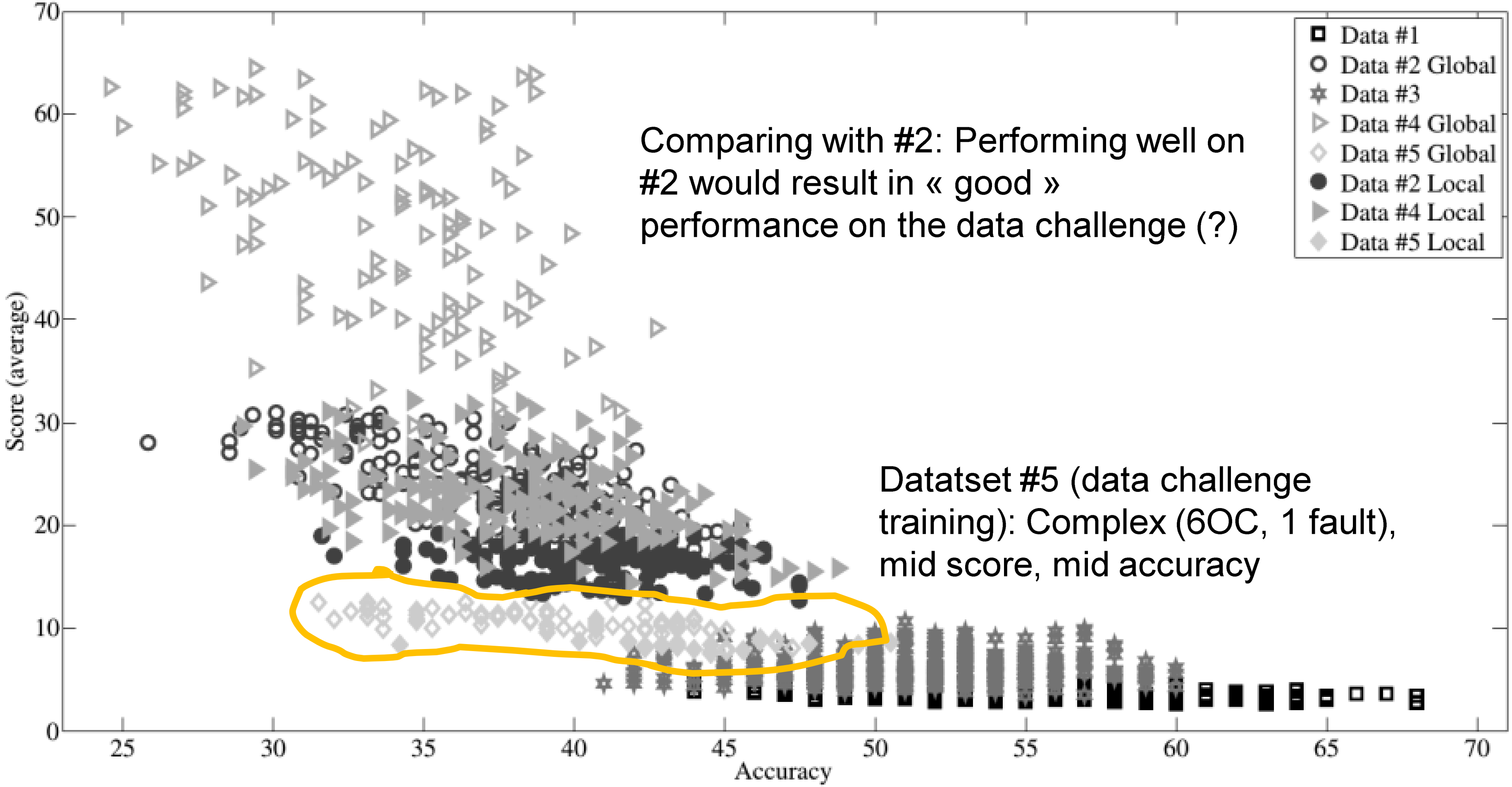
Results



Comparing with #4: OC are more influent than fault modes on the degradation prediction (for RULCLIPPER)

Datatset #2: Complex (6OC, 1 fault), higher score, lower accuracy

RULCLIPPER Algorithm
Results

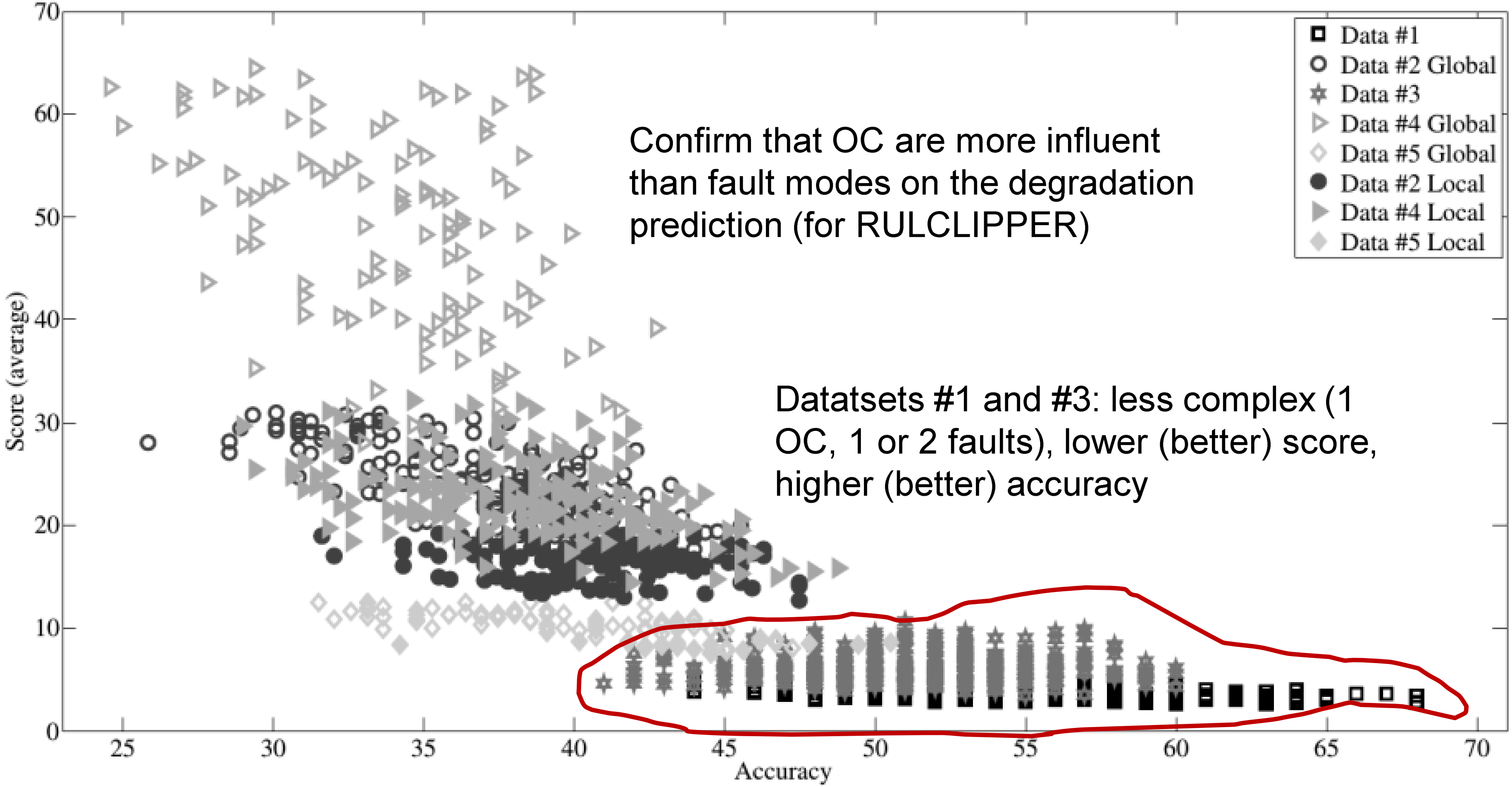


Comparing with #2: Performing well on #2 would result in « good » performance on the data challenge (?)

Datatset #5 (data challenge training): Complex (6OC, 1 fault), mid score, mid accuracy

RULCLIPPER Algorithm

Results

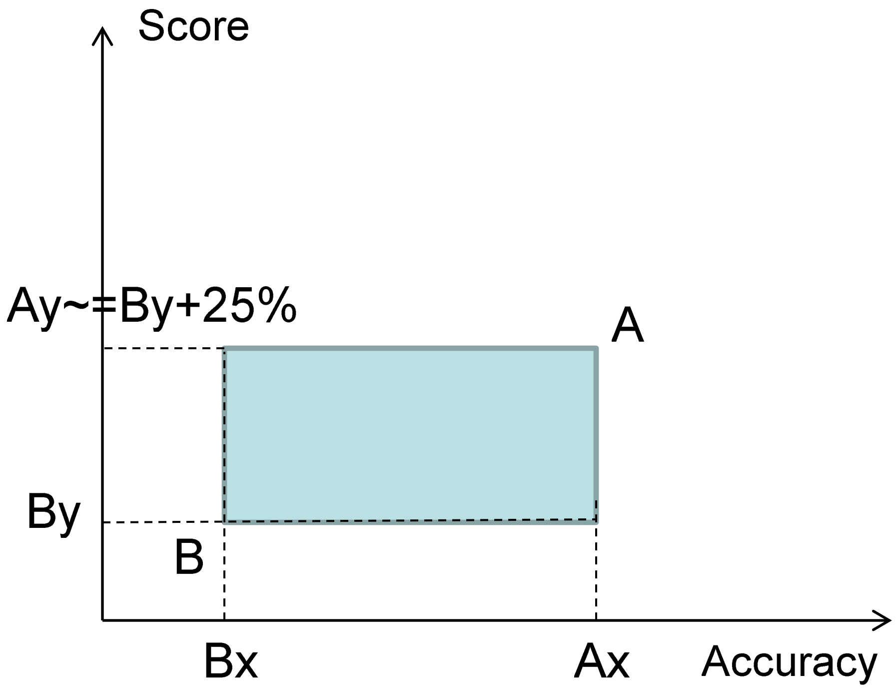


Confirm that OC are more influent than fault modes on the degradation prediction (for RULCLIPPER)

Datatsets #1 and #3: less complex (1 OC, 1 or 2 faults), lower (better) score, higher (better) accuracy

RULCLIPPER Algorithm

Sensitivity of input sensors for HI estimation



For the ~500 **combinations of sensors**, about ~6000 parameterisations of standard combination rules of RULs were tested

For **each input sensor subset** (among ~500) used to compute the HI

B=(Bx,By): select the combination rule yielding **the lowest score** (By), Bx is the corresponding accuracy

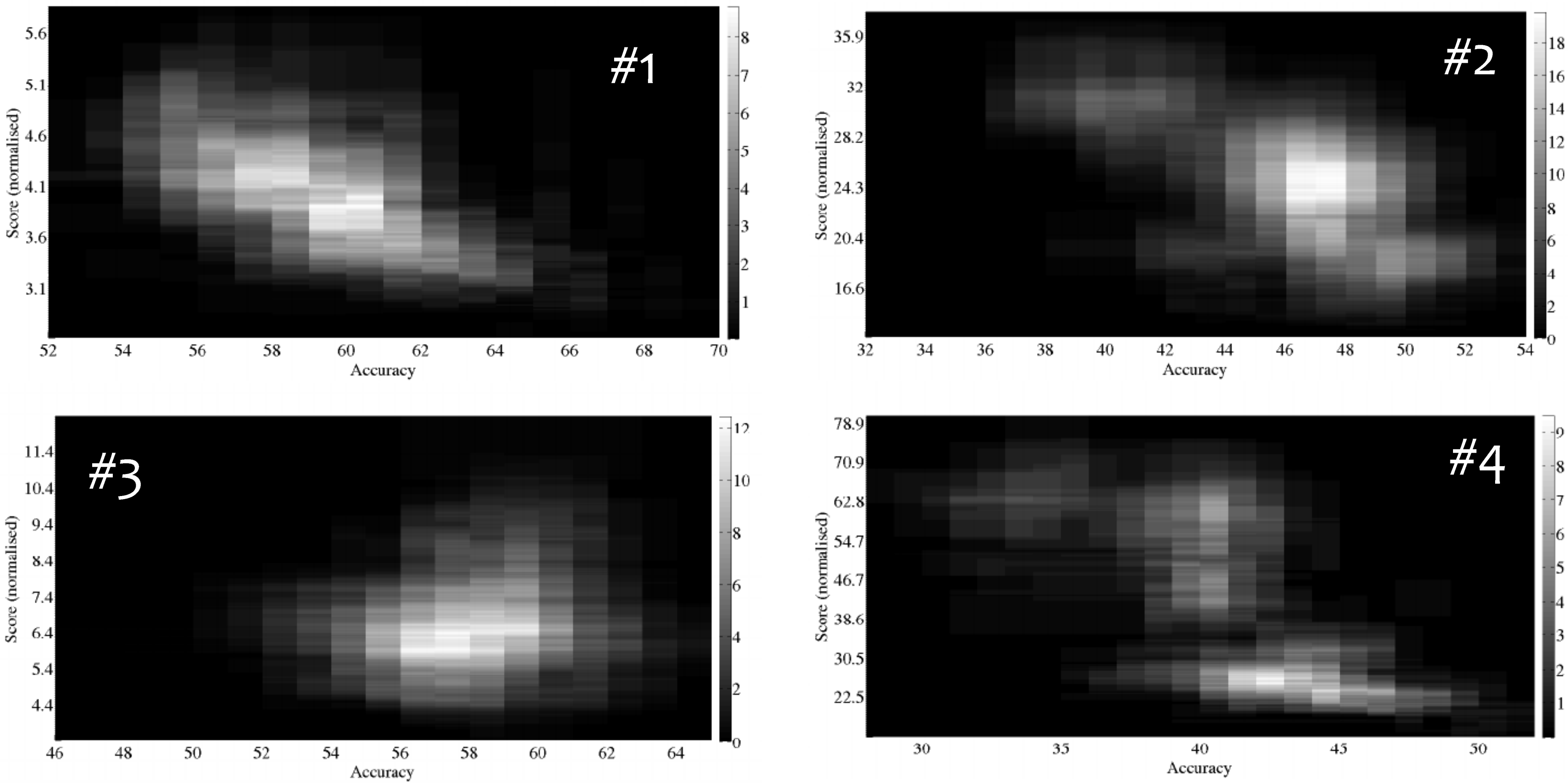
A=(Ax,Ay): select **another combination rule** with the **best accuracy Ay** for which the score falls within the interval [By ; **Ay~=By+25%**]

Then A and B defines a **rectangle** in the score-accuracy plane

RULCLIPPER Algorithm

Sensitivity of input sensors for HI estimation

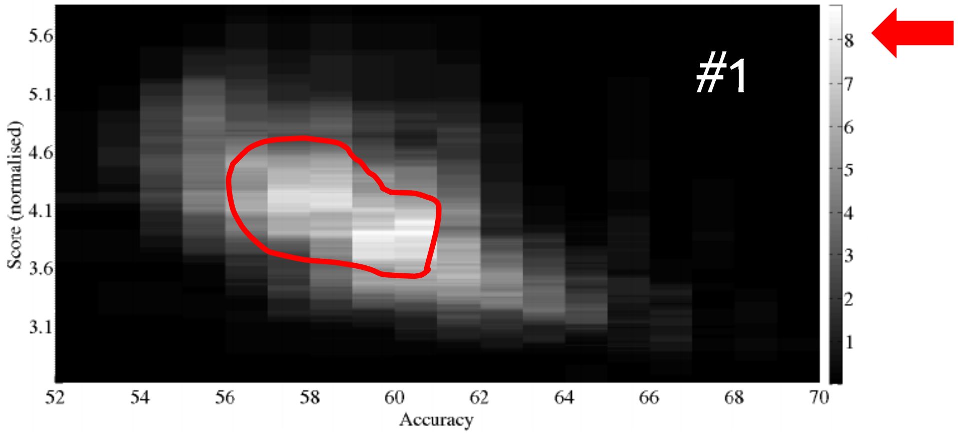
- For the ~500 combinations, estimate and cumulate all rectangles



RULCLIPPER Algorithm

Sensitivity of input sensors for HI estimation

❖ The selection of the right subset of sensors is critical if the dataset is complex (number of OC, number of fault modes)



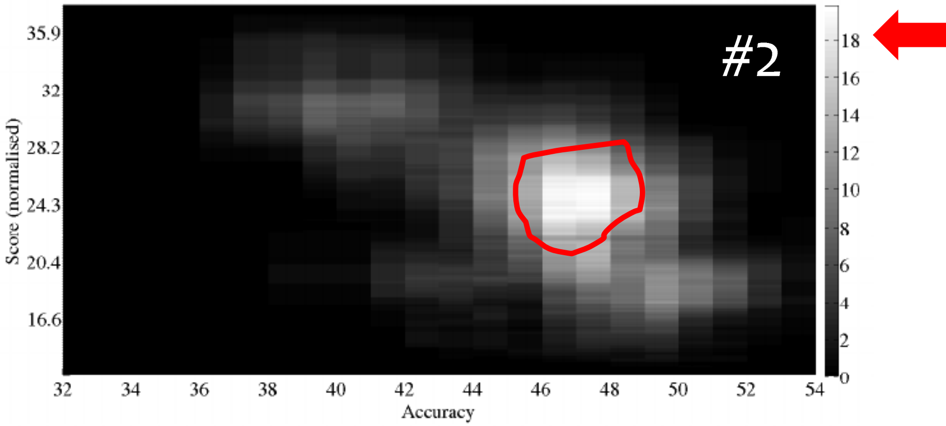
#1: The accumulation of rectangles gives form to a « cube »

- Most of subsets of sensors yield good results
- Easy to parameterise the algorithms

- The selection of the right subset of sensors is critical if the dataset is complex (number of OC, number of fault modes)

#2: Operating conditions yield two « clouds », one with a high peak:

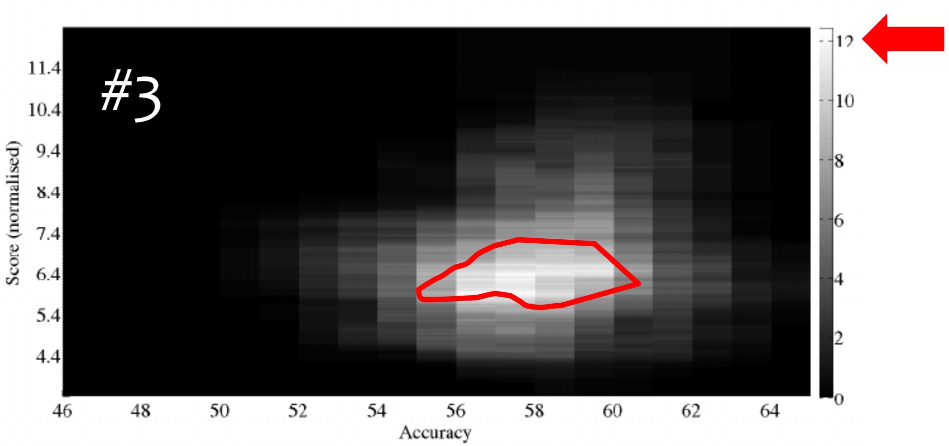
- The spread means that the input sensors become critical to compute the HI
- The peaks is the position of the best parameterisations



RULCLIPPER Algorithm

Sensitivity of input sensors for HI estimation

- The selection of the right subset of sensors is critical if the dataset is complex (number of OC, number of fault modes)



#3: The second fault mode do not affect too much the results:

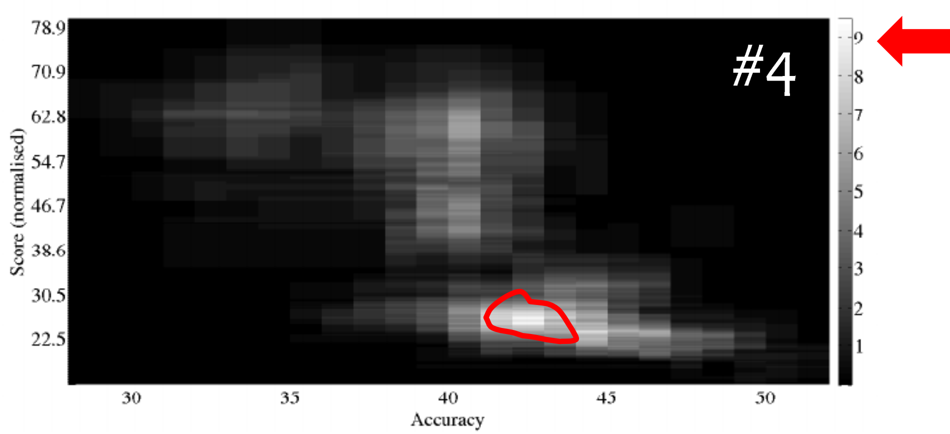
- translation of the « cloud » to the upper left hand-side compared to #1
- The peak is higher meaning that the subset is more critical than #1

RULCLIPPER Algorithm

Sensitivity of input sensors for HI estimation

- The selection of the right subset of sensors is critical if the dataset is complex (number of OC, number of fault modes)

#4: For the most difficult dataset, four « clouds » appear: The combination of fault modes and operating conditions greatly increases the sensitivity of sensor subsets on the performances



RULCLIPPER Algorithm

Comparison with the state-of-the-art – All datasets

LabeX Action
Integrated smart systems

Scores obtained in 2008

Algo. (pseudo.) / Data	$\#5_T$	$\#5_V$
heracles (1)	737 (3rd)	5636 (1st)
FOH (2)	512 (2nd)	6691 (2nd)
LP (3)	n.a.	25921
sunbea	436.8 (1st)	54437 (22nd)
bobosir	1263	8637
L6	1051	9530
GoNavy	1075	10571
beck1903	1049	14275
Sentient	809	19148
A	975	20471
mjhutk	2430	30861
RelRes	1966	35863
phmnrc	2399	35953
SuperSiegel	1139	154999

After 2008

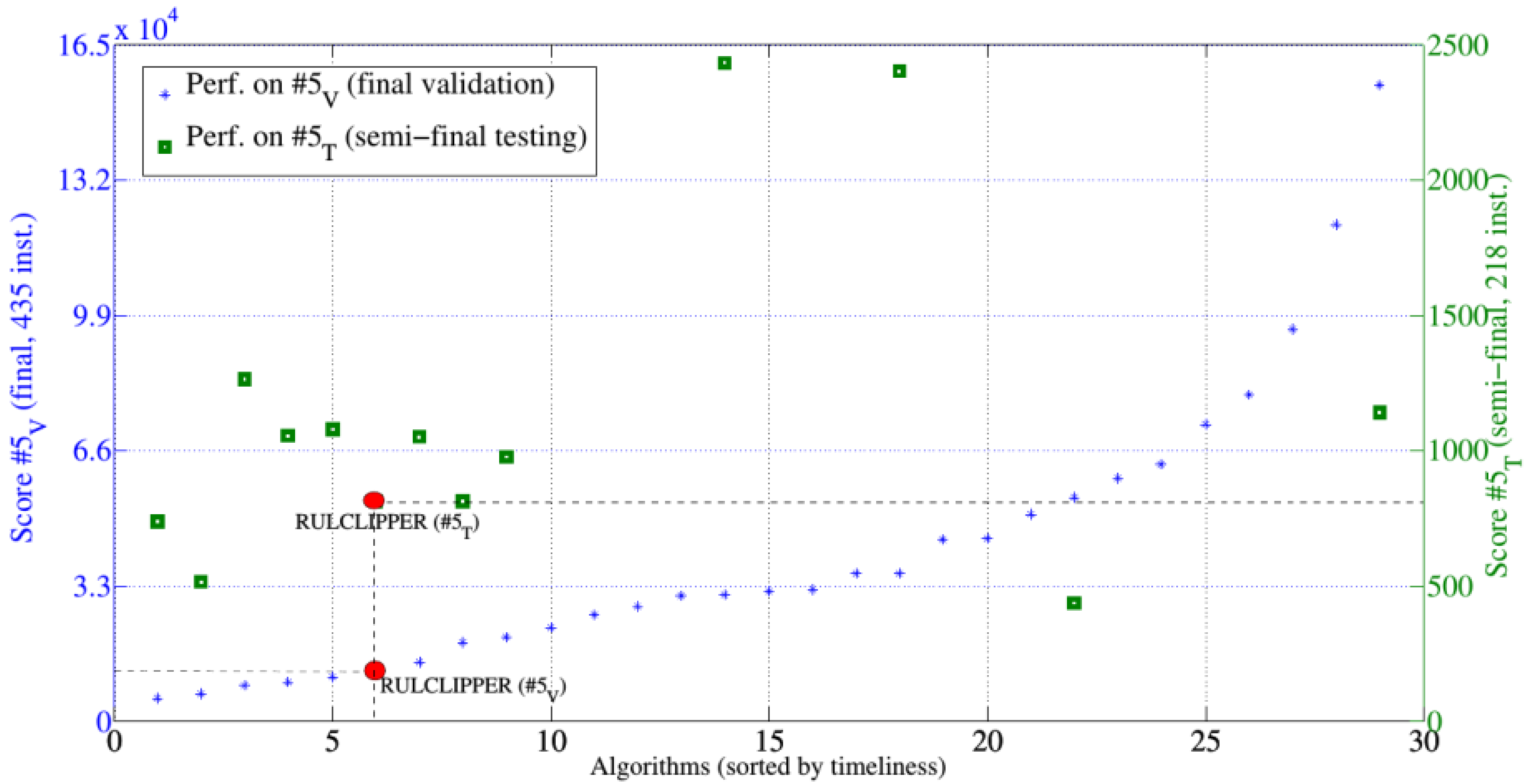
Algo. (pseudo.)	$\#5_T$	$\#5_V$
RULCLIPPER	**752**	**11572**
SBL (P. Wang, Youn, & Hu, 2012)	1139	n.a.
DW (Hu, Youn, Wang, & Yoon, 2012)	1334	n.a.
OW (Hu et al., 2012)	1349	n.a.
MLP (Riad, Elminir, & Elattar, 2010)	1540	n.a.
AW (Hu et al., 2012)	1863	n.a.
SVM-SBI (Hu et al., 2012)	2047	n.a.
RVM-SBI (Hu et al., 2012)	2230	n.a.
EXP-SBI (Hu et al., 2012)	2282	n.a.
GPM3 (Coble, 2010)	2500	n.a.
RNN (Hu et al., 2012)	4390	n.a.
REG2 (Riad et al., 2010)	6877	n.a.
GPM2B (Coble, 2010)	19200	n.a.
GPM2 (Coble, 2010)	20600	n.a.
GPM1 (Coble, 2010)	22500	n.a.
QUAD (Hu et al., 2012)	53846	n.a.

RULCLIPPER performance						
Dataset	#1	#2	#3	#4	$\#5_T$	$\#5_V$
Score	216	2796	317	3132	752	11672
Accuracy (%)	67	46	59	45	n.a.	n.a.
FPR (%)	56	51	66	49	n.a.	n.a.
FNR (%)	44	49	34	51	n.a.	n.a.
MAPE (%)	20	32	23	34	n.a.	n.a.
MAE	10	17	12	18	n.a.	n.a.
MSE	176	524	256	592	n.a.	n.a.

Better results have been obtained when considering performance measures individually (e.g. 70% on #1 vs 53% in previous work), but here only the **scoring** function was considered for ranking

RULCLIPPER Algorithm

Comparison with the state-of-the-art – Data challenge



Conclusions

LabeX Action
Integrated smart systems

- The RULCLIPPER algorithm is proposed to estimate the **RUL of engineering systems** where the health indicators are imprecise
  - RULCLIPPER is made of elements inspired from the **computational geometry** community and relies on the adaptation of case based reasoning to manage **imprecision** in training and testing **instances**.
  - It is an original and efficient approach for RUL estimation, **validated** and **compared** to past work using **the six datasets** coming from the **turbofan engine** simulator (C-MAPSS), including the so-called
    - ✓ turbofan datasets (four datasets)
    - ✓ and the data challenge (two datasets)

    These datasets are considered as complex due to the presence **of fault modes and operating conditions.**
- In addition to RULCLIPPER
  - A method was proposed to **estimate the health indicator**
  - The **selection of the most relevant sensors** was tackled.
  - **Information fusion rules / ensembles** were studied to combine RULs

Perspectives

- As for all similarity-based matching algorithms (T. Wang, 2010; E. Ramasso et al. 2012, 2013), the computational cost associated to sort instances is the most important limitation, but:
  - In tests, an i7-vPro 8-cores CPU with tasks parallelisation was used leading low computational costs: Loading and formatting data (from NASA) + HI estimation took ~2 sec.; Intersection with an IHI of length 220 units (among longest ones) took only ~3.8 **m**sec!
  - A procedure can be used to make polygons convex (approximations), that could drastically reduces computational time
  - Computational geometry has become a very active field in particular to improve memory and time requirements, with applications in multimedia (computer graphics such as games) for which **CUDA implementations** on processor arrays (using graphic cards) were proposed. With such implementations, **real-time** and **anytime** prognostics can be performed.
- The extension of RULCLIPPER to **multiple health indices** is also under study, in particular by using **polytopes**.

www.labex-action.fr

Thanks for your attention

Thanks to the organising committee

This work has been carried out in the Laboratory of Excellence ACTION through the program "Investments for the future" managed by the National Agency for Research (reference ANR-11-LABX-01-01).

Contact: emmanuel.ramasso@femto-st.fr
http://members.femto-st.fr/emmanuel-ramasso/

References

- D. Simon 2012: Challenges in Aircraft Engine Gas Path Health Management, 2012 Turbo Expo
- S. Garg and D. Simon, Challenges in Aircraft Engine Control and Gas Path Health Management, 2012 Turbo Expo
- Frederick, D., DeCastro, J., & Litt, J. (2007). User's guide for the commercial modular aero-propulsion system simulation (C-MAPSS) (Tech. Rep.). Cleveland, Ohio 44135, USA: National Aeronautics and Space Administration (NASA), Glenn Research Center.
- Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008a). Damage propagation modeling for aircraft engine run-to-failure simulation. In Prognostics and health management, 2008. phm 2008. international conference on (pp. 1–9)

- See also references included in *[Investigating computational geometry for failure prognostics]* to appear in *[PHM-Europe 2014 conference]* and *[International Journal on Prognostics and Health Management, 2014]*