UNIVERSITÀ DEGLI STUDI DI BERGAMO
Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Control Automation Lab

ADAPTIVE LEARNING, ESTIMATION AND SUPERVISION OF DYNAMICAL SYSTEMS (ALES)

Case study 3: Leak detection in an industrial valve

Master Degree in COMPUTER ENGINEERING

Data Science and Data Engineering Curriculum

SPEAKER
Ing. Pietro Boni

PLACE
University of Bergamo

Syllabus

1. **Recursive and adaptive identification**
   1.1 Recursive ARX estimation (RLS)
   1.2 Least Mean Squares (LMS)
   1.3 Instrumental Variables (IV)

2. **Subspace and MIMO identification**
   2.1 Singular Value Decomposition
   2.2 Impulse data: Ho-Kalman, Kung algorithms
   2.3 Generic I/O data: the MOESP algorithm

3. **Closed-loop identification**

4. **Supervision of dynamical systems**
   4.1 Introduction to fault diagnosis
   4.2 Model-based fault diagnosis
   4.3 Parity space approaches
   4.4 Observer-based approaches
   4.5 Signal-based fault diagnosis
   4.6 Knowledge-based fault diagnosis

CASE STUDIES

- Virtual Reference Feedback Tuning
- Nuclear particles classification
- Leak detection in an industrial valve
- Bearing fault identification

Outline

1. Problem statement and Experimental setup
2. Recursive system identification - RLS
3. Leak/NO Leak Classification – Idea and Algorithm
4. Leak /NO Leak Classification – Logistic Regression
5. Leak /NO Leak Classification – Anomaly Detection
6. Leak /NO Leak Classification – New Prototype
7. Conclusion

Outline

1. Problem statement and Experimental setup

2. Recursive system identification - RLS

3. Leak/NO Leak Classification – Idea and Algorithm

4. Leak /NO Leak Classification – Logistic Regression

5. Leak /NO Leak Classification – Anomaly Detection

6. Leak /NO Leak Classification – New Prototype

7. Conclusion

Problem statement and experimental setup

Leak Detection is used to determine **if**, and in some cases **where**, a leak has occured in a system which contains liquids and/or gases (pipeline).

The primary purpose of **Leak Detection Systems (LDS)** is to help pipeline controllers to **detect** and, in certain cases, **localize** leaks. LDS provide alarms and display other related data to the pipeline controllers to assist decision-making.

Literature (both based on model-based approaches and data-driven approaches) is focused on leak detection for gas/oil transportation, and leak detection in urban water networks **where sensors could be placed optimally and in a large number**.

Problem statement and experimental setup

More specifically, in literature it is possible to find methods that:

- create a fluid-dynamics model of the pipeline, or
- use fuzzy systems, or
- use different statistical classifiers, or
- use transient wave reflection theory.

It is important to note that all these different methods have in common the idea of **dividing the pipeline in segments and place sensors in almost all of these segments**.

In this way, methods have different «**views**» of the pipeline and could understand its behaviour and perform **leak detection and localization** (with the latter needing pipeline segmentation).

Problem statement and experimental setup

However, it **is not** always possible divide the pipeline and place sensors optimally along the pipeline.

Considering **small industry or home** plumbing systems, it is very likely that sensors can **only** be placed on the **water supply connection valve**. Moreover, in order to reduce costs, the number of sensors is generally small.

In these contexts, **it is not possible** to define a mathematical model of the pipeline or use a data-driven method from the literature.

Moreover, the **impossibility** to perform a pipeline segmentation **make leak localization not possible**.

Problem statement and experimental setup

Case study objective: create a specific method that **only** use data from the sensors placed on the plumbing system valve in order to **detect a leak**.

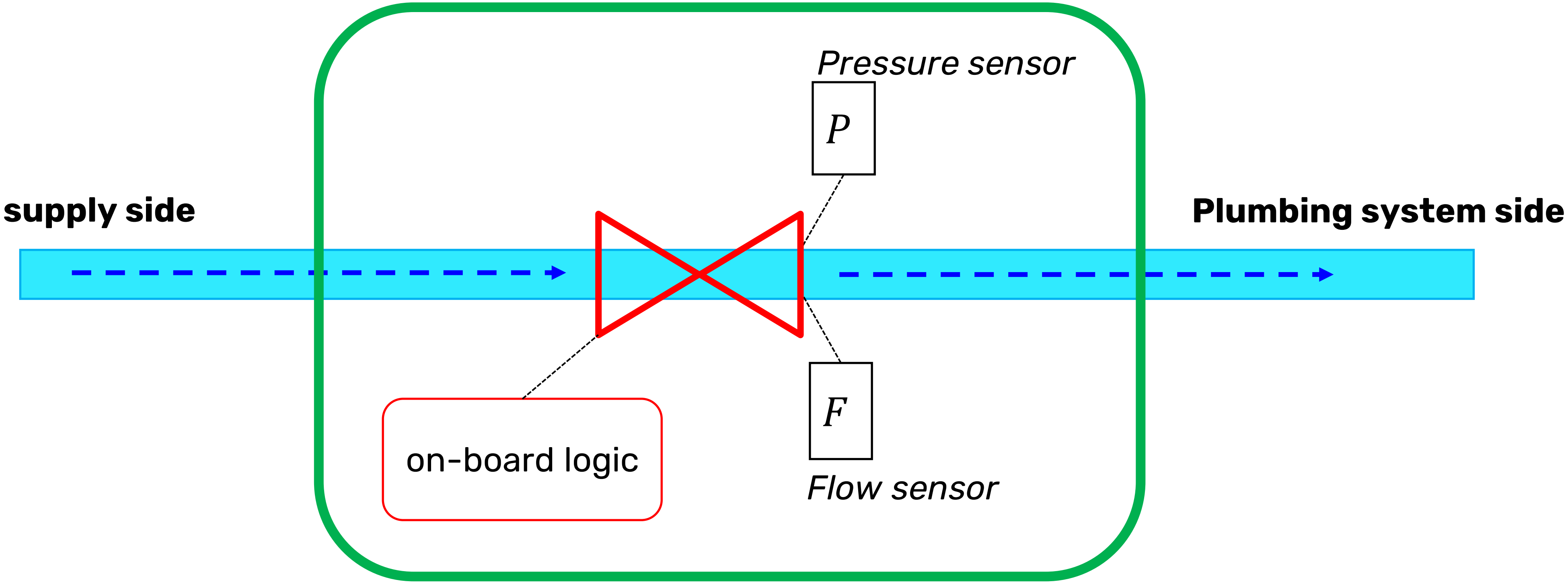
The detection algorithm will be implemented on the on-board logic of the valve.

Additional requirements

- The method must be usable on **different plumbing systems.**
  - A «**training**» session will have to be performed when the valve is installed.
  - Training step that needs **only leak-free data** is preferred.
- The method must be able to detect a leak even if the **plumbing system is in use** (e.g. the tap is turned on).

Problem statement and experimental setup

It is possible to schematise the system valve as follows:

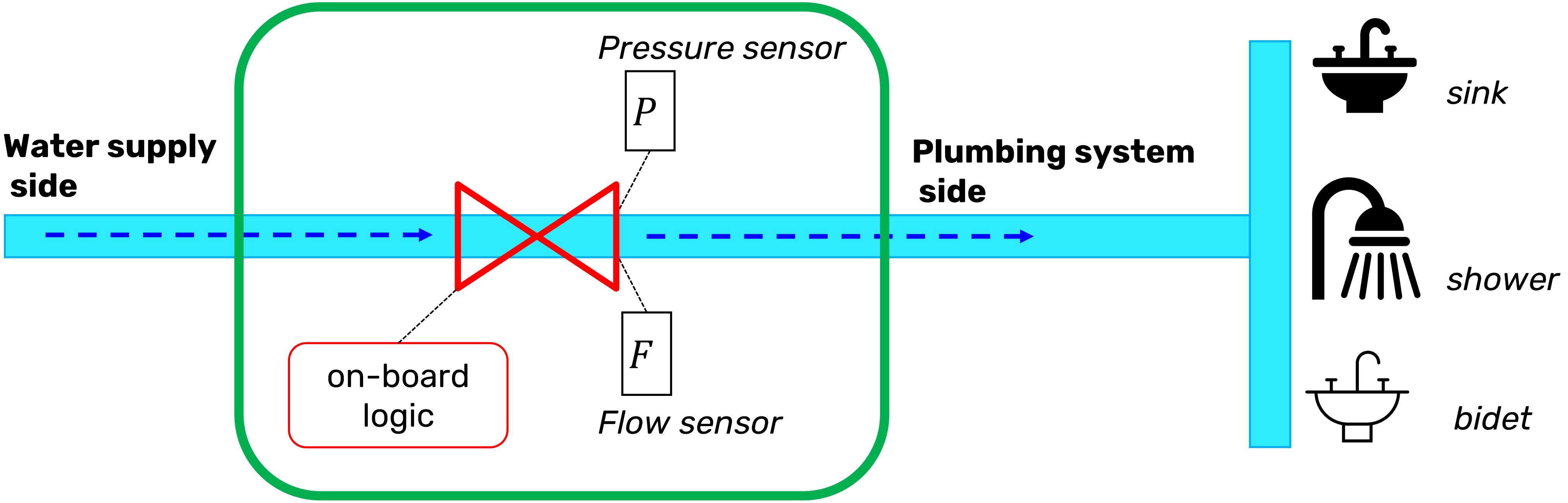


on-board logic: gathers data and runs leak detection algorithm

Smart-Valve

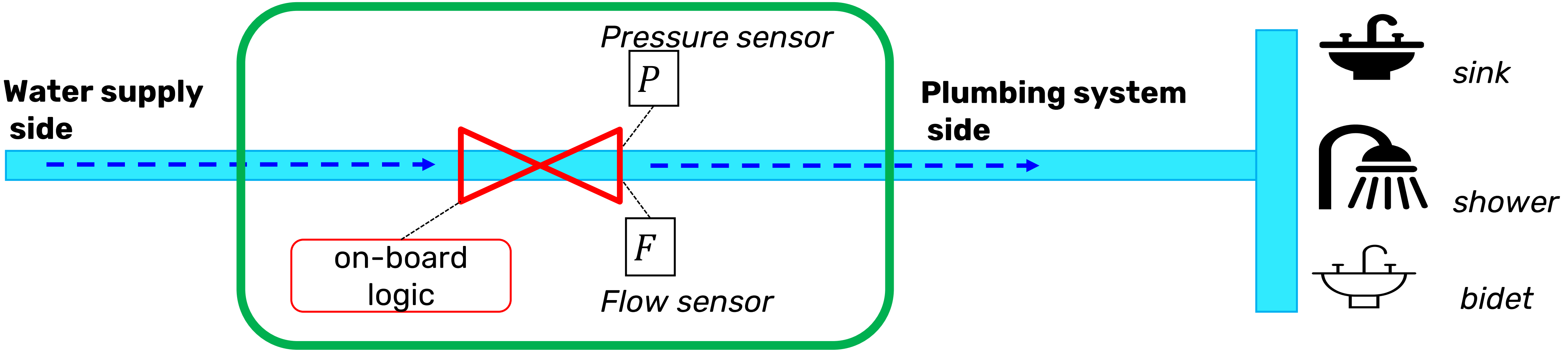
Problem statement and experimental setup

In order to simulate a **home use** of the plumbing system, **the experimental setup** also includes a sink, a bidet and a shower.



Practical note: To gather sensors data, in the development phase, a computer is connected to the on-board logic.

Problem statement and experimental setup



Practical consideration: a utility can be «open» or «closed», so the system can be in two different condition: «utility open» or «utility closed». The leak «exists» or «doesn't exist».

		UTILITY	
		OPEN	CLOSED
LEAK	EXISTS	1	2
	DOESN'T EXIST	3	4

The system can be in only one condition at time.

Problem statement and experimental setup

		UTILITY	
		OPEN	CLOSED
LEAK	EXISTS	1	2
	DOESN’T EXIST	3	4

Practical consideration: a «**depressurisation test**» to detect a leak **when utilities are closed** is already implemented and **it works well**.

So, when utilities are closed, it is also possible to use this test to perform a double check on the condition of the system.

Outline

1. Problem statement and Experimental setup
2. **Recursive system identification - RLS**
3. Leak/NO Leak Classification – Idea and Algorithm
4. Leak /NO Leak Classification – Logistic Regression
5. Leak /NO Leak Classification – Anomaly Detection
6. Leak /NO Leak Classification – New Prototype
7. Conclusion

Recursive system identification - RLS

Since a mathematical model cannot be defined for this specific case study, it is possible to approach the problem with a **recursive black-box system identification method**.

More specifically, the **Adaptive Recursive Least Square** method has been chosen:

1. A model structure has been chosen so that there is a **fixed number of parameters** to estimate.

$$y(t) = \theta_1 y(t-2) + \theta_2 u(t-1) + e(t)$$

- $u(t) = P(t)$
- $y(t) = F(t)$

$considering\ N\ samples$:

$$\boldsymbol{y} \in \mathbb{R}^N$$

2. The parameters estimation is **initialised to zero**:

$$\widehat{\boldsymbol{\theta}}^0(t) = \left[\hat{\theta}_1 \quad \hat{\theta}_2\right]^T = [0 \quad 0]^T$$

3. When new data is measured, the estimation $\widehat{\boldsymbol{\theta}}(t)$ **is updated to fit with new data**.

Recursive system identification - RLS

The **adaptive RLS form 3** is used in this case study:

Adaptive RLS form 3 summary

- $\mu = 0.99$
- $d = 2$

- Basic recursion: $$\underset{d\times 1}{\hat{\boldsymbol{\theta}}(t)} = \hat{\boldsymbol{\theta}}(t-1) + K(t)\cdot\varepsilon(t)$$
- Definitions: $$\underset{d\times 1}{K(t)} = \underset{d\times d}{P(t)}\underset{d\times 1}{\boldsymbol{\varphi}(t)} \qquad \underset{1\times 1}{\varepsilon(t)} = y(t) - \boldsymbol{\varphi}(t)^\top\hat{\boldsymbol{\theta}}(t-1)$$
- Auxiliary recursion: $$\underset{d\times d}{P(t)} = \frac{1}{\mu}\left[P(t-1) - \underset{1\times 1}{\beta(t-1)^{-1}}\cdot P(t-1)\boldsymbol{\varphi}(t)\boldsymbol{\varphi}(t)^\top P(t-1)\right]$$

$$\underset{1\times 1}{\beta(t-1)} = \mu + \underset{1\times d}{\boldsymbol{\varphi}(t)^\top} P(t-1)\underset{d\times 1}{\boldsymbol{\varphi}(t)}$$

- $\mu \to 1$: **low** tracking **speed**, **high precision** (small estimation variance)

Outline

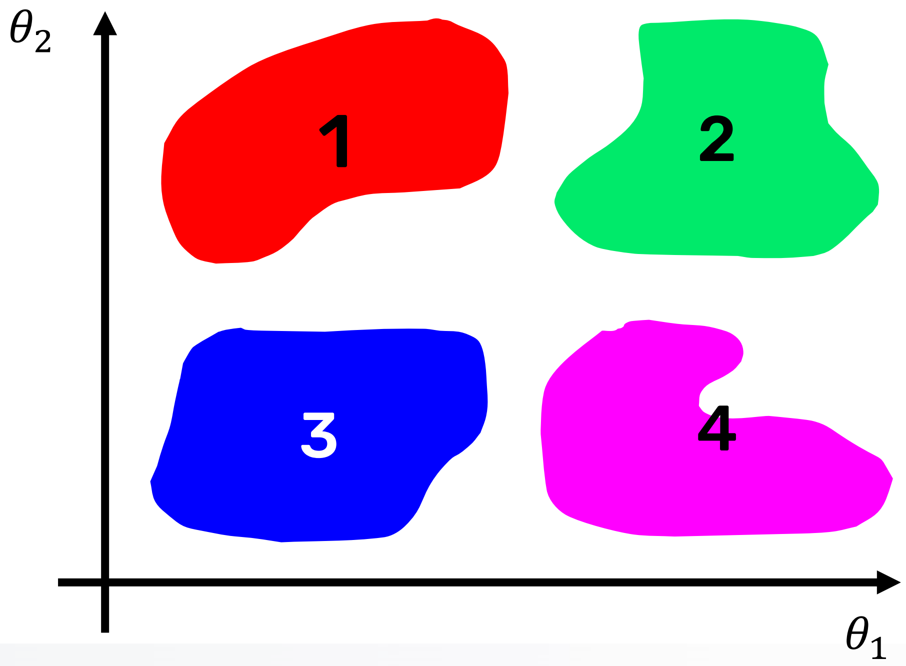
1. Problem statement and Experimental setup
2. Recursive system identification - RLS
3. **Leak/NO Leak Classification – Idea and Algorithm**
4. Leak /NO Leak Classification – Logistic Regression
5. Leak /NO Leak Classification – Anomaly Detection
6. Leak /NO Leak Classification – New Prototype
7. Conclusion

Leak/NO Leak Classification – Idea and Algorithm

Since **the system can be in only one condition at time**, some questions arise.

- *Does each condition have a* ***«special» parameters configuration*** *that characterise it?*
- *Do parameters values lie in* ***separated regions*** *of the plane* $\hat{\theta}_1(t), \hat{\theta}_2(t)$ *?*

		UTILITY	
		CLOSED	OPEN
LEAK	DOESN'T EXISTS	1	2
	EXIST	3	4



Leak/NO Leak Classification – Idea and Algorithm

		UTILITY	
		CLOSED	OPEN
LEAK	DOESN'T EXIST	1	2
	EXIST	3	4

The answer to these questions is yes, BUT

after an analysis on $\hat{\theta}_1(t)$ and $\hat{\theta}_2(t)$ it emerged that:

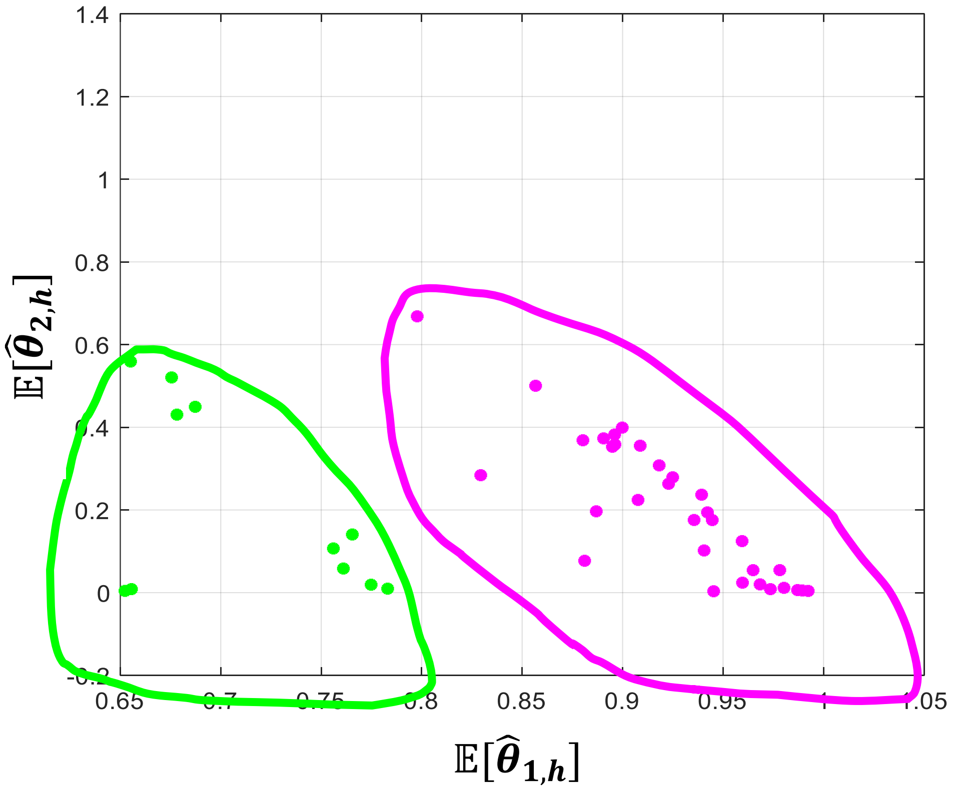
- Cases **1** and **3** can be classified with a simple check on the flow value.
  - When utilities **are closed** the flow must be equal to zero $F = 0$. **If** $\boldsymbol{F \neq 0 \;\Rightarrow leak}$
- For cases **2** and **4** (**utilities open**) the plane to consider **is not** $\hat{\theta}_1(t), \hat{\theta}_2(t)$ **but** $\mathbb{E}[\hat{\theta}_{1,h}], \mathbb{E}[\hat{\theta}_{2,h}]$ where $\boldsymbol{\hat{\theta}_{1,h}, \hat{\theta}_{2,h}} \in \mathbb{R}^{2\times N_w}$ are vectors that contain N_w estimates $\hat{\theta}_1(t)$ and $\hat{\theta}_2(t)$, respectively.
  - the attention is not on the values of the single estimate $\boldsymbol{\hat{\theta}}(t)$ but on the «**history**» of the estimates, because **the difference between the two conditions (2 and 4) only occurs when considering the transient opening of utilities**.

N_w is a parameter of the method that has to be tuned.

Leak/NO Leak Classification – Idea and Algorithm

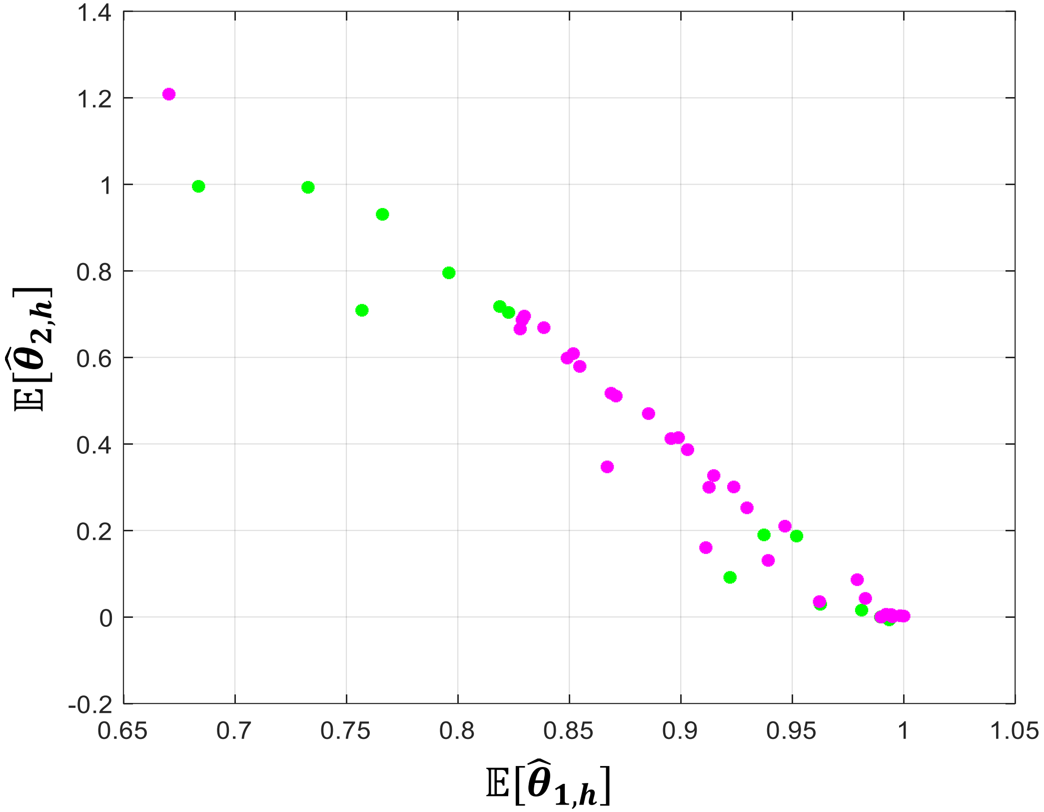
The difference between the two conditions (2 and 4) only occurs **when considering the transient opening of utilities**.

considering the transient



Two separated plane regions

Not considering the transient



Regions are not separated

But why $\mathbb{E}[\widehat{\boldsymbol{\theta}}_{1,h}], \mathbb{E}[\widehat{\boldsymbol{\theta}}_{2,h}]$?

Leak/NO Leak Classification – Idea and Algorithm

If we consider $\hat{\theta}_1(t), \hat{\theta}_2(t)$ **instead of** $\mathbb{E}[\hat{\theta}_{1,h}], \mathbb{E}[\hat{\theta}_{2,h}]$ it is impossible to consider the opening transient because it takes several instants of time, not only one t.

So, it is necessary first to store different estimates $\hat{\theta}_1(t), \hat{\theta}_2(t)$, and then to calculate the means of the stored estimates.

In this way, one obtains a point in the plane defined by $\mathbb{E}[\hat{\theta}_{1,h}], \mathbb{E}[\hat{\theta}_{2,h}]$ that characterises the system condition.

This is an OFFLINE approach, because it needs a batch of N_w data to calculate $\mathbb{E}[\hat{\boldsymbol{\theta}}_{1,h}], \mathbb{E}[\hat{\boldsymbol{\theta}}_{2,h}]$

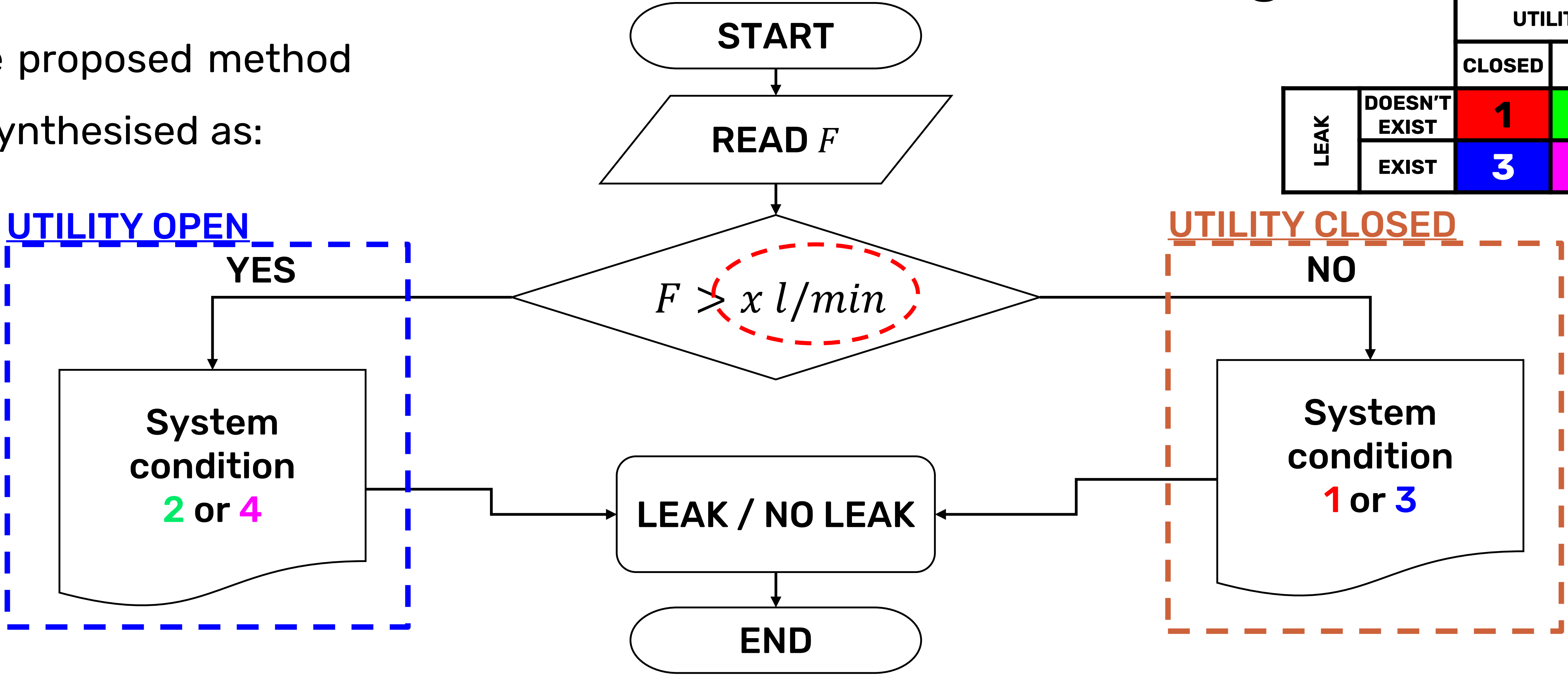
N_w is a parameter of the method that has to be tuned.
It has to be such that $\hat{\boldsymbol{\theta}}_{1,h}, \hat{\boldsymbol{\theta}}_{2,h}$ contains the opening transient data.

Leak/NO Leak Classification – Idea and Algorithm

The proposed method

is synthesised as:

		UTILITY	
		CLOSED	OPEN
LEAK	DOESN'T EXIST	1	2
	EXIST	3	4



The threshold x is chosen with respect to a characterisation of the plumbing system.
It is the minimum flow that can come out of the utilities.

Leak/NO Leak Classification – Idea and Algorithm

UTILITY OPEN

System condition
2 or 4

RLS

$\mathbb{E}[\widehat{\boldsymbol{\theta}}_{1,h}], \mathbb{E}[\widehat{\boldsymbol{\theta}}_{2,h}]$
(OFFLINE approach)

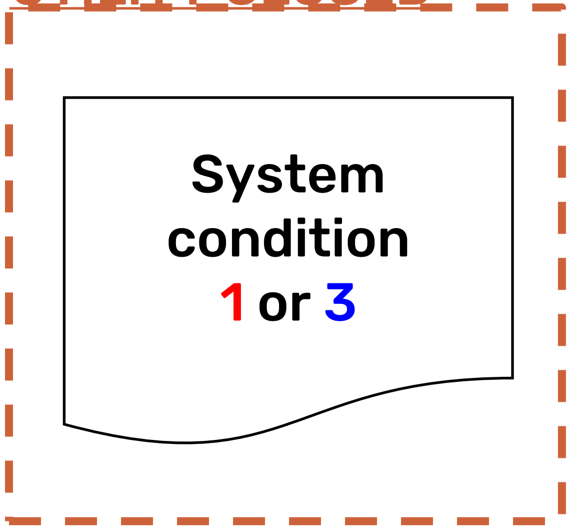
LEAK / NO LEAK

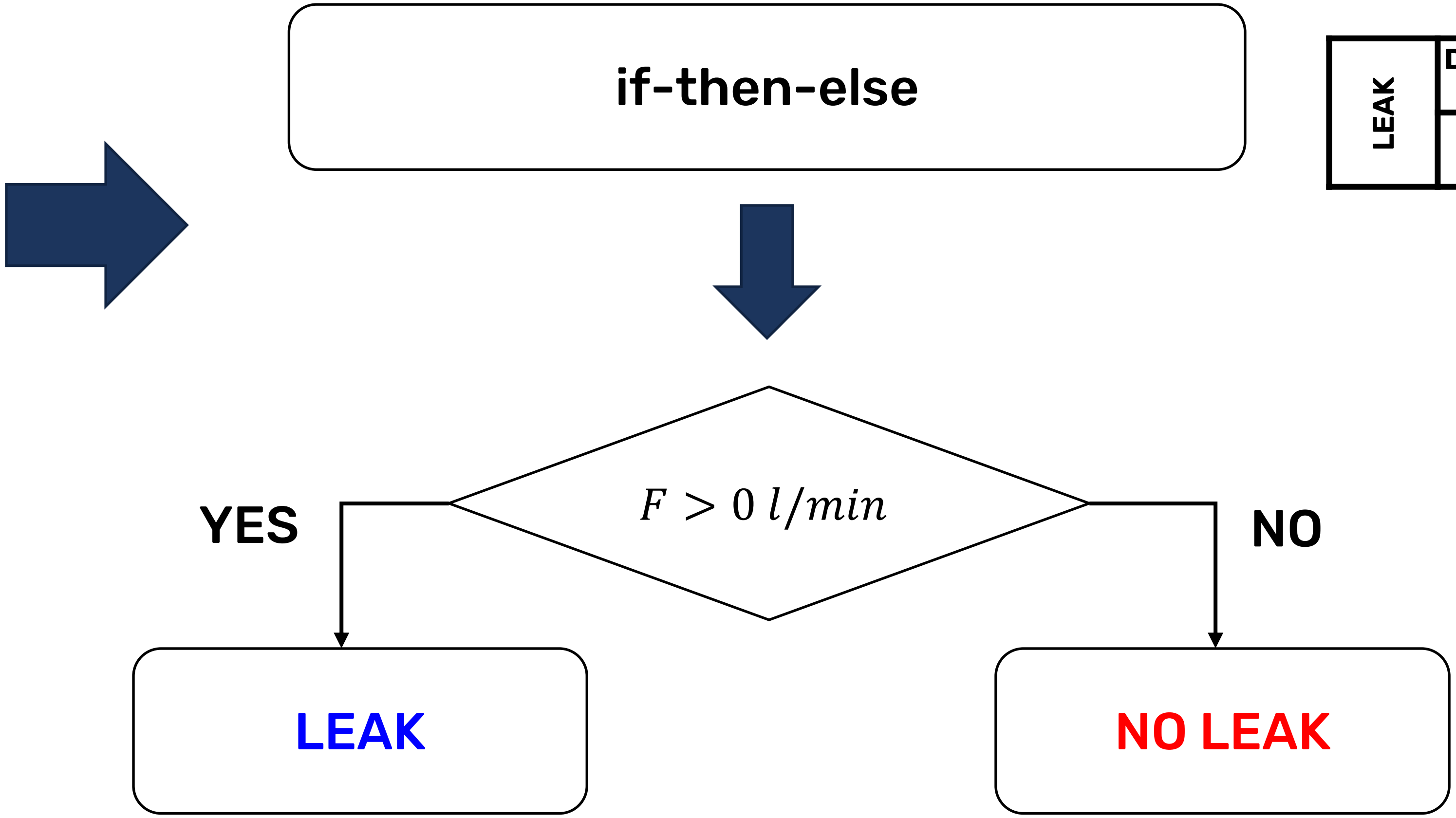
		UTILITY	
		CLOSED	OPEN
LEAK	DOESN'T EXIST	1	2
	EXIST	3	4

Leak/NO Leak Classification – Idea and Algorithm

		UTILITY	
		CLOSED	OPEN
LEAK	DOESN'T EXIST	1	2
	EXIST	3	4

UTILITY CLOSED





Outline

1. Problem statement and Experimental setup
2. Recursive system identification - RLS
3. Leak/NO Leak Classification – Idea and Algorithm
4. **Leak /NO Leak Classification – Logistic Regression**
5. Leak /NO Leak Classification – Anomaly Detection
6. Leak /NO Leak Classification – New Prototype
7. Conclusion

Leak/NO Leak Classification – Logistic Regression

		UTILITY	
		CLOSED	OPEN
LEAK	DOESN'T EXIST	1	2
	EXIST	3	4

Since cases **1** and **3** are treated with an if-then-else, this section (as the following ones) is focused only on cases **2** and **4** .

Classification step is done with the **logistic regression method**.

- **Generic steps** :
  1. (**TRAIN**) train a logistic regression model on known labeled leak/no leak data.
  2. (**VALIDATION**) Validate the trained model with known labeled data (different from training data)

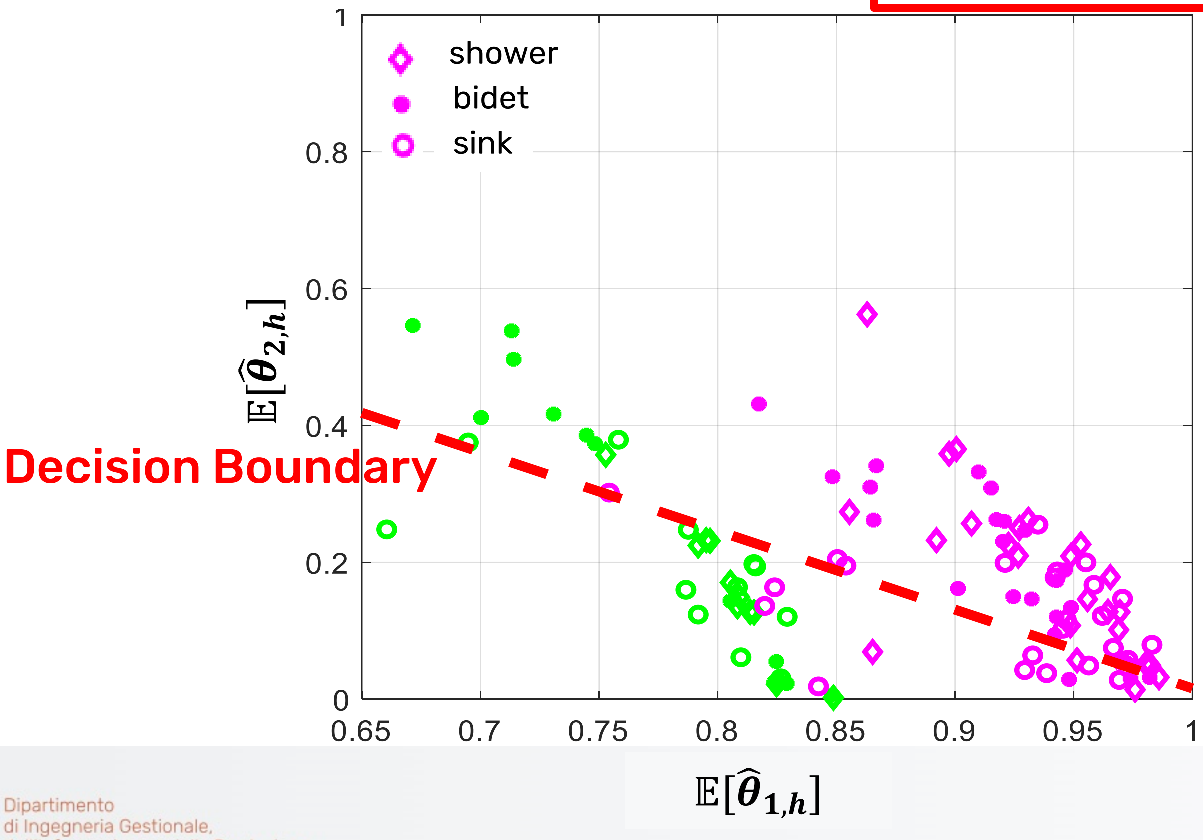
Train and Validation step need leak/no leak data, so it is necessary to be able to insert a test leak on the system.

The model will be used to classify new data

Leak/NO Leak Classification – Logistic Regression

Practical consideration: train a model using data from **only one utility** is «dangerous» because in this way only a very little portion of the system is consider and so certain behaviours can be omitted, compromising model accuracy.

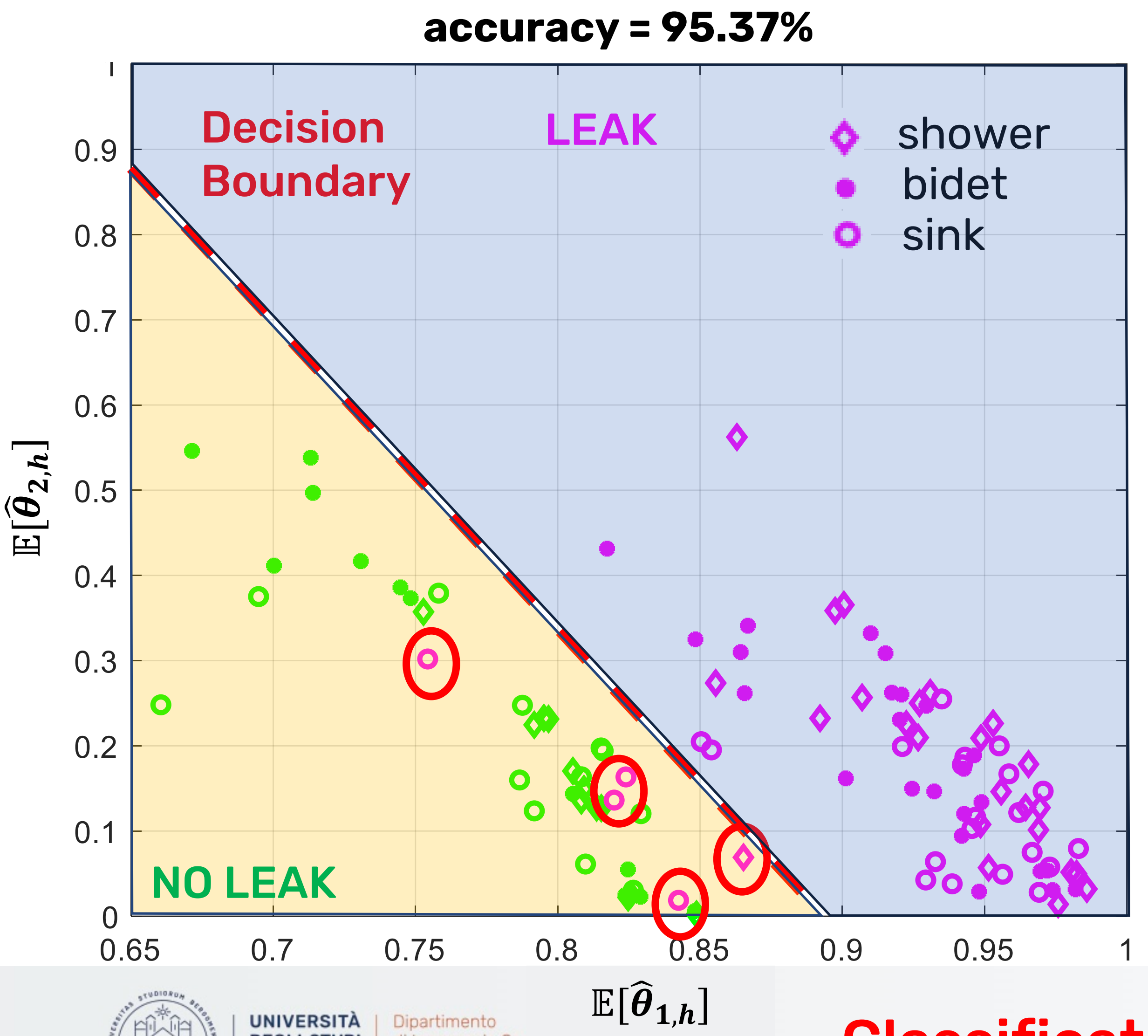
Training data only from sink – accuracy = 76.85%



$$accuracy = \frac{\#correct}{\#total} * 100$$

Leak/NO Leak Classification – Logistic Regression

Obtained results



- **Training set:**
  - Open utility at 100%.

For all utility:

- 3 test no-leak
- 3 test 1.2 l/min leak
- 3 test 0.6 l/min leak

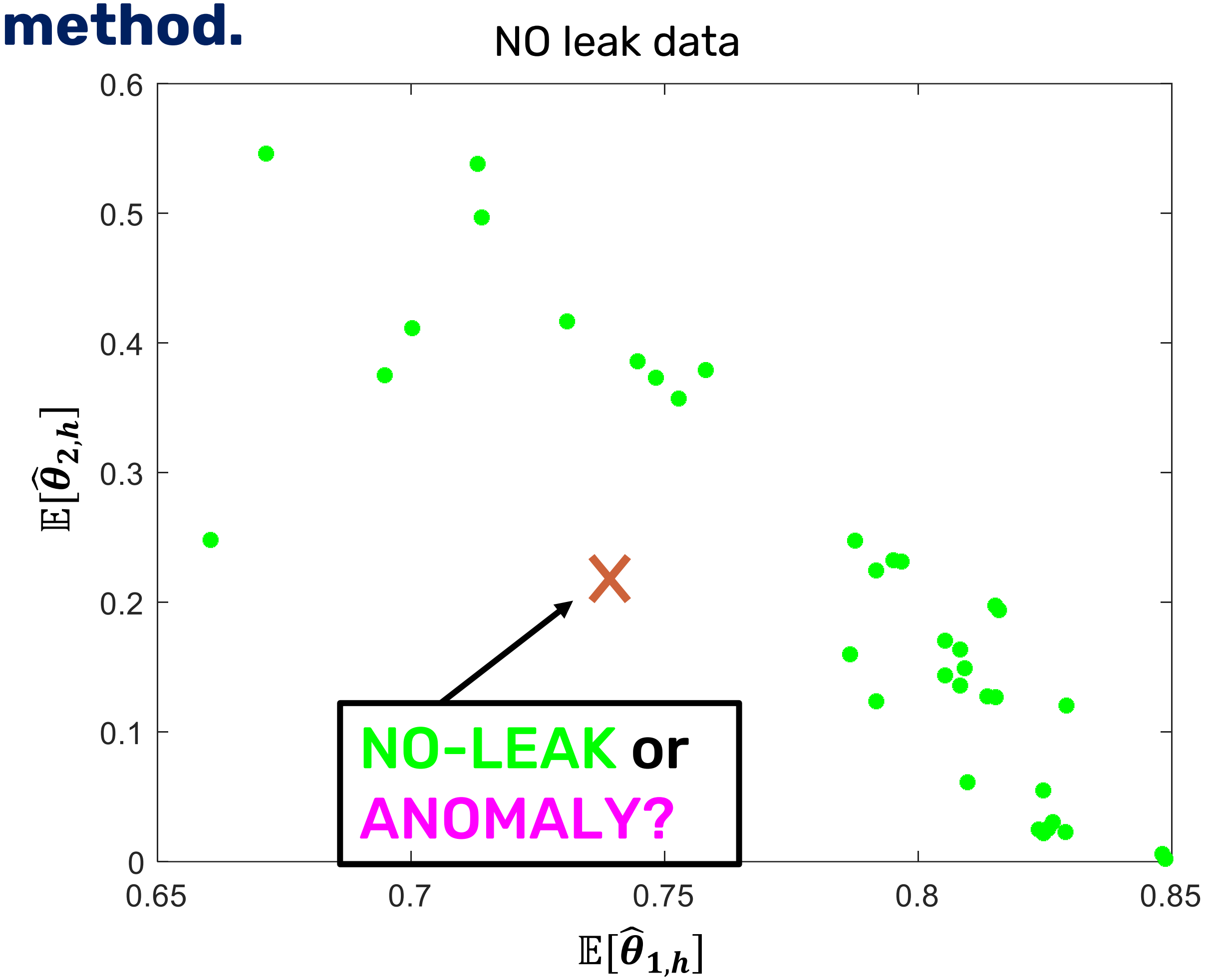
- $accuracy = 95.37\%$

Classification errors

Outline

Leak/NO Leak Classification – Anomaly Detection

An alternative approach for leak/no leak classification is an anomaly detection method.



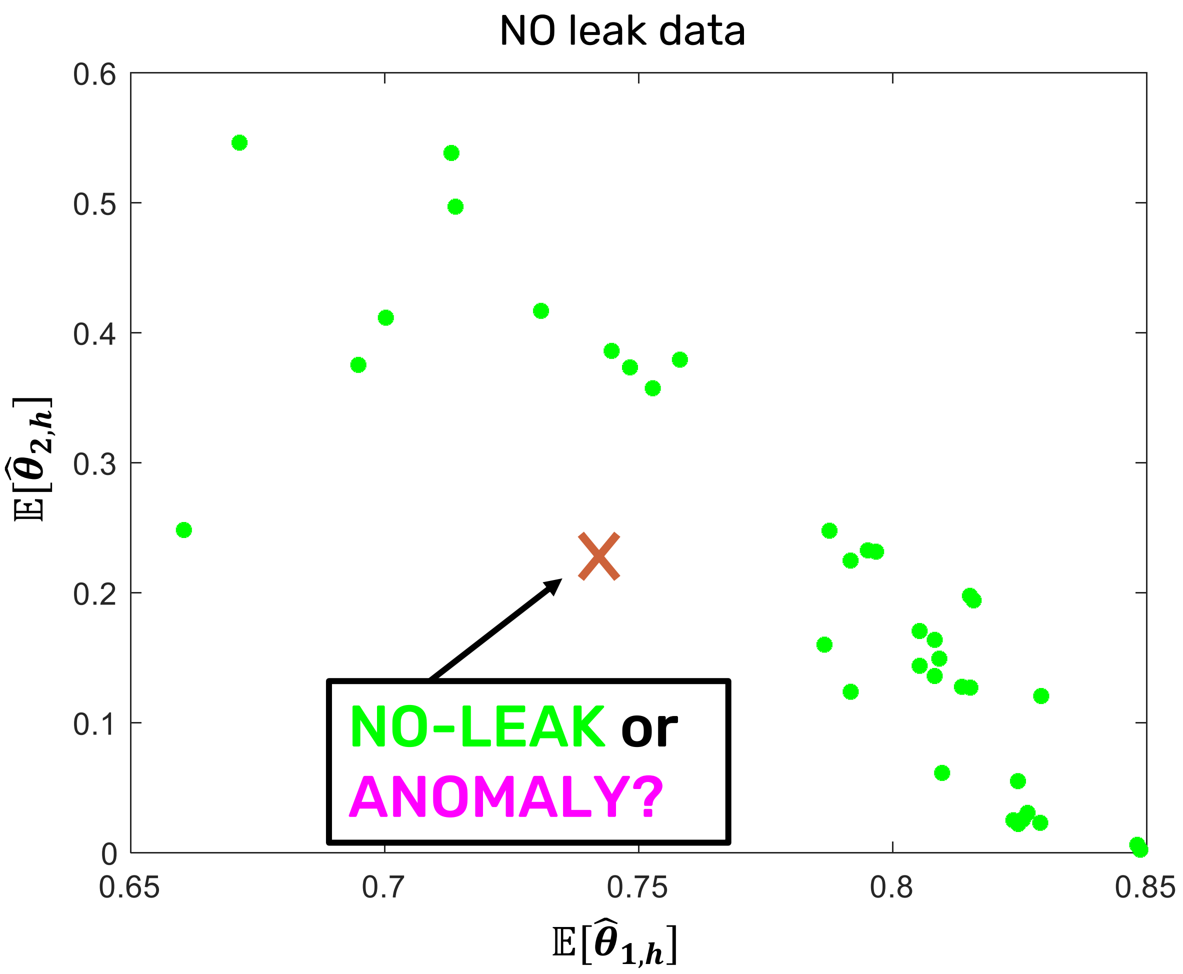
Considering NO leak data, the plane $\mathbb{E}[\hat{\theta}_{1,h}], \mathbb{E}[\hat{\theta}_{2,h}]$ in generated.

Consider now a new data and plot the corresponding point ✕ in the plot.

✕ $= (\mathbb{E}[\hat{\boldsymbol{\theta}}_{1,h}], \mathbb{E}[\hat{\boldsymbol{\theta}}_{2,h}])$

It is not known if the new data is leak/NO leak (anomaly).

Leak/NO Leak Classification – Anomaly Detection



How can one tell if the point ✕ is an anomaly?

One can **fit a probability distribution on a set of training data** (e.g. NO leak data)

The new point ✕ has associated a probability $p($✕$)$ that indicates the **probability to belong to the trained probability distribution.**

Setting a **probability threshold** ϵ:

$$if\ p(\times) < \varepsilon \Rightarrow ANOMALY$$

Leak/NO Leak Classification – Anomaly Detection

The probability threshold ϵ represents the probability level **below which** the point $\times$ **is considered outside to the trained probability distribution**.

Typically, ϵ is a very small value of probability.

So, if $p(\times) < \epsilon$ one can say that $\times$ has a very small probability to belong to the trained distribution.

To train a probability distribution it is necessary to make assumptions on the data distribution.

In this case study, it is assumed that data are **normally distributed**.

Leak/NO Leak Classification – Anomaly Detection

Trained probability distribution -> Multivariate Normal Distribution

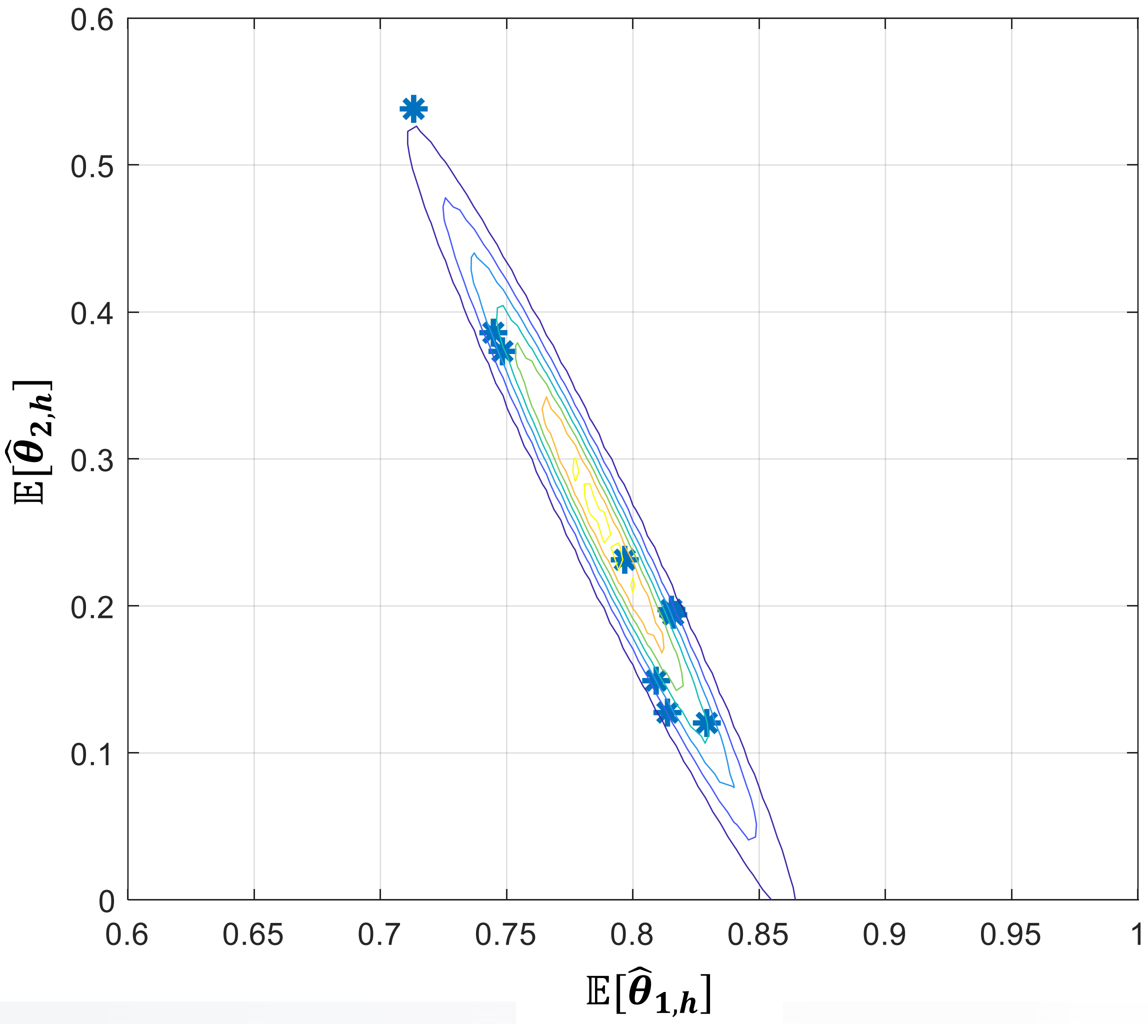
$$\boldsymbol{X} = (\mathbb{E}[\widehat{\boldsymbol{\theta}}_{\mathbf{1},\boldsymbol{h}}], \mathbb{E}[\widehat{\boldsymbol{\theta}}_{\mathbf{2},\boldsymbol{h}}])$$

The two features $\mathbb{E}[\widehat{\boldsymbol{\theta}}_{\mathbf{1},\boldsymbol{h}}], \mathbb{E}[\widehat{\boldsymbol{\theta}}_{\mathbf{2},\boldsymbol{h}}]$ **are not modelled separately** but a unique distrubution is fitted. So:

$$p(\mathbb{E}[\widehat{\boldsymbol{\theta}}_{\mathbf{1},\boldsymbol{h}}]\mathbb{E}[\widehat{\boldsymbol{\theta}}_{\mathbf{2},\boldsymbol{h}}], \boldsymbol{\mu}, \Sigma)$$

With

$$\boldsymbol{\mu} \in \mathbb{R}^2, \Sigma \in \mathbb{R}^{2\times 2}$$



Leak/NO Leak Classification – Anomaly Detection

It is possible to synthesise the anomaly detection method as follows.

1. Gather no-leak data and get the corresponding point in the plane $\mathbb{E}[\widehat{\boldsymbol{\theta}}_{\mathbf{1},\boldsymbol{h}}], \mathbb{E}[\widehat{\boldsymbol{\theta}}_{\mathbf{2},\boldsymbol{h}}]$.
2. Fit a multivariate normal distribution on the gathered points
3. Choose a probability threshold ϵ
4. When a new data comes, derive the corresponding point ✕ on the plane $\mathbb{E}[\widehat{\boldsymbol{\theta}}_{\mathbf{1},\boldsymbol{h}}], \mathbb{E}[\widehat{\boldsymbol{\theta}}_{\mathbf{2},\boldsymbol{h}}]$
5. Compute $p($ ✕ $)$ and observe $if\ p($ ✕ $) < \varepsilon \Rightarrow ANOMALY$

Practical consideration: **training set consists of only no-leak data**. It is not necessary to insert a leak in the plumbing system to gather training data.

Leak/NO Leak Classification – Anomaly Detection

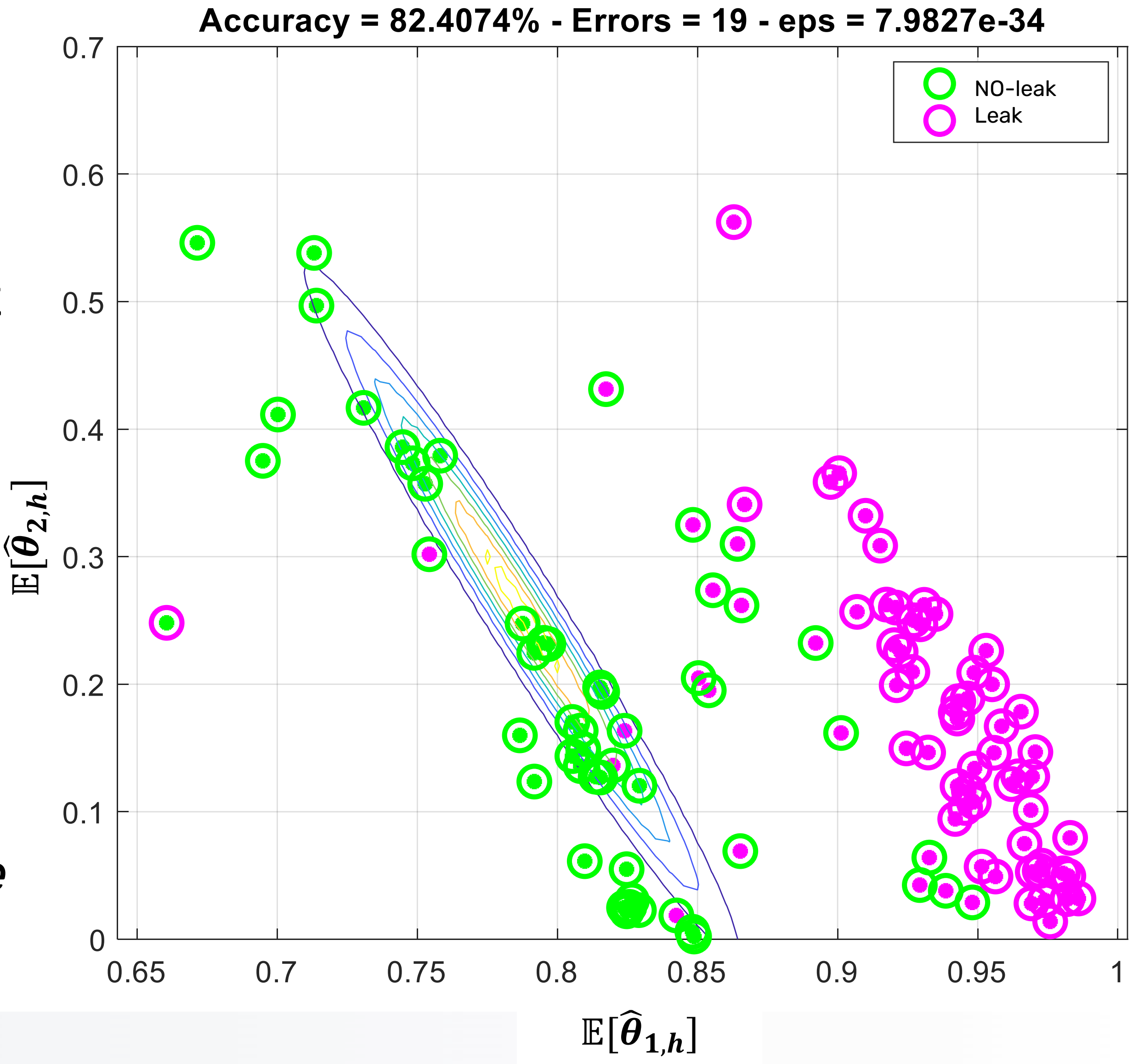
Obtained results:

Training set:

- 9 points (**no-leak**, utilities open at 100%)

$$accuracy = 82.40\%$$

Only few contour lines are plotted in figure. The green and magenta dots are the true labels.



Outline

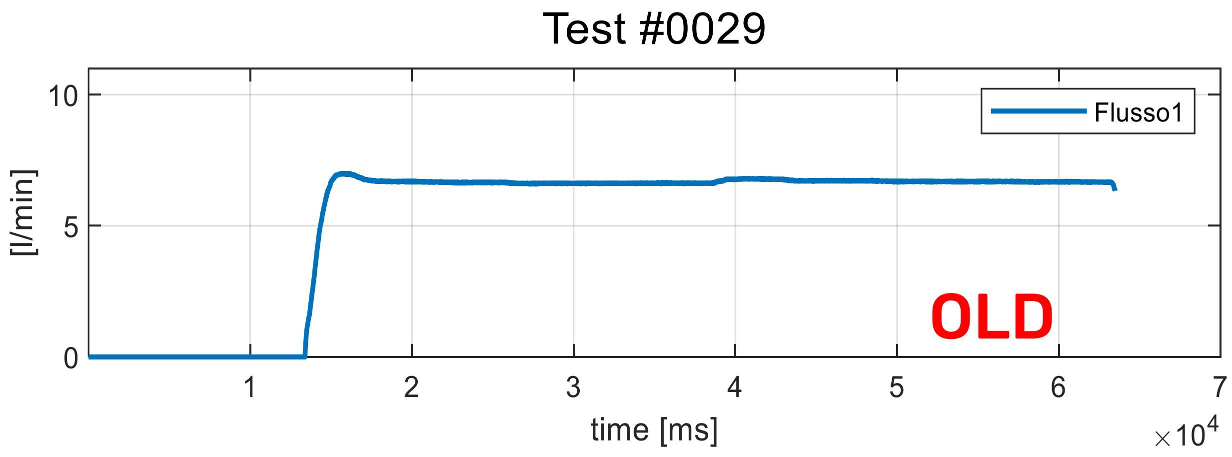
1. Problem statement and Experimental setup
2. Recursive system identification - RLS
3. Leak/NO Leak Classification – Idea and Algorithm
4. Leak /NO Leak Classification – Logistic Regression
5. Leak /NO Leak Classification – Anomaly Detection
6. **Leak /NO Leak Classification – New Prototype**
7. Conclusion

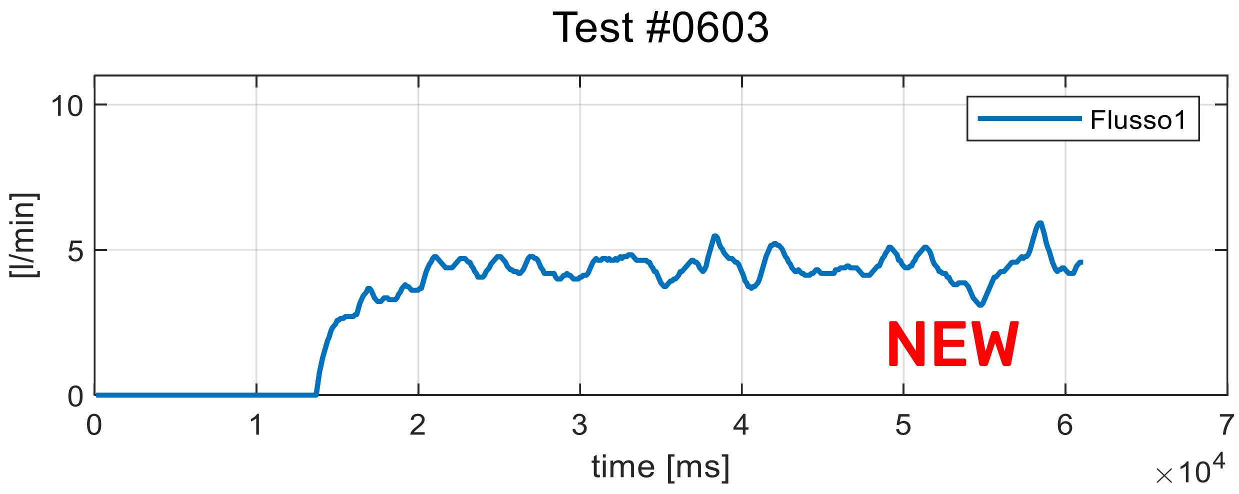
Leak/NO Leak Classification – New Prototype

Once both two methods presented before was developed, due to cost and valve construction reasons, **sensors have been changed**.

New sensors provide much more noisy measurements and their resolution is greater.

These characteristics affect the perfomance of both the presented methods.





Leak/NO Leak Classification – New Prototype

With the new prototype, the method that use **RLS + logistic regression** fails in the classification step.

So, only the method **RLS + Anomaly Detection** has been implemented on the valve's on-board logic.

The if-then-else logic for cases 1 and 3 remains the same.

Leak/NO Leak Classification – New Prototype

UTILITY OPEN

System condition 2 or 4

adaptive Recursive Least Squares (RLS)

$\mathbb{E}[\widehat{\boldsymbol{\theta}}_{1,h}], \mathbb{E}[\widehat{\boldsymbol{\theta}}_{2,h}]$
(OFFLINE approach)

NO LEAK/LEAK

		UTILITY	
		CLOSED	OPEN
LEAK	DOESN'T EXIST	1	2
	EXIST	3	4

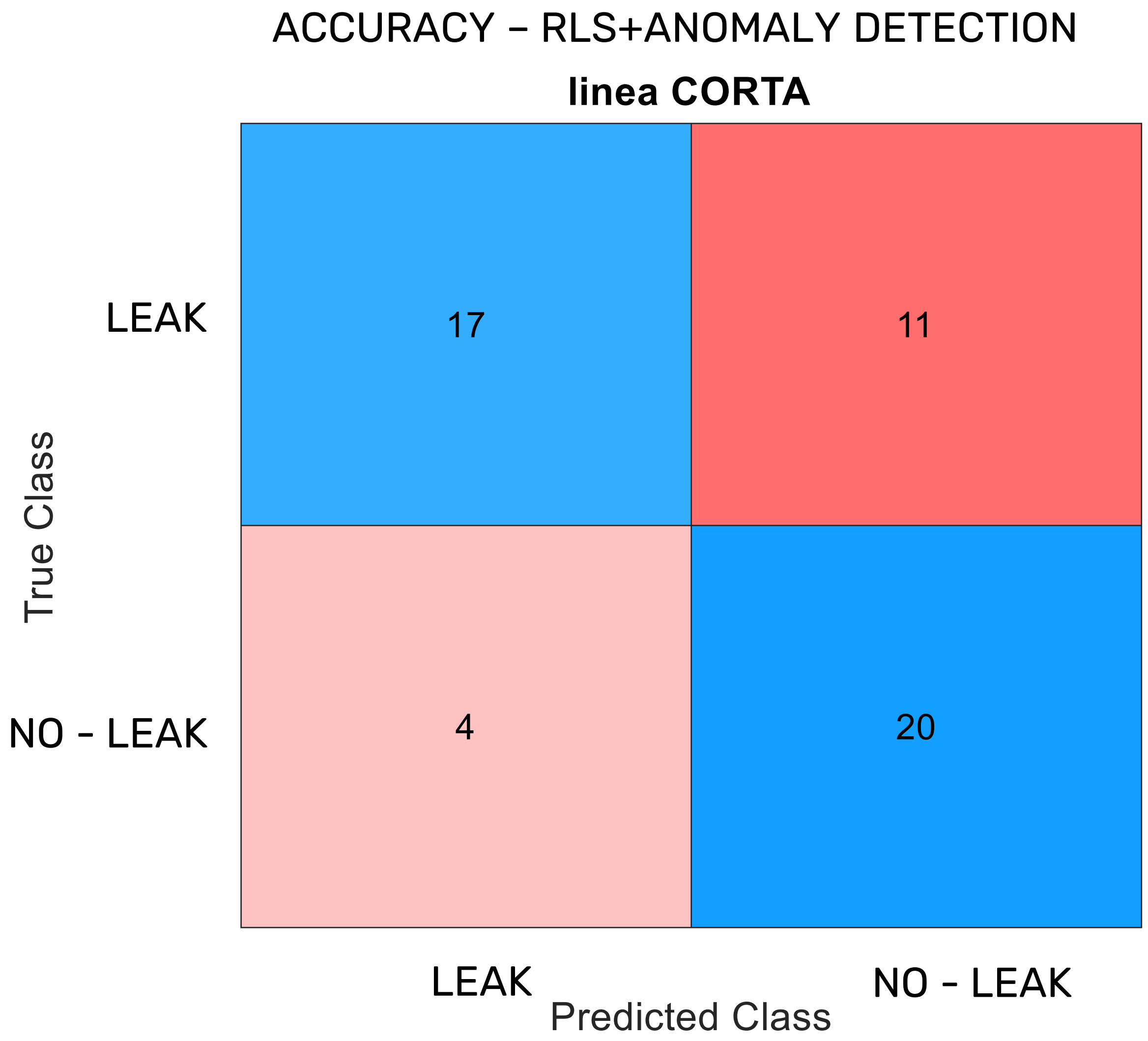
Tuning parameters:

- Data window lenght N_w
- Forgetting factor RLS μ
- Anomaly threshold ϵ

ANOMALY DETECTION with MULTIVARIATE NORMAL DISTRIBUTION

Outline

Conclusion



It was possible to validate the method on two different configurations of the test plumbing system:
- *«linea CORTA»*
- *«linea LUNGA»*

- 17 leak correctly detected
- 11 leak **not** detected (**falsi NO-leak**)
- 20 NO-leak correctly classified
- 4 NO-leak classified as leak (**false leak**)

52 total amount of test

$71.15\%\ \boldsymbol{correct}$

$21.15\%\ false\ \boldsymbol{NO-leak}$

$7.7\%\ false\ \boldsymbol{leak}$

Conclusion

ACCURACY – RLS+ANOMALY DETECTION

linea LUNGA

True Class \ Predicted Class	LEAK	NO - LEAK
LEAK	7	5
NO - LEAK	**0**	12

It was possible to validate the method on two different configurations of the test plumbing system:

- *«linea CORTA»*
- *«linea LUNGA»*

- 7 leak correctly detected
- 5 leak **not** detected (**falsi NO-leak**)
- 12 NO-leak correctly classified
- 0 NO-leak classified as leak (**false leak**)

24 total amount of test

$79.17\%\ \boldsymbol{correct}$

$20.83\%\ false\ \boldsymbol{NO - leak}$

UNIVERSITÀ DEGLI STUDI DI BERGAMO

Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione