Metrology for the Factory of the Future

Mathematical Modelling and Algorithms for the Aggregation of data from Industrial Sensor Networks

Yuhui Luo

Data Science Dept
NPL

Gertjan Kok

Applied Mathematics
VSL

Loïc Coquelin

Département Science des Données et Incertitude
LNE

23 Sep 2021

Contents

Acknowledgement

- This presentation contains contributions from our partners, including VSL, LNE and University of Cambridge.

- As a presenter of their work, my thanks go to Kavya Jagan, Liam Wright and Peter Harris of NPL and Bang Xiang Yong of University of Cambridge for their supports.

Background

Current Industrial Trend

- Large amount of sensors become available
- Increased computational capacity
- The wider applications of ML algorithm
- Concerns of confidence of sensor data and algorithm output

A few challenges in metrology for the Factory of Future

- Data aggregation & Data cleaning
- Data Fusion
- Network synchronisation and timing
- Redundant measurements
- Network design for mixed quality sensors
- Multiagent system

Network Synchronisation and Timing

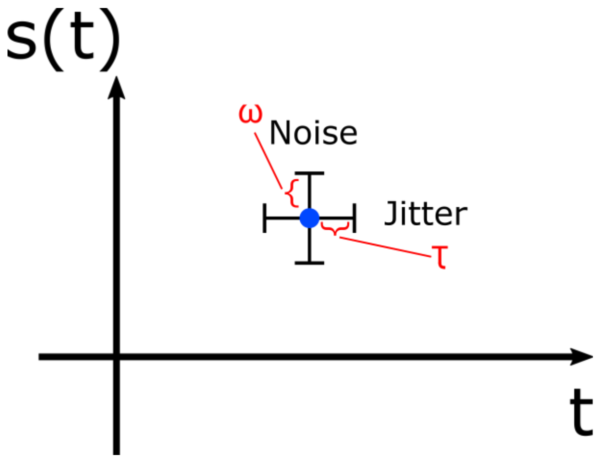
Our aim was to formulate and answer the following questions:

- For the measured data, how to model the data in the presence of synchronisation effects and jitter?
- How to estimate the model parameters?
- How to evaluate the uncertainty of the result?

Data model for synchronisation effects and jitter

- Model for measured signal

$$v(t) = s(t + \xi) + \eta$$



- $s(t)$: true data
- η : measurement error with variance ω^2
- ξ : timing error with variance τ^2
- η and ξ are assumed to be samples drawn independently from Gaussian distributions that are independent of time and of the signal and of each other.

- Under some (mild) assumptions

$$\mathrm{E}[v(t)] \approx s(t) + \frac{1}{2}(\tau^2)s^{(2)}(t)$$

$$\mathrm{V}[v(t)] \approx (\tau^2)\{s^{(1)}(t)\}^2 + \omega^2$$

Data model for sychronisation noise and jitter - Continued

- Since the underlying data is unknown, cubic polynomial $\psi(t; \boldsymbol{a})$ is used to approximate the underlying data in a short window, where $\boldsymbol{a}$ are the parameters of the cubic polynomial function.

- The parameters to be estimated: $\boldsymbol{a}$ (parameter of cubic polynomial), ω^2 (variance of noise) and τ^2 (variance of jitter)

- In some circumstances, repeated measurements of the signal can be used to estimate noise and jitter variances.

- However, in several measurement scenarios, repeated measurements are not recorded or not possible and the noise and jitter variances need to be estimated on the basis of the available data.

Parameter Estimation

- Use Bayesian approach, likelihood function:

$$h(\boldsymbol{v}|\boldsymbol{a},\tau^2,\omega^2) \propto \prod_{j=1}^{N} \frac{1}{\sqrt{\tau^2\psi^{(1)}(t_j,\boldsymbol{a})^2+\omega^2}} \exp\left\{-\frac{(v_j-\psi(t_j;\boldsymbol{a})-0.5\,\tau^2\psi^{(2)}(t_j;\boldsymbol{a}))^2}{2(\tau^2\{\psi^{(1)}(t_j;\boldsymbol{a})\}^2+\omega^2)}\right\}$$

- $\psi^{(1)}(\bullet)$ and $\psi^{(2)}(\bullet)$ are the first and second derivatives of the cubic polynomial

- By Bayes' theorem, the posterior distribution is given by

$$g(\boldsymbol{a},\tau^2,\omega^2|\boldsymbol{v}) \propto h(\boldsymbol{v}|\boldsymbol{a},\tau^2,\omega^2)g(\boldsymbol{a})g(\tau^2)g(\omega^2)$$

Uncertainty of the Result from Parameter Estimation

- Selection of the prior used to determine the in the posterior distribution

$$g(\boldsymbol{a}, \tau^2, \omega^2 | \boldsymbol{v}) \propto h(\boldsymbol{v} | \boldsymbol{a}, \tau^2, \omega^2) g(\boldsymbol{a}) g(\tau^2) g(\omega^2)$$

where $g(\boldsymbol{a}) \propto 1$ - non-informative prior

$g(\omega^2), g(\tau^2) \propto (\tau^2)^{-(\alpha+1)} \exp \frac{-\beta}{\tau^2}$ - inverse gamma distribution prior

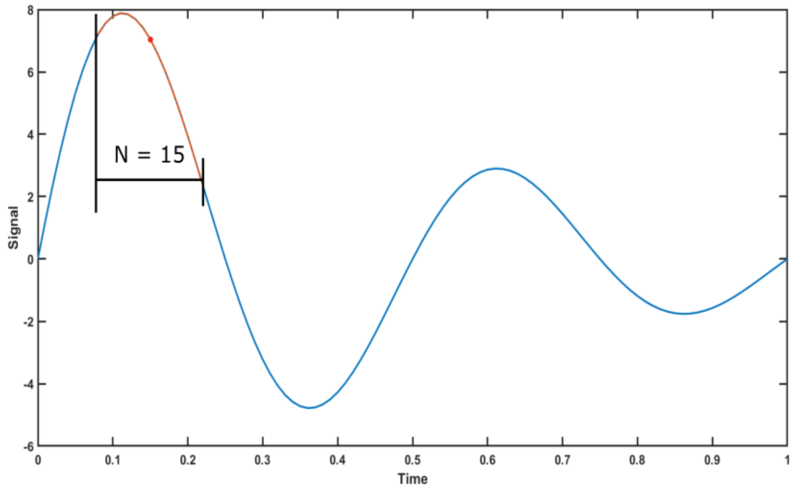
- Generation of the posterior distribution – Metropolis-Hastings algorithm

- See: Jagan, K., Wright, L. and Harris, P., A Bayesian approach to account for timing effects in industrial sensor networks, 2020 IEEE International Workshop on Metrology for Industry 4.0 IoT

Numerical Example

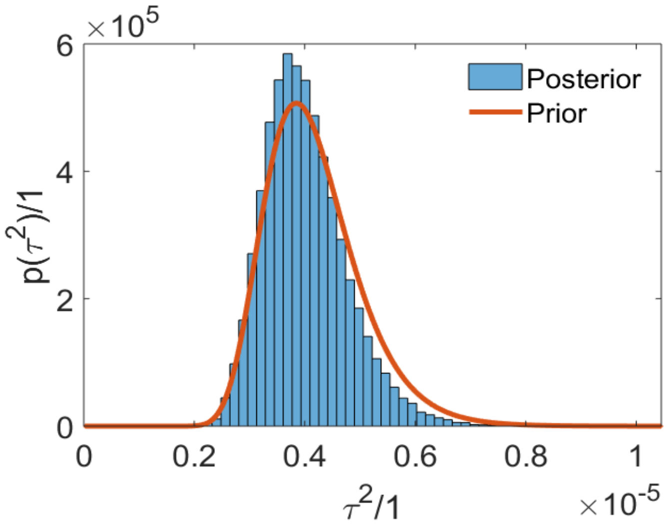
- Using the simulated signal

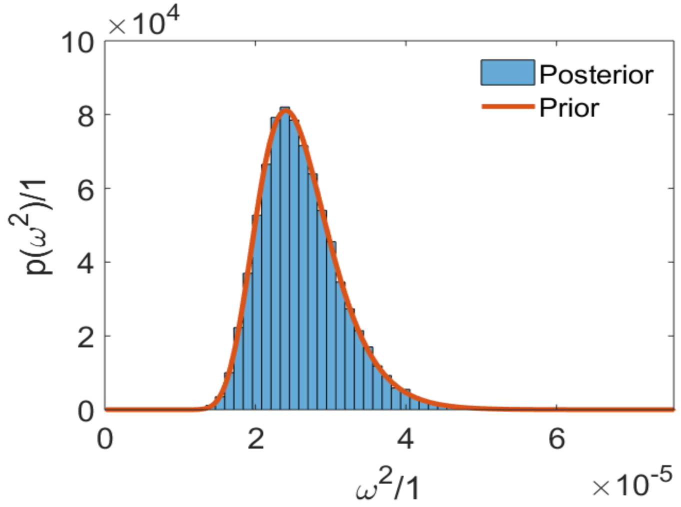
$$s(t) = a\, e^{-bt} \sin(2\pi f t)\,, \qquad 0 \le t \le 1$$

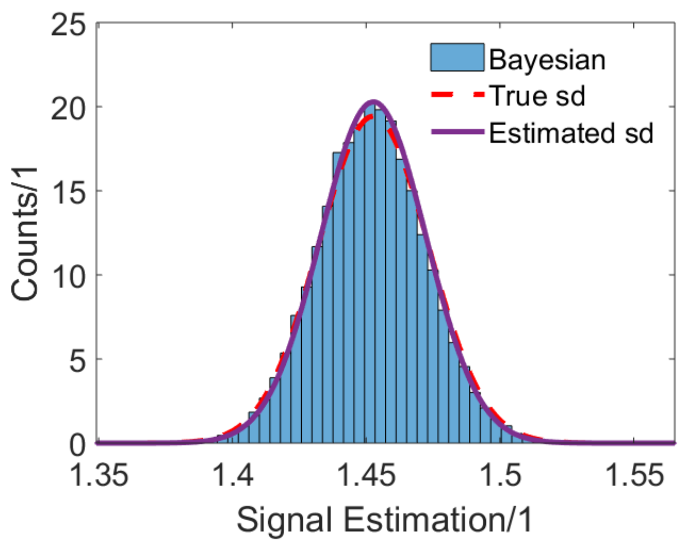


- Where $a = 10$, $b = 2$ and $f = 2$. Signal is sampled with uniform spacing of 0.01s. Jitter and noise levels are set as $\tau = 0.002$ and $\omega = 0.005$ respectively.

Numerical Example - Continued

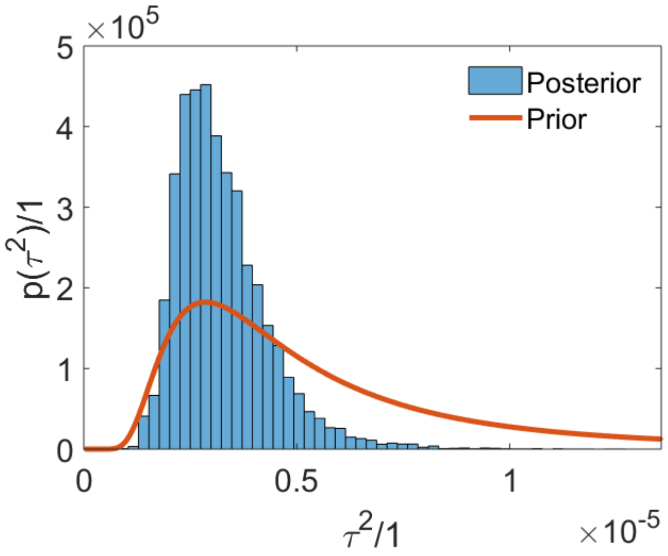


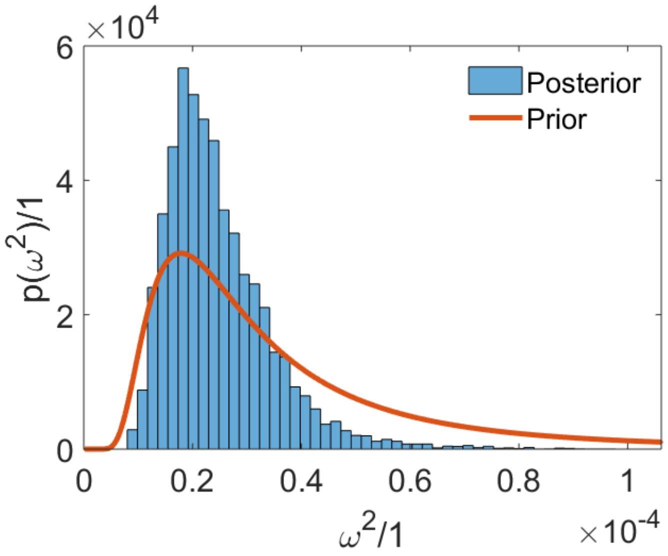


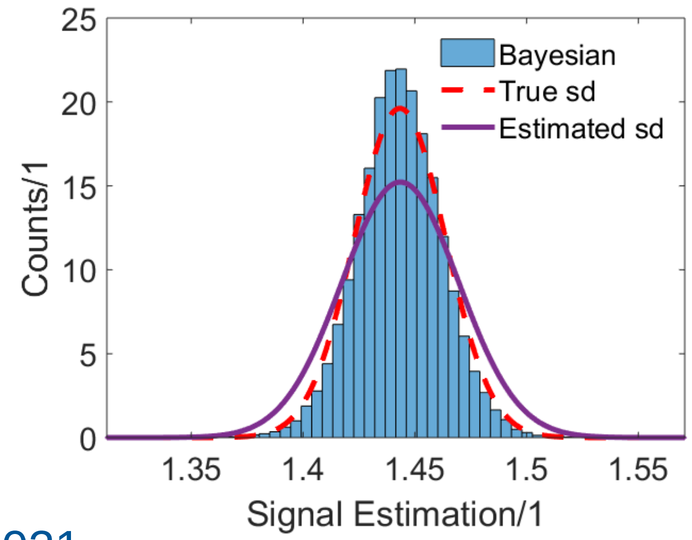


- This is the result of using well informed priori.
- It produces posteriors that are very similar to the prior distribution
- Good agreement between the distributions of the signal estimate for mid-point of window

Numerical Example - Continued







- Less informed prior and simulated data produce less dispersed posteriors that the priors.
- Surprisingly good agreement between the distributions of the signal estimate for mid-point of window.
- Software will be available in Met4FoF Github repository.

Redundant measurement

Our aim was to formulate and answer the following questions:

- What is redundancy exactly? How can you quantify it?
- How can you take advantage of redundancy?
- How can these ideas be applied to the three testbeds?
- Which non-testbed specific software tools can we make freely available?

What is redundancy in a metrological sensor network?

- **“The property that there are multiple, independent ways of deriving the value of the measurand from the set of measured sensor values, quantified by a notion of gradual uncertainty increase when sensors fail / are removed.”**

- Related concepts:
  - sensor replication: can the data of one sensor be replicated by (a group of) other sensors?
  - sensor relevance: how relevant is a sensor for knowing the measurand value?

- Most useful metrological redundancy metrics:
  - Red-Excess: Calculate the **number of sensors** present in the network **in excess of the minimum number** necessary to determine the values of the measurand
  - Red-Unc-*m*: **Maximum increase in uncertainty** of measurand ***Y*** when taking out *m* relevant sensors

- See: Kok, G. and Harris, P. *Quantifying Metrological Redundancy in an Industry 4.0 Environment,* 2020 IEEE International Workshop on Metrology for Industry 4.0 IoT

How can one take advantage of redundancy?

- Sanity check of network based on measured values, e.g.:
  - Check if synchronized events
  - Check algebraïc and integral equations

$$u(\hat{y}) = \left(\mathbf{e}^{\mathrm{T}} V_{\mathbf{y}}^{-1} \mathbf{e}\right)^{-1/2}$$
$$\hat{y} = u^2(\hat{y})\, \mathbf{e}^{\mathrm{T}} V_{\mathbf{y}}^{-1} \mathbf{y}$$

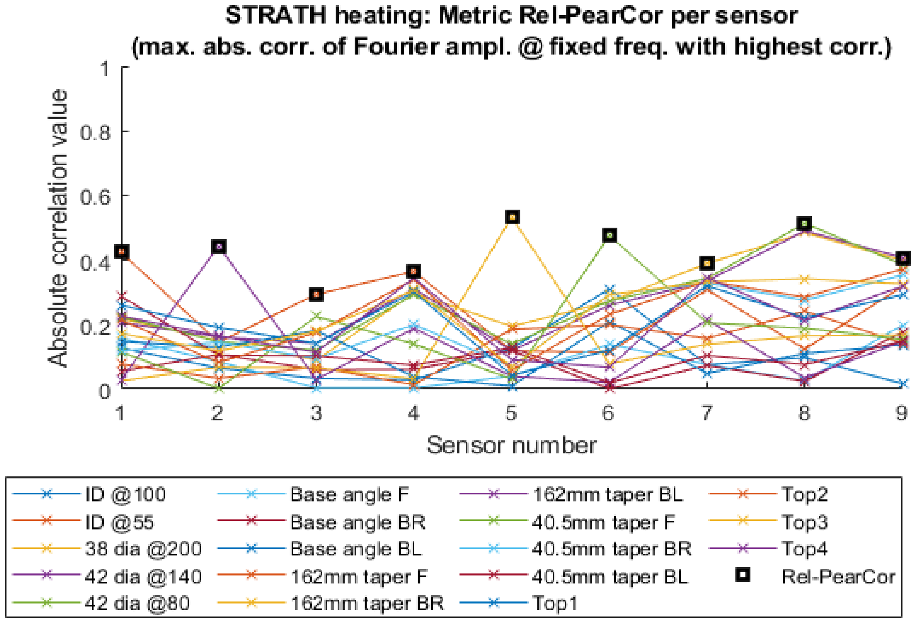
- Identify and reject faulty measurement values
- Reduce measurement uncertainty
  - Using classical tools also used for analyzing intercomparison data
  - Extended to case: $\boldsymbol{y} = \boldsymbol{a} + A\,\boldsymbol{x}$, i.e. possibility to reject individual sensor values x_i instead of estimates y_j of the measurand only. ‘Largest Consistent Subset of Sensors (LCSS) – algorithm’

- Interpolation and calibration of uncalibrated sensors
  - Study of various mathematical models (e.g. Gaussian Processes) for distributed measurement in SPEA testbed. What is the best model, what is the uncertainty?

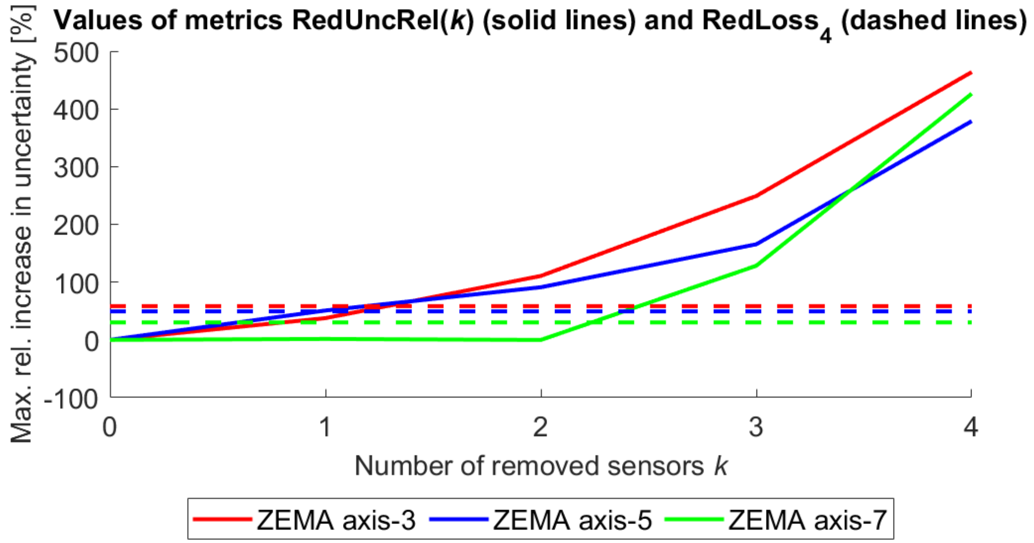
- See: Kok, G. and Harris, P. *Uncertainty Evaluation for Metrologically Redundant Industrial Sensor Networks*, 2020 IEEE International Workshop on Metrology for Industry 4.0 IoT

Application to testbeds

- STRATH:
  - Sanity checks, e.g. comparing integrated speed with position
  - Redundancy metrics: Assess relevance of each sensor for measurand based on Pearson correlation value of selected data feature



- ZEMA:
  - Redundancy metrics, e.g. Red-Unc-*m* metric: Uncertainty increase when taking out sensors (model dependent)
  - Combing residual life time estimates based on calculated uncertainties
  - Application of LCSS algorithm to remove erroneous sensor values. (No improvement in prediction quality...)



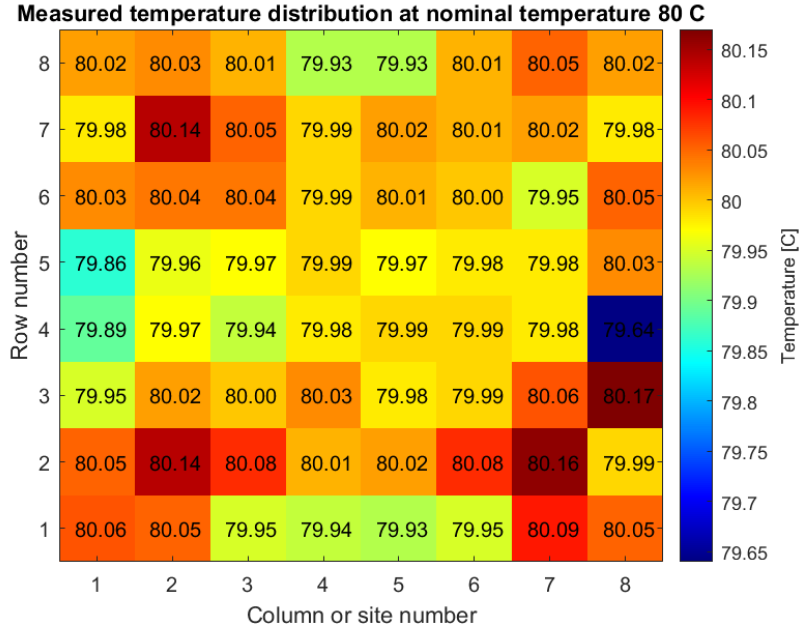
Application to testbeds (2)

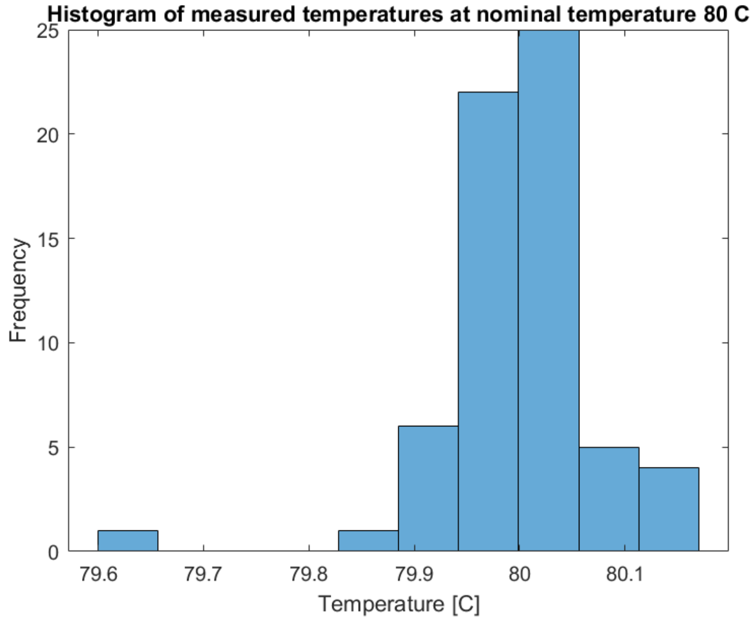
- SPEA:
  - Identify strange values
  - Interpolate between reference sensors
  - Nearest Neighbour model with additional uncertainty seems more appropriate than Gaussian Process with significant correlation structure

- Some general software tools integrated in AgentMet4FoF framework and separate, non-agent versions (LCSS)

- Software available in Met4FoF Github repository: https://github.com/Met4FoF/Met4FoF-redundancy

- Documentation available in ReadTheDocs: https://met4fof-redundancy.readthedocs.io/en/master/





National Metrology Institute

Network design for mixed quality sensors

Our aim was to formulate and answer to following questions:

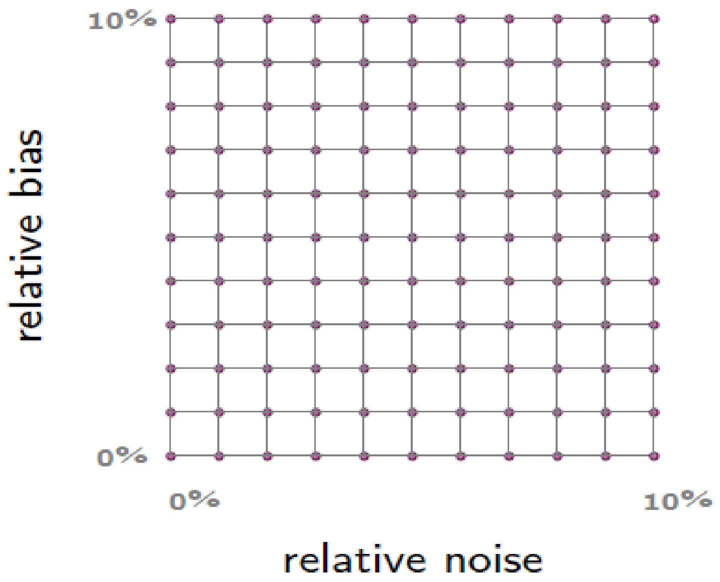
- How to design an experiment to model the influence of mixed quality sensors?
- Taking into account the quality issue of sensors, how to select the relevant sensor and the relevant features from a big pool?
- What happen if uncertainties related to features are concerned ?

Design of Experiment and Data Model

- The mixed quality sensors are modelled as data with different levels of additive noise.

- Evaluation on a 2D cartesian Grid

  - relative bias : 0% to 10% of the mean
  - additive noise : 0% to 10% of the mean
  - 100 classification accuracy evaluations



Design of Experiment and Data Model -Continued

- Notation
  - $\mathcal{D}$, the training set such that $\mathcal{D} = \{(X_{\cdot i}, t_i), i = 1, \ldots, n\}$
  - n_s, the number of sensors
  - s_k feature set of sensor k
  - d_k, the number of features extracted from sensor k
  - d, the total number of extracted features, $d = \sum_{k=1}^{n_s} d_k$
  - $X_{\cdot i}$, the input vector storing the i-th observation from the whole set of sensors
  - $\phi(X_{\cdot i}) \in \mathbb{R}^d$, the d-dimensional feature vector extracted from the input $X_{\cdot i}$

- Modelling

The target t_i is representative of the true model y_i with the addition of noise ϵ_i

$$\begin{aligned} t_i &= y_i + \epsilon_i, \\ t_i &= w^T \phi(X_{\cdot i}) + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \tau^{-1}) \end{aligned}$$

Features Selection

- The algorithms of Relevance Vector Machine (RVM) and its extension Relevant Group Selector (RGS) are applied to select relevant sensors and the relevant features.
- The RVM and RGS algorithms are hierarchical Bayesian formulations that allow to introduce grouping (features from different sensor will correspond to a different group) and to simultaneously perform feature selection within groups to reduce over-fitting of the data.
- Introducing the $\lambda_{1,j}$ and $\lambda_{2,k}$ whose inverse are related to the weights of sensors and the features respectively, the RGS algorithm model can be summarized as:

$$
\begin{aligned}
p(t|X, w, \tau) &= \prod_{i=1}^{n} \mathcal{N}(t_i | w^T \phi(X_{\cdot i}), \tau^{-1}) \\
p(w|\lambda_1, \lambda_2) &= \prod_{k=1}^{n_s} \prod_{j \in s_k} \mathcal{N}\left(w_j | 0, (\lambda_{1,j} + \lambda_{2,k})^{-1}\right)
\end{aligned}
$$

Features Selection - Continued

- Further sparse feature selection is proposed by either weight thresholding or elbow method

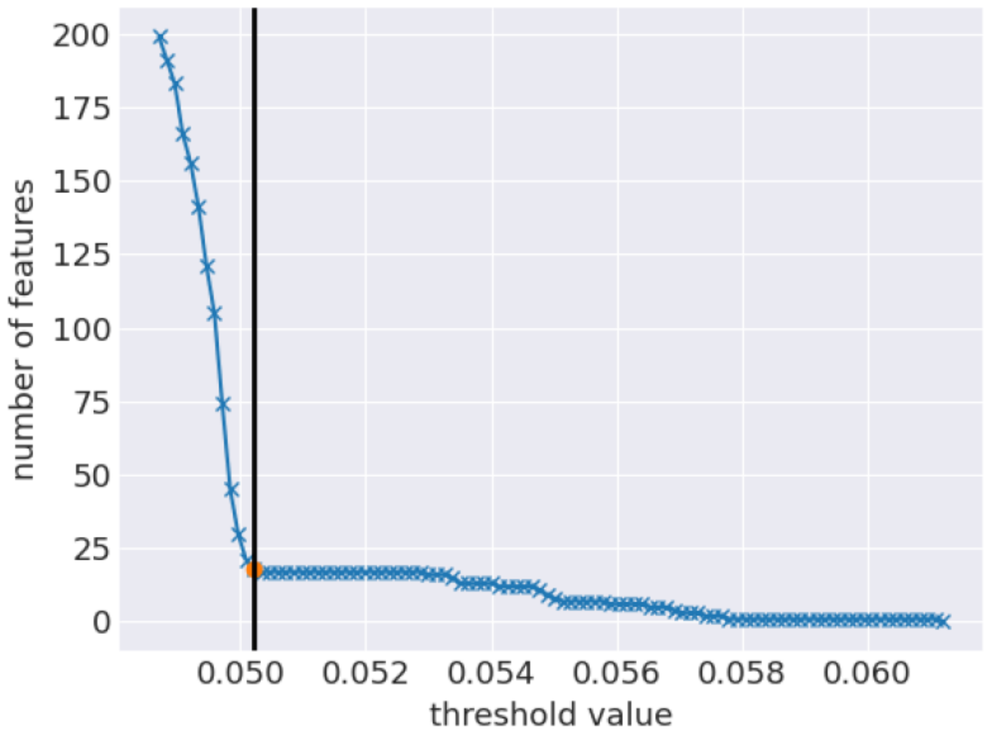


illustration of the elbow method to find the optimal number of features to select

When Uncertainty of Feature Included

- Homogeneous Uncertainty – features have same variance in uncertainty

$$\Phi_{\cdot i} \sim \mathcal{N}_d\left(\phi(X_{\cdot i}), \sigma_\phi^2 I_d\right)$$

$$p(t|X, w, \tau) = \prod_{i=1}^{n} \mathcal{N}(t_i|w^T \Phi_{\cdot i}, \tau^{-1})$$

$$p(w|\lambda_1, \lambda_2) = \prod_{k=1}^{n_s} \prod_{i \in s_k} \mathcal{N}\left(w_j|0, \sigma_\phi^2 + (\lambda_{1,j} + \lambda_{2,k})^{-1}\right)$$

- Heterogeneous uncertainty case – different features has difference variance in uncertainty

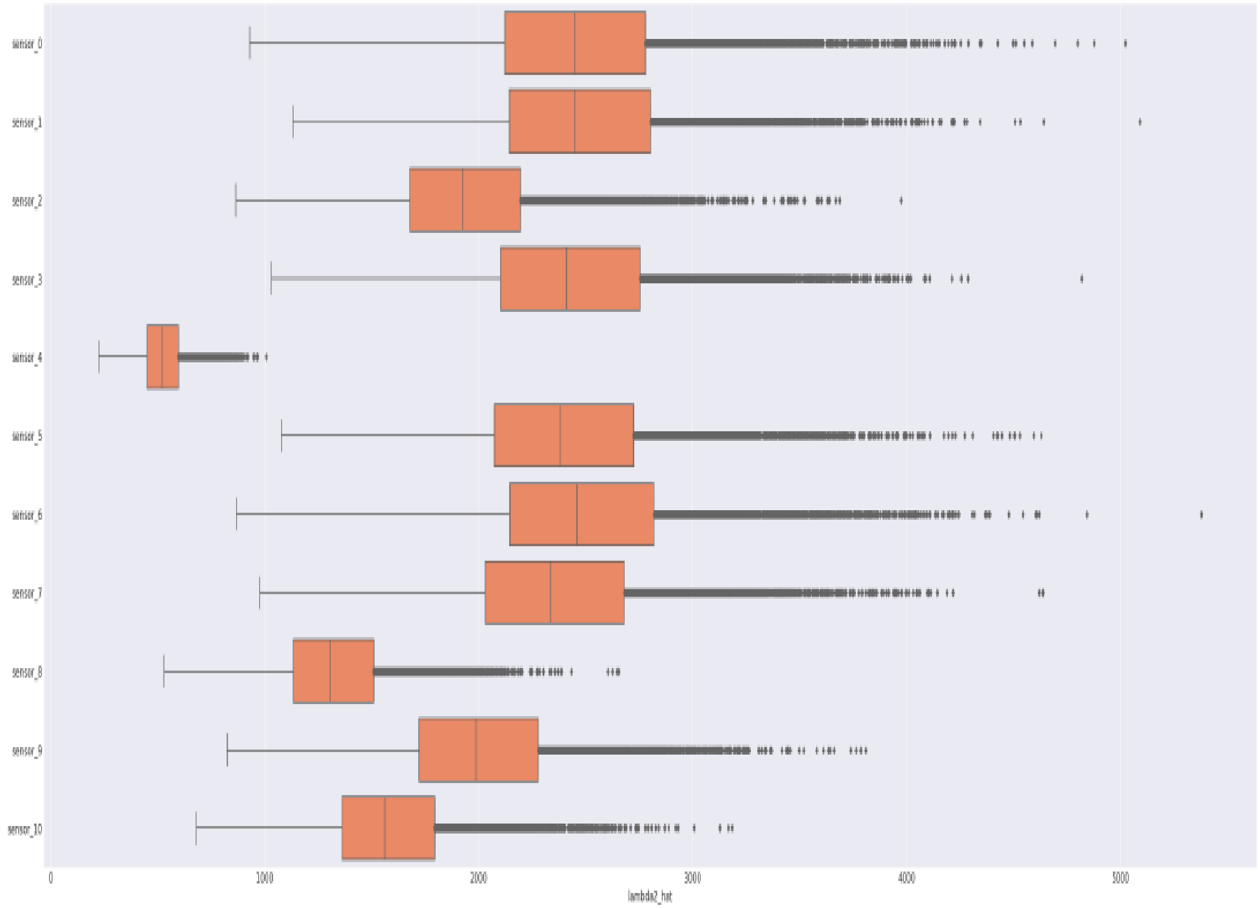
$$\Phi_{\cdot i} \sim \mathcal{N}_d\left(\phi(X_{\cdot i}), \Sigma_\phi\right)$$

$$p(t|X, w, \tau) = \prod_{i=1}^{n} \mathcal{N}(t_i|w^T \Phi_{\cdot i}, \tau^{-1})$$

$$p(w|\lambda_1, \lambda_2) = \prod_{k=1}^{n_s} \prod_{j \in s_k} \mathcal{N}\left(w_j|0, \sigma_{\phi_j}^2 + (\lambda_{1,j} + \lambda_{2,k})^{-1}\right)$$

ZeMA Result

- sensor 0: microphone
- sensor 1: acceleration plain bearing
- sensor 2: acceleration piston rod
- sensor 3: acceleration ball bearing
- sensor 4: axial force
- sensor 5: pressure
- sensor 6: velocity
- sensor 7: active current
- sensor 8: motor current phase 1
- sensor 9: motor current phase 2
- sensor 10: motor current phase 3



- sensor 4 $\longrightarrow$ most relevant sensor
- sensor 8
- sensor 10
- sensor 2, 9
- sensor 0, 1, 3, 5, 6, 7 $\longrightarrow$ irrelevant sensors

Department for Business, Energy & Industrial Strategy

FUNDED BY BEIS

EMPIR EURAMET

The EMPIR initiative is co-funded by the European Union's Horizon 2020 research and innovation programme and the EMPIR Participating States

The National Physical Laboratory is operated by NPL Management Ltd, a wholly-owned company of the Department for Business, Energy and Industrial Strategy BEIS).