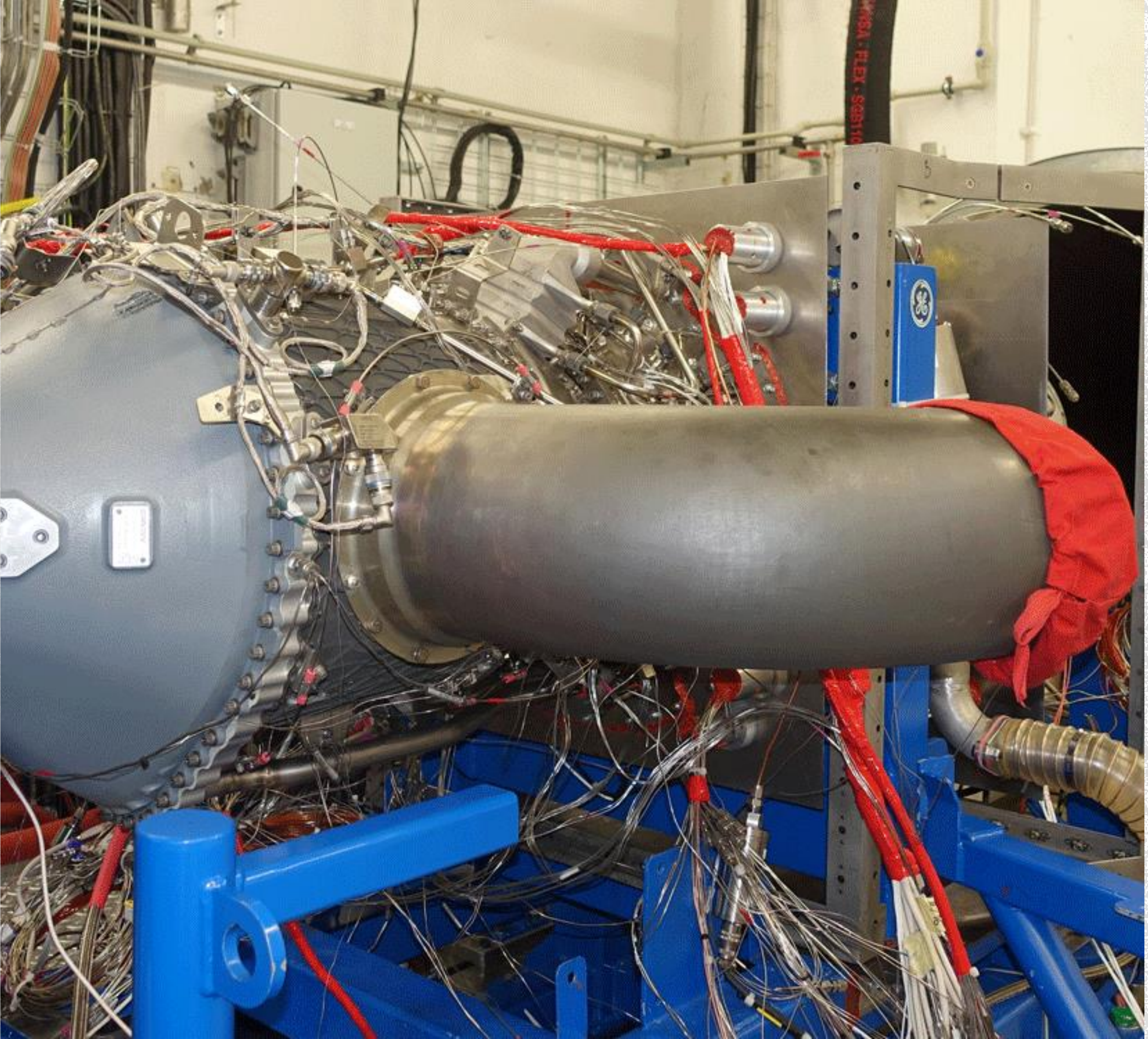
Sensor Selection via Auto-Encoders

Showcase for the aerospace industry

By Fran Urbano



GE AVIATION CATALYST ENGINE

- Around 1,000 sensors in the testbed
- Around 50 sensors in the commercial engine

PERFECT ANALYTICAL METHOD

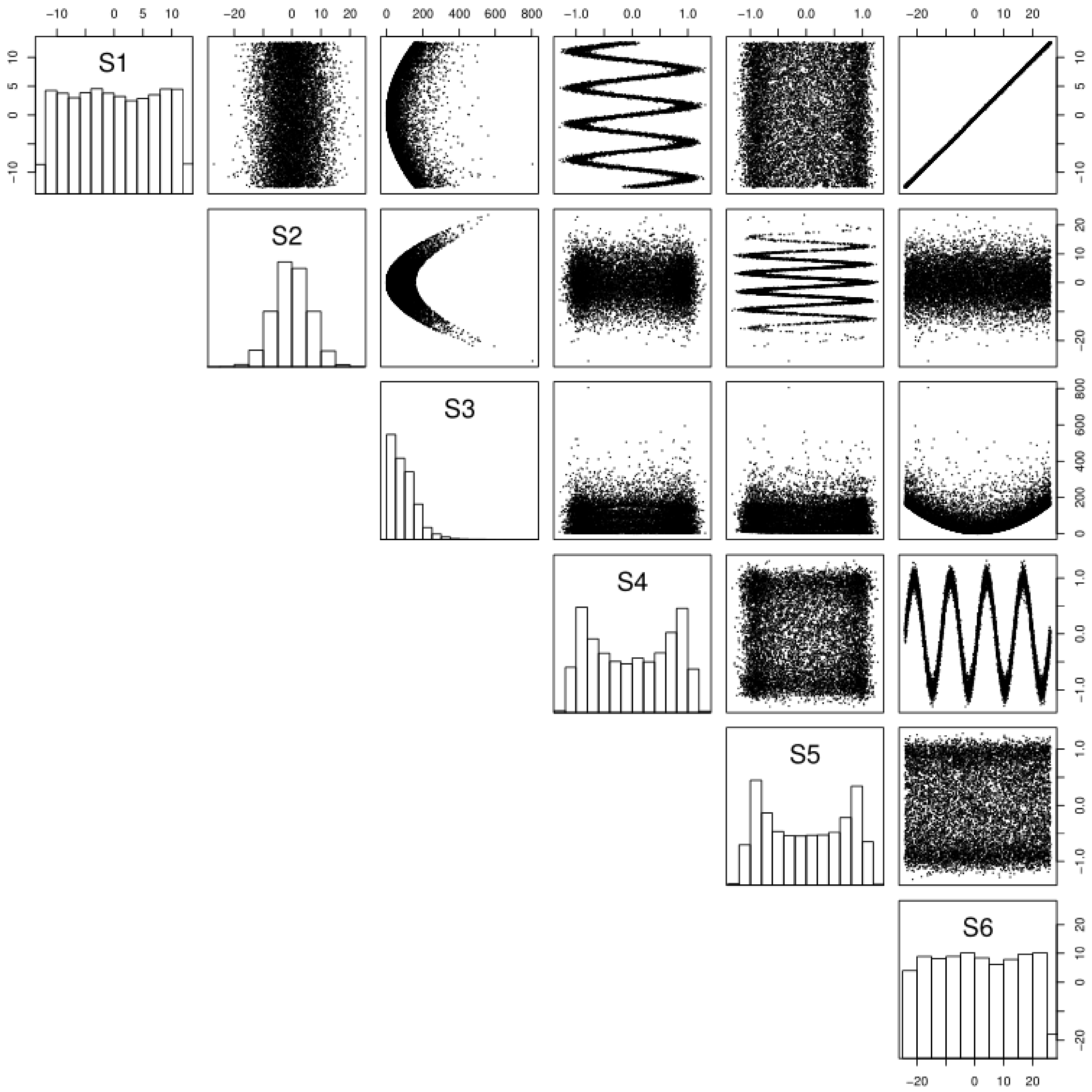
- $\binom{6}{1} + \binom{6}{2} + \cdots + \binom{6}{6} = 2^6 - 1 = 63$
- $\binom{1000}{1} + \binom{1000}{2} + \ldots + \binom{1000}{1000} = 2^{1000} - 1 =$

  10 715 086 071 862 673 209 484 250 490 600 018 105 614 048 117 055 336 074 437 ⋱
  503 883 703 510 511 249 361 224 931 983 788 156 958 581 275 946 729 175 531 468 ⋱
  251 871 452 856 923 140 435 984 577 574 698 574 803 934 567 774 824 230 985 421 ⋱
  074 605 062 371 141 877 954 182 153 046 474 983 581 941 267 398 767 559 165 543 ⋱
  946 077 062 914 571 196 477 686 542 167 660 429 831 652 624 386 837 205 668 069 ⋱
  375

- $\binom{1000}{50} =$ 9 460 461 017 585 217 846 063 722 277 728 044 918 729 694 001 668 654 064 793 569 ⋱
  321 343 252 697 198 115 263 280

REAL LIFE METHOD

- Engineer Driven Decisions
- Past Experience & Know-How
- Regulatory choices
- Technical Requirements
- Redundancies
- Linear heuristics if any at all



SIX SENSORS EXAMPLE BUSINESS CASE

The business wants to reduce the number of sensors.

The maximum loss of information per sensor the business finds acceptable is of 10%

Sensors S1, S2 and S3 are cheaper

SIX SENSORS EXAMPLE

$$X \sim \mathcal{U}(-4\pi, 4\pi),\ x = X(\omega)$$

$$Y \sim \mathcal{N}(0, 2\pi),\ y = Y(\omega)$$

$$\varepsilon \sim \mathcal{N}(0, 0.1)$$

$$S1 = x + \varepsilon$$

$$S2 = y + \varepsilon$$

$$S3 = x^2 + y^2 + \varepsilon$$

$$S4 = \sin(x) + \varepsilon$$

$$S5 = \cos(y) + \varepsilon$$

$$S6 = 2x + 1 + \varepsilon$$

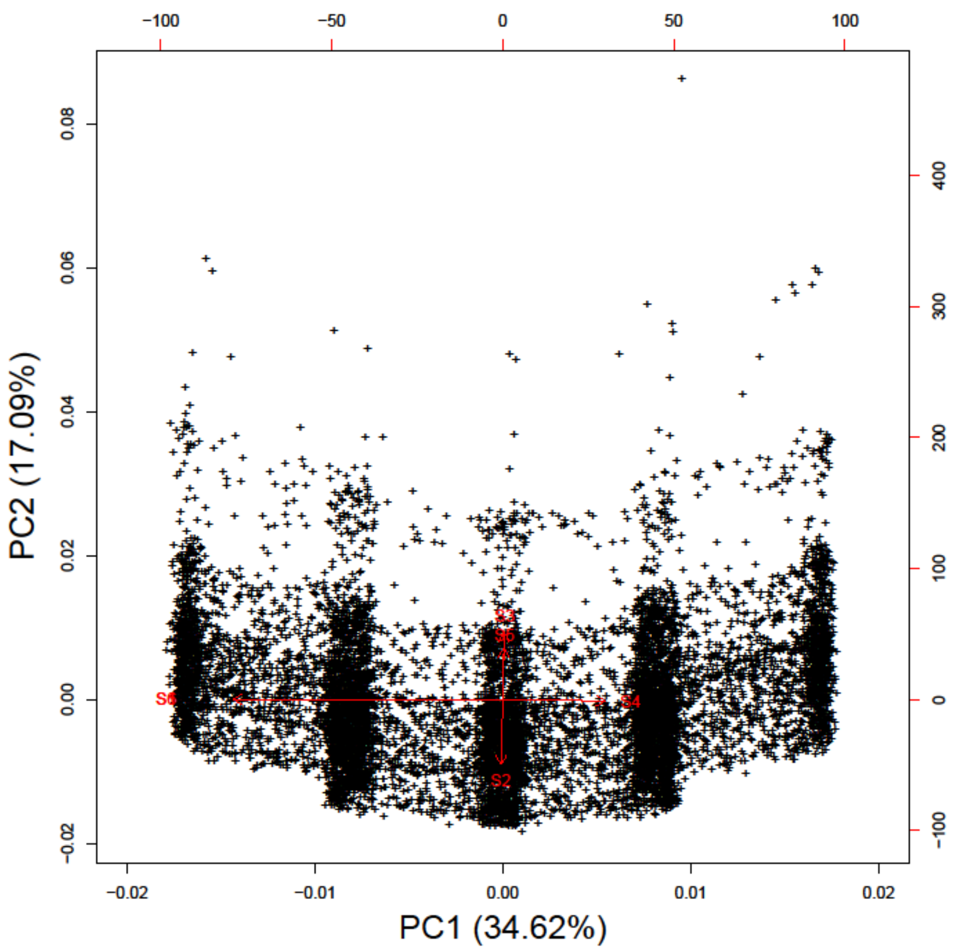
Two sources of information

Only two sensors are linearly related

LINEAR HEURISTIC

1. We apply a PCA to the sensor's scaled data.
2. If the number of sensors is not fixed, we estimate what is the number of sensors we need to meet business requirements based on the number of principal components required to explain a given proportion of variance information.
3. We account for the weight of the of the sensors in each principal component as well as the correlations between sensors to select the most relevant sensors and discard the others.
4. We confirm that the remaining sensors can appropriately predict the data contained within the discarded sensors.
5. If we don't meet the business requirements we increase the number of sensors estimated in step two and repeat the process until we do so and, if we are well above, we can try selecting a fewer number of sensors until we cannot eliminate more.

PCA



	PC1	PC2	PC3	PC4	PC5	PC6
Standard deviation	1.4413	1.0126	0.9969	0.9912	**0.9598**	0.0107
Proportion of Variance	0.3462	0.1709	0.1656	0.1637	**0.1535**	0.0000
Cumulative Proportion	0.3462	0.5171	0.6827	0.8464	**1.0000**	1.0000

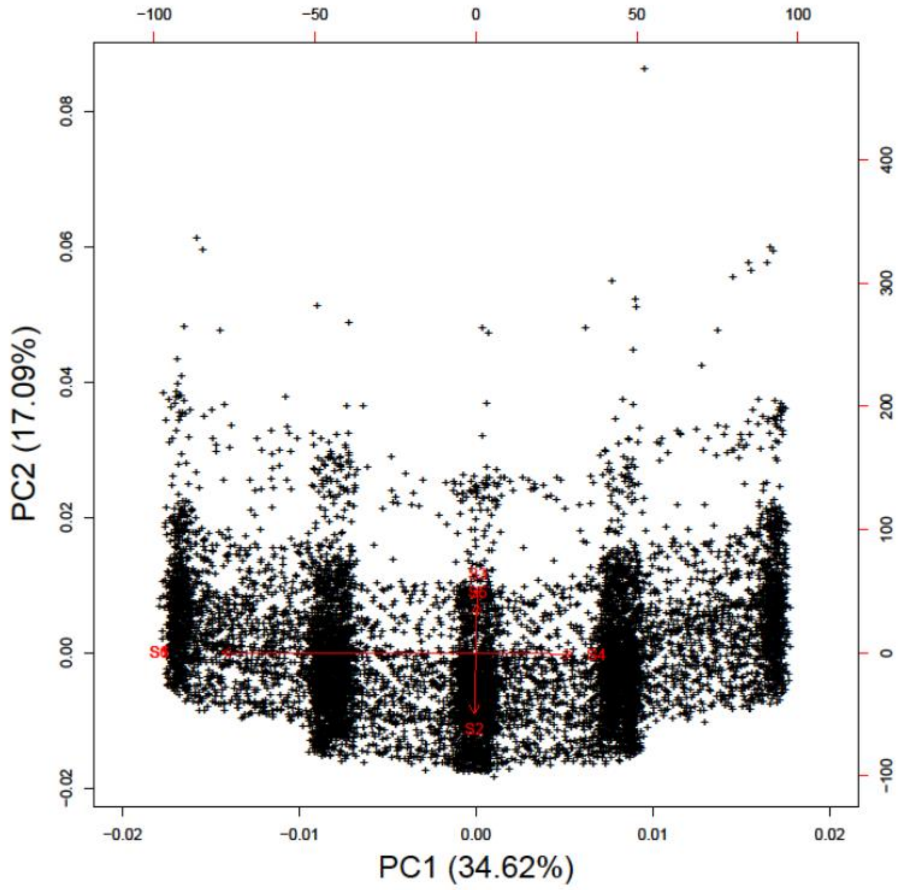
SELECTING SENSORS

	PC1	PC2	PC3	PC4	PC5	PC6
S1	-0.2365	0.0003	-0.0038	0.0022	0.0221	-0.0000
S2	-0.0016	-0.1023	0.0719	-0.1051	0.0144	0.0000
S3	0.0011	0.1077	-0.0383	-0.1211	0.0048	-0.0000
S4	0.0713	-0.0024	-0.0200	0.0111	0.1467	-0.0000
S5	0.0010	0.0813	0.1422	0.0231	0.0161	0.0000
S6	-0.0000	0.0003	-0.0039	0.0022	0.0221	0.0000

Table 5: PCA Rotation Matrix - Correlation & POV Scaled

Sensor	Rank	Score
S2	1	0.2953
S3	2	0.2730
S1	3	0.2650
S5	4	0.2636
S4	5	0.2516
S6	6	0.0285

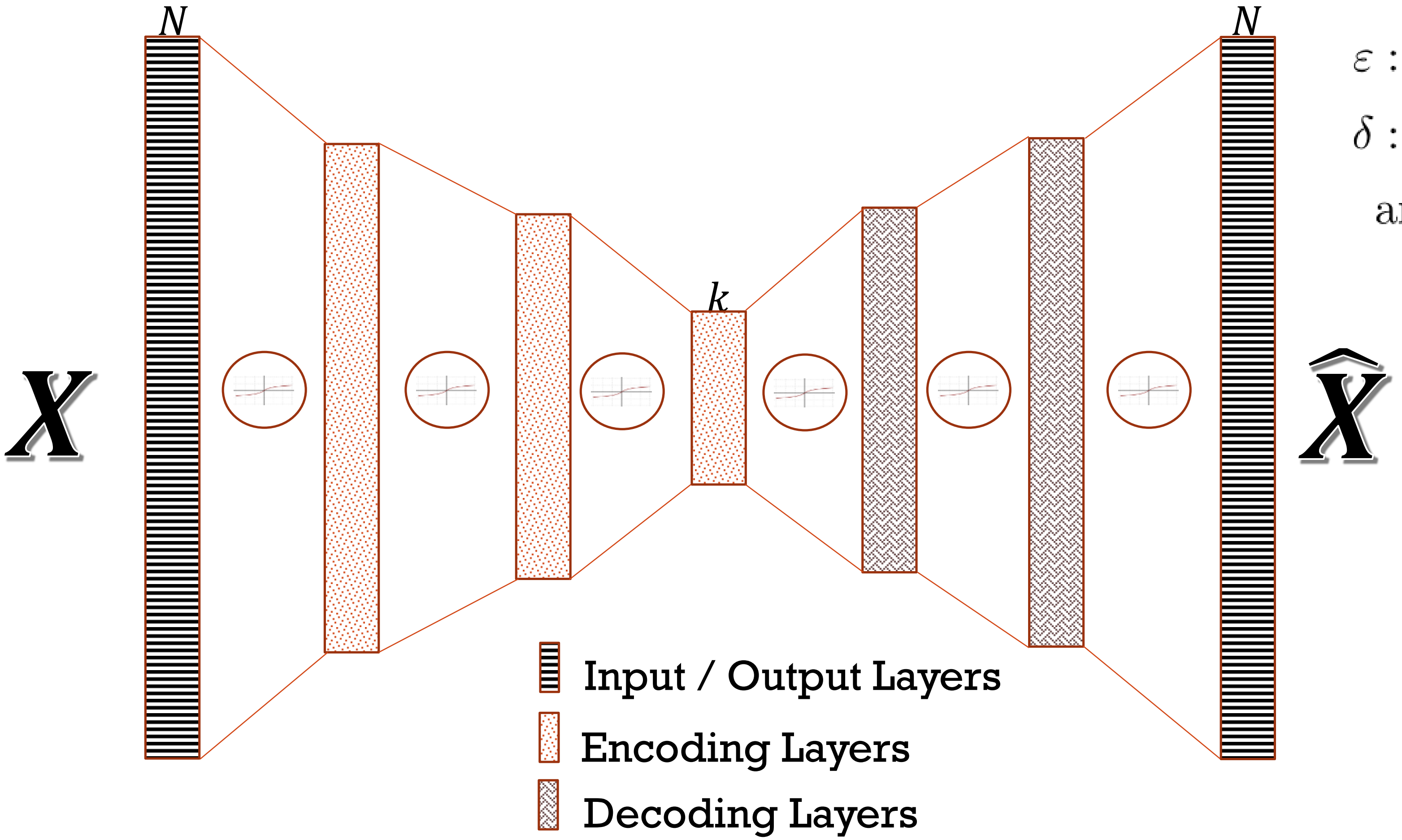
Table 6: Sensors Scores and Rank



NON-LINEAR HEURISTIC — AUTOENCODER

1. We apply a linear methodology to remove/select sensors with linear relationships —linear methods are much faster and easier to apply to detect linear relationships.
2. If the number of sensors is not set, we train the autoencoder with a number of central nodes M equal to the maximum number of sensors we can afford in our commercial product from the N total number of sensors measured or simulated.
3. We implement a **deregulator** —and its secondary goals if required— and analyze the activation of the encoder part of the autoencoder to select the sensors.
4. We retrain the autoencoder muting the unselected sensors and we calculate the information loss and, if acceptable, we reduce M —either in a binary search fashion to guarantee a log_2(M) number of steps or evaluating the training weights— and we retrain the autoencoder until we find the smallest k number of central nodes that have an acceptable information loss.

AUTOENCODER

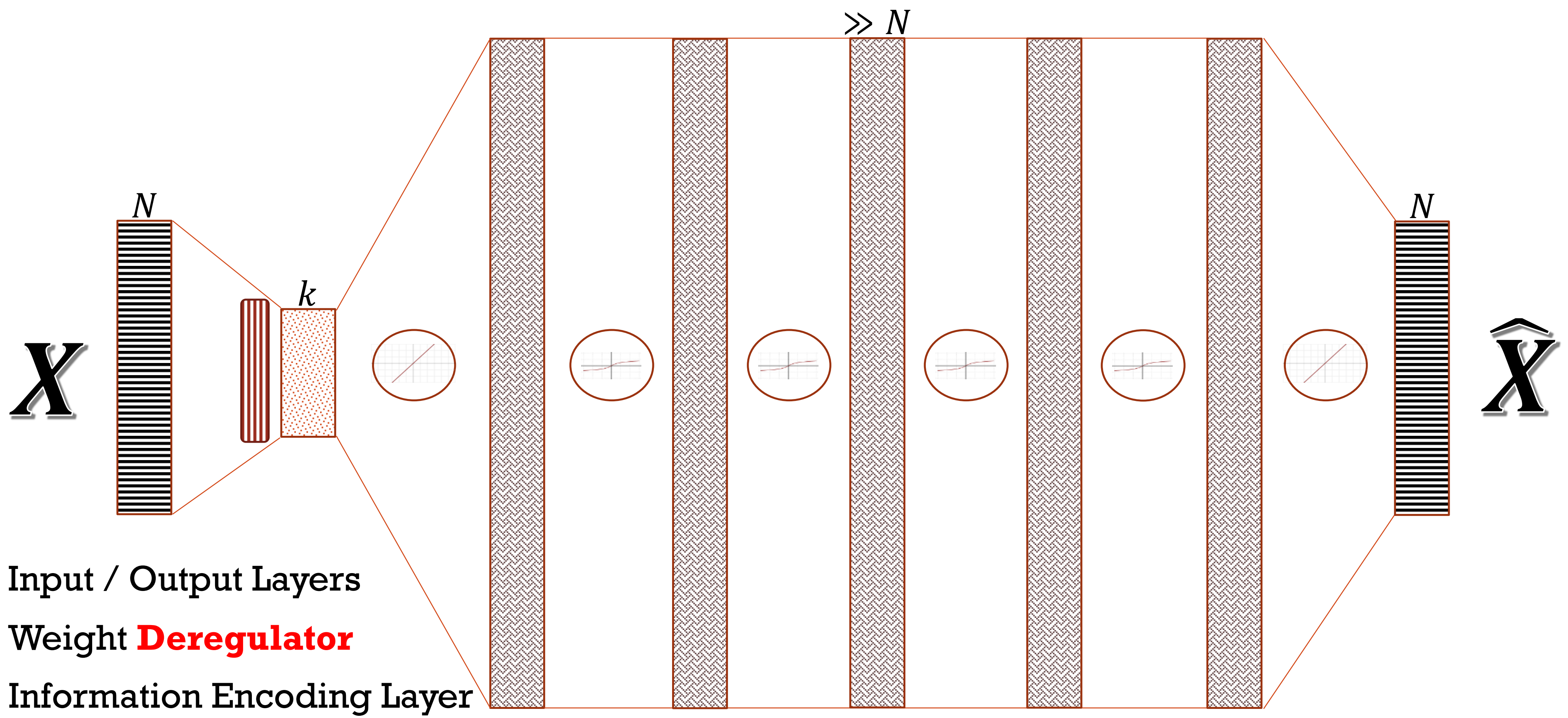


$$\varepsilon : X^n \to Y^k$$

$$\delta : Y^k \to X^n$$

$$\arg\min_{\varepsilon,\delta} \|X^n - (\delta \circ \varepsilon)(X^n)\|^2$$

SELECTION AUTOENCODER



Input / Output Layers

Weight **Deregulator**

Information Encoding Layer

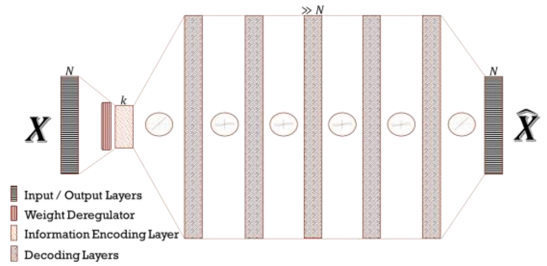
Decoding Layers

DEREGULATOR

$$\varepsilon : X^n \to Y^k$$

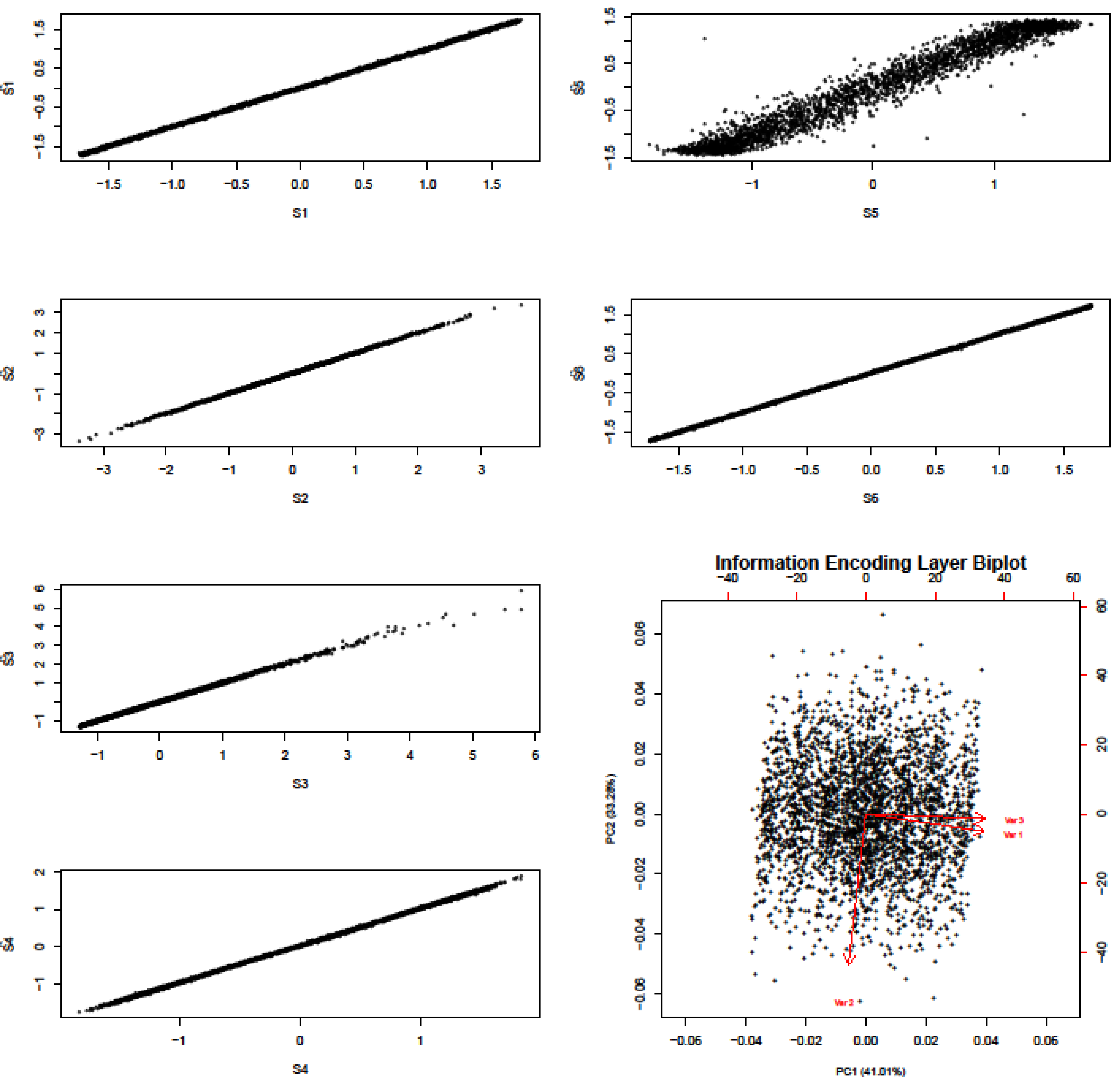
$$\delta : Y^k \to X^n$$

$$\arg\min_{\varepsilon,\delta} \|X^n - (\delta \circ \varepsilon)(X^n)\|^2$$



$$\mathbf{W} = \begin{bmatrix} w_{1,1} \cdot \delta_1 & w_{1,2} \cdot \delta_2 & w_{1,3} \cdot \delta_3 & \cdots \\ w_{2,1} \cdot \delta_1 & w_{2,2} \cdot \delta_2 & w_{2,3} \cdot \delta_3 & \cdots \\ w_{3,1} \cdot \delta_1 & w_{3,2} \cdot \delta_2 & w_{3,3} \cdot \delta_3 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

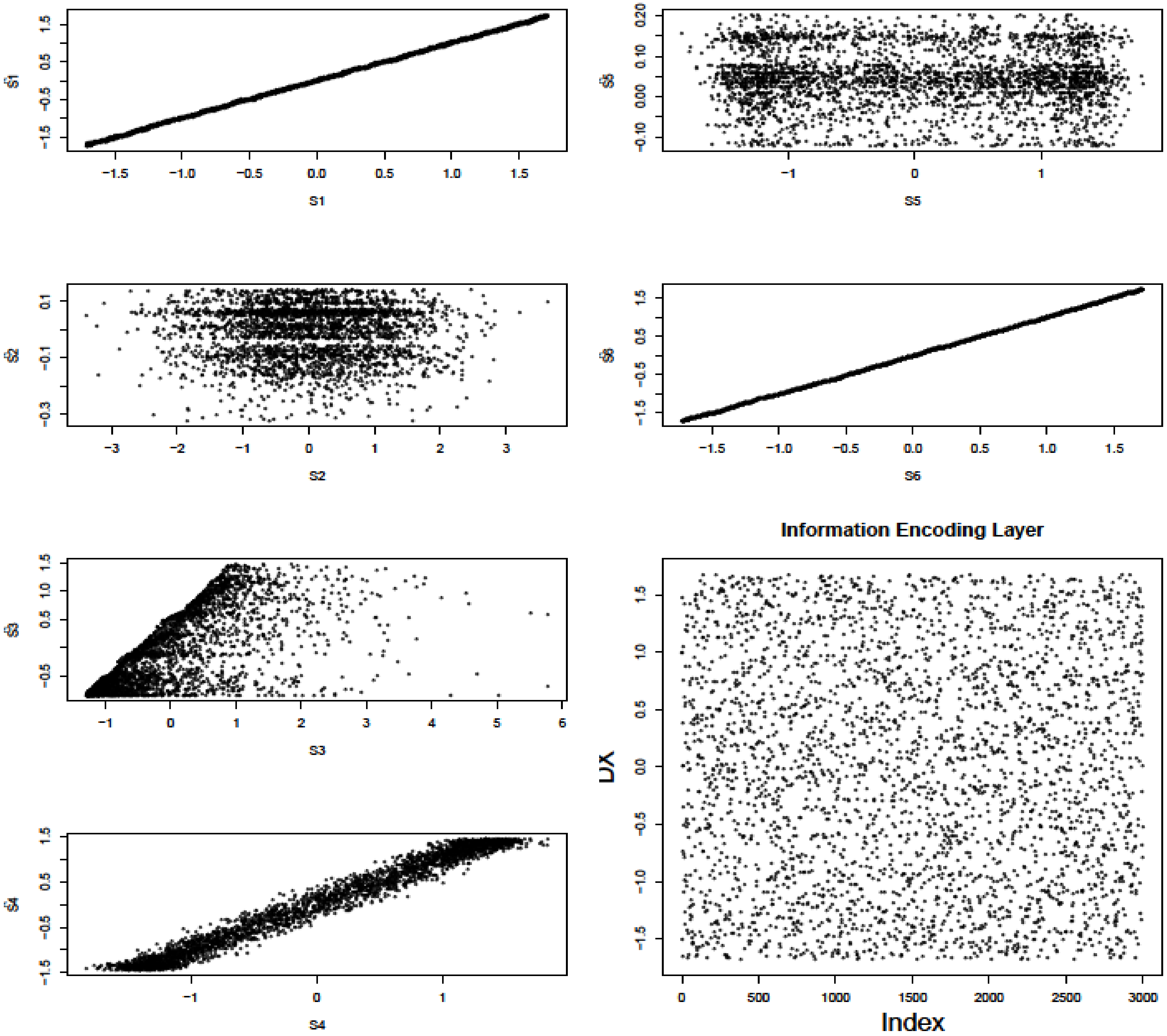
$$d(\mathbf{W}; c, r) = c \cdot \sum_j \prod_i |w_{i,j} \cdot \delta_j| + r \cdot \sum_i \prod_j |w_{i,j} \cdot \delta_j|$$

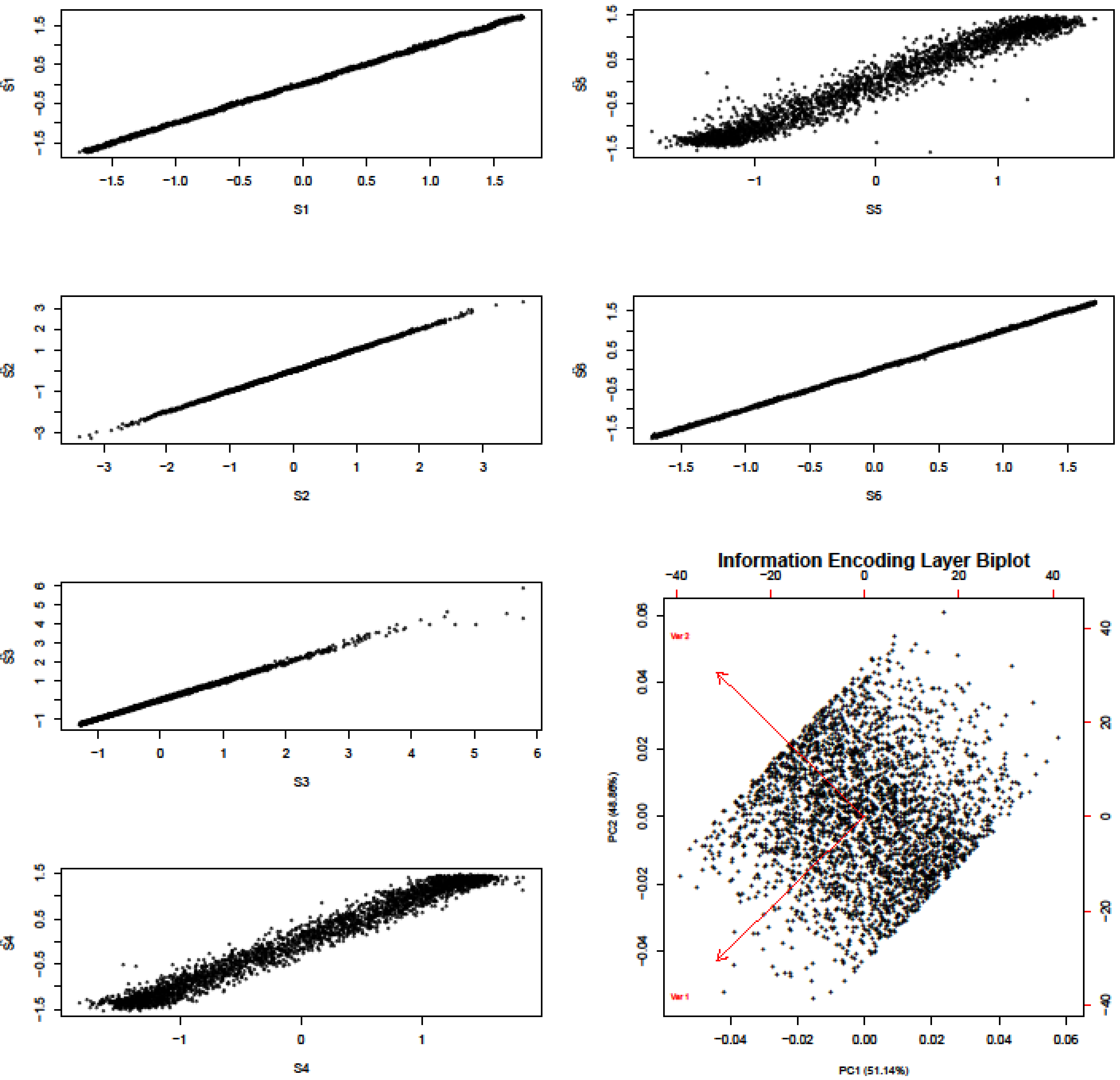


SELECTION AUTOENCODER K=3

	1	2	3
1	0.0000	0.0003	0.0000
2	-0.0000	**-0.3473**	-0.0001
3	-0.0002	0.0002	0.0000
4	**1.0000**	0.0001	-0.0000
5	-0.0001	0.0002	-0.0001
6	-0.0001	0.0002	**-0.1834**

SELECTION AUTOENCODER K=1

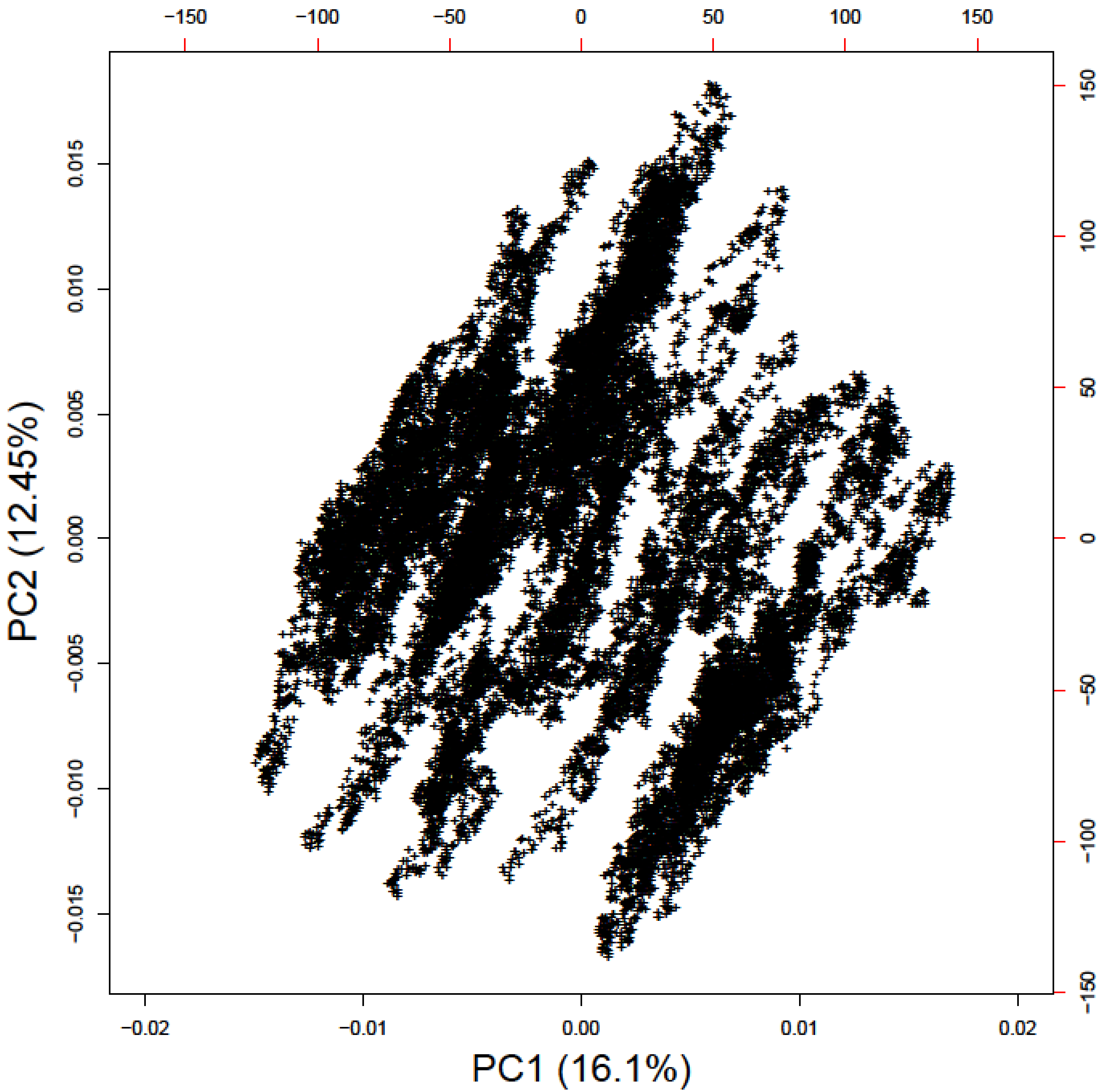




SELECTION AUTOENCODER K=2

	1	2
1	0.0004	-0.0000
2	**0.4023**	0.0000
3	-0.0001	-0.0000
4	-0.0003	-0.0001
5	-0.0000	0.0000
6	0.0004	**-1.0000**

	1	2
1	**0.8943**	0.0001
2	0.0001	**1.0000**
3	0.0006	-0.0002
4	0.0002	-0.0003
5	0.0000	-0.0000
6	0.0001	0.0003



GE CATALYST ENGINE

- 1062 Sensors
- 50 PC explain around 75%
- Autoencoder takes a couple of days to be trained in a laptop and many meta-parameters to be considered.
- Both, the linear and the Autoencoder selection will be tested and compared with predictive non-linear approach.

Q & A

STOCHASTIC FLIGHT DATA GENERATION

ACCOUNTING FOR FAULT BEHAVIOR