2018 European Control Conference (ECC)
June 12-15, 2018. Limassol, Cyprus

# Machine Learning-based Laundry Weight Estimation for Vertical Axis Washing Machines

Gian Antonio Susto, *Member, IEEE,*, Marco Maggipinto, Giulia Zannon,
Fabio Altinier, Elena Pesavento, Alessandro Beghi, *Senior Member, IEEE*

*Abstract*— **In laundry treatment appliances, the weight of the laundry loaded by the user inside the drum dramatically affects the operating behavior. Therefore, it is important to obtain a good estimate of the said quantity in order to correctly configure the machine before the washing/drying starts. In Vertical Axis Washing Machines the laundry weight is computed by exploiting the quantity of water absorbed by the clothes. However, such approach does not grant accurate results because the water absorption depends on the clothes fabric. For this reason, we propose a Soft Sensing approach for weight estimation that exploits the information obtained from physical sensors available on board without added costs. Data-driven Soft Sensors are developed, where, using Machine Learning techniques, a statistical model of the phenomenon of interest is created from a set of sample data.**

## I. Introduction

Fabric care home appliances have reached a very broad diffusion, with more than 90% of households in Western Europe and North America being equipped with washing machine and more than 40% in the rest of the world [1]. Home appliances fabric care manufactures are continuously working to find innovative solutions and technologies with the aim of increasing the competitiveness and appeal on the market of their products. The decision process of a customer interested in purchasing a laundry treatment machine is extremely influenced by its characteristics in terms of water/energy consumption. In Washing Machines (WM) a good estimate of the laundry weight would positively affect the running costs and also the user experience and washing performances, by enabling an effective optimization of the washing cycle.

Unfortunately, laundry weight sensors often are not a cost-effective solution. The weight of the laundry can also be estimated by using the amount of water absorbed by the clothes or be provided by the user before starting the machine. Both these methods do not provide an accurate estimate for the following reasons:

- The amount of water absorbed depends on the fabric (eg. synthetic, cotton);
- The user is not able to correctly perceive the weight and its estimate is dramatically influenced by the volume of the laundry.

The aforementioned methods also have drawbacks that are not related with the accuracy of the estimation. On the one hand, the amount of loaded water is not adapted to the actual laundry weight, leading to an undesired waste of resources. On the other hand, the user may prefer not to directly provide the required information on the laundry weight to the machine. An alternative approach is the use of a *Soft Sensor* (SS) for laundry weight estimation. A SS is a technology that exploits the information provided by a set of available data (i.e. physical sensors or process/equipment information) to infer the value of a desired quantity, that may be unmeasurable or costly to be measured, without added costs. In the situation at hand, the SS estimates can be exploited to improve the washing cycles in terms of: (i) water/energy consumption (ii) washing quality and duration; (iii) user experience.

We can discern two different categories of SSs [2]:

- *model-driven* - they create a model of the phenomenon under investigation based on the knowledge of the physical laws that govern it;
- *data-driven* - they create a statistical model of such phenomenon based on a set of historical data.

In the problem at hand, sensor data are influenced by the load weight once the WM is in motion. However, even simple drum movements are not suitable for model-based approaches for two main reasons:(i) the complexity of the underlying problem to be described, where electrical, mechanical and fluid dynamics elements should be included; (ii) the variability of the load motion inside the drum. Due to the complexity of the process at hand, in this work a data-driven approach is adopted.

Data-driven approaches are typically based on *Machine Learning* (ML) algorithms that, using a set of sample data whose output is known, aim at creating a predictive model that, given a new input, is able to infer the value of the output (the laundry weight in the application considered here) with an error that can be statistically characterized. ML techniques have been adopted in a vast range of industrial applications, such as semiconductor manufacturing [3], [4] under the name of 'Virtual Metrology', chemical processes monitoring [5] under the name of 'Soft Sensors', and automotive [6], [7] industry under the name of 'Virtual Sensors'.

To the best of the authors knowledge, this paper represents the first example of ML-based SS for vertical axis-WMs. In the related literature, various methods for load weight estimation in WMs have been proposed. In [8] an estimate of the normalized moment of load inertia is obtained by computing the difference in acceleration of the drum when two different torque values are applied; such solution is based on the availability of a measure of the moment of inertia with empty drum and then estimated by using a lookup table. A different approach is shown in [9] where the goal is to detect the presence of an unbalanced load and to estimate its weight. In this case the suspensions elongation/compression is measured and then the weight is derived via the Newton's equations. In the related literature, ML based SSs are not a common choice in fabric care appliances due to the difficulties in collecting an accurate dataset of a suitable size and because the implementation in a resource constrained environment, as the firmware of a WM, is usually problematic. For these reasons, our study is limited to simple algorithms that do not require large datasets; we then adopted sparse *regularization* techniques [10] in order to reduce the final model size, easing the implementation and preventing overfitting.

Two models will be evaluated: a forst model that performs a weight estimation at the beginning of the cycle with the laundry still dry and a second model that performs the estimation after the water is loaded inside the WM.

The rest of the paper is organized as follows: Section II is dedicated to an high-level description of vertical axis WM, while Section III is devoted to illustrate the ML methodologies employed in this work. In Section IV the SS algorithm is detailed, whose performances in the available use case are reported in Section V. Finally, concluding remarks and future developments of this work are discussed in Section VI.

## II. Use case

Washing Machines are a common household appliances. It is possible to distinguish two different kinds of WMs: Vertical Axis and Horizontal axis. The first category is characterized by the vertical orientation of the drum rotational axis and is usually popular in the American, Asian, and Australian markets. The second category, instead, is common in the European market and has the drum oriented horizontally. Different orientations of the rotational axis obviously dramatically change how the gravity force affects the washing behavior, hence, it is difficult to create a comprehensive solution for weight estimation. For this reason our work is focused only on Vertical Axis Washing Machines (VA-WM). In VA-WMs the user loads the laundry into the drum from an opening on top of the machine. Then, the detergent can either be placed in a detergent tray or directly over the clothes as in the products used in this work. Once the machine is started there is an initial phase where the dry clothes are uniformly distributed inside the drum. Once the load is well balanced, the water is loaded and the washing cycle starts.

The washing is performed via an *agitator* placed at the base of the drum that is connected to an electric motor. The agitator rapidly rotates clockwise and counterclockwise, moving the clothes and pushing the water inside the fabric; then, the friction between garments removes the dirt. The agitator can assume two different shapes: the Traditional Agitator extends over the entire height of the drum, the Low Profile Agitator (LPA) instead has a limited height and it requires a smaller amount of water to obtain the same washing performances. For this reason the producers usually adopt the second technology. It is extremely important for the correct functioning of the machine that the agitator can move freely and that the clothes do not get stuck in the gap between the agitator and the drum, to avoid damages to both the motor and the clothes. The water loaded inside the drum plays a key role in preventing this situation to occur and it must allow the clothes to be fully immersed. For this reason, a careful adaptation of the water amount to the quantity of laundry present is required; a precise weight estimator is thus fundamental in these machines in order to sharply adapt the cycle without wasting resources. The agitator/motor speed is controlled in a Motor Control Board (MCB) where the firmware of the WM is implemented and where the algorithm proposed in this work is intended to reside. The limited computational resources available on the MCB are a constraint on the complexity of the weight estimation algorithms that can be implemented. For this reason, it is proposed to use sparse regularization methods [10], as described in the following Section.

## III. Methodologies

### A. Elements of Machine Learning

The goal of a SS is to exploit 'free' measures/information to infer the value of a desired quantity (the laundry weight in our work) which otherwise would be too costly or impossible to be obtained. To achieve such goal, estimation algorithms are realized with the aid of ML tools to create a statistical models of the phenomena of interest from a set of sample data. A ML model takes as input a set of values called *predictors* or *features* [11], that are organized in a matrix $X \in \mathbb{R}^{n \times p}$, called *design-matrix*. More in details, the design matrix is composed by $n$ *observations*, that are the examples available in the dataset; the generic $i^{th}$ observation is a row vector $x^{(i)}$ of dimension $p \in \mathbb{N}^{+}$, where the values of $p$ variables, called *features*, that describe such observation, are collected in an ordered way (the $j$-th element of the corresponds to the $j$-th features). SS problems belong to the class of *Supervised Learning* (SL) problems. In SL settings, a set of *labeled* data is available, which means that a label, called *output* or *response*, is associated to each observation. The output values are organized in a column vector: $Y = [y^{(1)T}, y^{(2)T}, \dots, y^{(n)T}]^T$, with the generic $y^{(j)} \in \mathbb{S}^m$, where $S$ is a generic set; in SS problems, $m$ is typically equals to 1. The goal of SL is to find a function $f(\cdot)$ that well describes the phenomenon under consideration, by minimizing a prediction error on the available historical data; the predicted value for a new input $x_{\text{new}}$ is $\hat{y} = f(x_{\text{new}})$. A generic historical dataset can be expressed in mathematical terms as: $\mathcal{D} = \left\{(x^{(i)}, y^{(i)})\; i = 1, \dots, n\right\}$.

SL can be categorized [11] depending on the characteristics of the output; in particular, a *regression* problem arises when the output can assume continuous values ($S = \mathbb{R}$). On the other hand, when the output assumes categorical values, that is, $y^{(i)} \in \{0, 1, \dots, K\}\; i = 1, \dots, n$ where $K$ is the number of classes which the output can belong to, the SL problem can be categorized as a *classification* one. In this work, since the desired output is the weight of the laundry considered as a continuous value in the range $[0, 10]$ kg, we are dealing with a regression problem[1], where the input data will be a set of 'informative', ad-hoc designed, quantities extracted from the signals sampled from the sensors available on board.

### B. Linear regression and Regularization

In SL the prediction model is expressed in the form of a function $y = f(x)$ providing a very general approach where the function $f$ can assume any form. In practice, some constraints on the structure of $f$ must be imposed in order to deal with a treatable problem and to have models with a desirable form. A common choice for regression is to assume $f$ to be linear with respect to the input features:

$$f(x) = \theta^T x, \tag{1}$$

where $\theta$ is a $p + 1$ dimensional vector and $x_0 = 1$ in order to include an intercept term.

The regression algorithm with the linear assumption on the function $f$ is called *linear regression* and the estimation of the optimal function is reduced to compute the value of the parameters vector $\hat{\theta}$ that minimizes the Residual Sum of Squares (RSS) on the available dataset :

$$\hat{\theta} = \underset{\theta}{argmin}\, \frac{1}{2}||X\theta - Y||^2. \tag{2}$$

That has a closed form solution:

$$\hat{\theta} = (X^T X)^{-1} X^T Y \tag{3}$$

The estimate of the parameters obtained in (3) is the classical Ordinary Least Squares (OLS) estimation. It is well-know how the OLS algorithm can be affected by numerical instability problems because the matrix $X^T X$ could be ill-conditioned; moreover, the final OLS solution can suffer from overfitting. For this reason, regularization techniques are employed where a complexity penalty $R$, is added to the cost function that penalizes the solutions with an high norm of the parameters vector.The cost function of a regularized linear regression can thus be expressed as follows:

$$J(\theta) = RSS(\theta) + \lambda R(\theta), \tag{4}$$

where $R(\theta)$ is the regularization penalty and $\lambda \geq 0$ is an hyper-parameter used to tune the trade-off between accuracy on the training data and model complexity. Usually the optimal value of the hyper-parameter is chosen via cross-validation methods on a validation set.

The most popular regularization functions that are widely adopted in the ML community and extensively studied in related literature are:

- *Ridge Regression (RR)* [12]: it employs a $L_2$ norm of the parameters vector as regularization function, hence, $R(\theta) = \sum_{i=1}^{p} |\theta_i|^2$.
- *Least Absolute Shrinkage and Selection Operator (LASSO)* [13]: it employs a $L_1$ norm of the parameters vector as regularization function, hence, $R(\theta) = \sum_{i=1}^{p} |\theta_i|$.

In the fabric care context, where the simplicity of the solution is a key aspect due to the limited computational resources available, the LASSO is a suitable choice since it provides a sparse [13] solution that is easily implementable.

## IV. Proposed Algorithm

The initial part of a VA-WM washing cycle is composed by three phases:

- Warm Up - the washing machine performs some movements in order to distribute the load, still dry, inside the drum;
- Water Loading - during this phase the water is loaded inside the drum;
- Agitation - the WM performs a series of fast movements, called strokes, in order to wash the clothes.

In the following, we will propose two different SS: one, after the Warm Up phase, for 'fast' load estimation and one, after the Agitation phase, for, in principle, more precise estimation given that such SS can exploit more information. The Water loading phase has not been used because it is affected by the water pressure of the plumbing system that may vary from home to home, providing therefore information that can hardly generalize all the real-world scenarios.

In order to train and evaluate the performances of the desired statistical models, a historical dataset has been collected. The dataset is composed by *n = 210* laboratory tests, each one containing a set of signals sampled from the sensors available on-board; each test is composed by different clothes typologies with weights varying from 0 kg to 10 kg.

Among the available signals, the time evolution of the drum speed and torque have proved useful for our purpose, since they are directly related to the different inertial momentum that characterize different weights.

### A. Warm Up model

During the Warm Up phase, the drum speed and torque follow a reference signal that the control algorithm provides, characterized by four repetitions of the same pattern, that from now on we will refer to as *commutations*; a reference signal for the drum speed during the Warm Up phase is depicted in Fig. 1, where commutations are separated by red dots.

Due to the constraints imposed by the limited amount of resources on board, it was unfeasible to employ complex algorithms for automatic features extraction [14] or to feed the raw signals directly as input to the estimator. For this reason, a set of hard-coded features has been extracted and organized in a design matrix, which has then been fed to the learning algorithm. Because of the same behavior of the reference signals in the warm up phase, the same features for each commutation has been extracted. A total of 20 features from the Warm Up phase is considered.

[1] It will be shown in Section V how the data available and the performance evaluation will be provided w.r.t. a set of 11 weight levels (weight values from 0 to 10 kg, with 1 kg of distance), similarly to a classification problem. However, given that the desired SS will be employed in any weight condition, a regression approach it is more suitable.

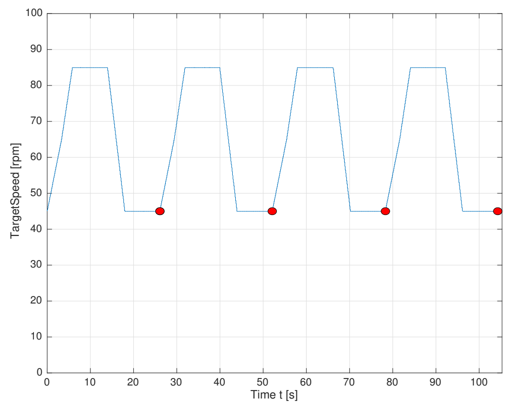


Fig. 1. Reference signal for the drum speed during the Warm Up phase. The red dots separate the four commutations.

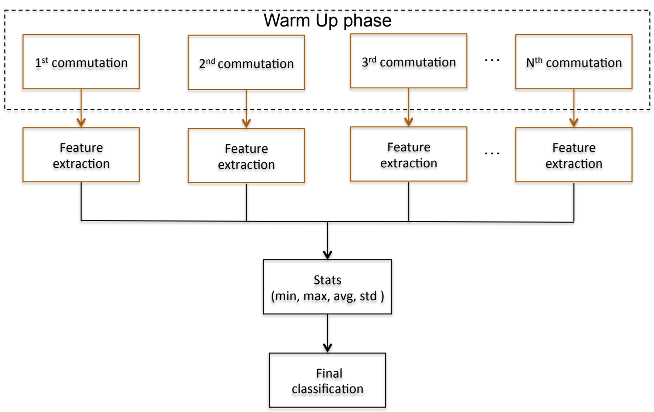


Fig. 2. Classification scheme for the Warm Up model using statistics (*WU Stats*).

In order to train the desired models, we adopted the LASSO regression for the reasons explained in Section III, considering the weight as a continuous variable. Then, we rounded the obtained estimation to the nearest integer (in a classification problem fashion), because a decimal precision is not required in this context.

The commutation movements allow to have different patterns on the sensor data due to the laundry weight and its arrangement inside the drum: while the first cause is actually the quantity we are aiming to describe, the second one can be considered as process noise. Moreover, from experimental analysis, commutations seems to have the same behavior independently from the the number of the commutations, ie. the fact that the commutation is the first, second, third or fourth in the washing process. For these reasons, it doesn't make much sense to have a design matrix where we distinguish the commutation number; instead the features extracted from each single commutation can than be treated in two alternative approaches:

- *WU Stats* - a set of statistics over the four commutations can be computed (eg. min, max, average, standard deviation) that are then used to make the weight prediction. Such statistics are the regressors, input variables of our model. This approach is depicted in Fig. 2.
- *WU Single* - this approach is illustrated in Fig. 3: with this solution a model is provided at commutation level; for each of the 4 commutations a weight estimation is provided using only the information provided by that commutation. After that a global estimate is provided as the mode over the 4 estimations. Some quantitative considerations, outside the scope of this paper, are applied to handle ties.

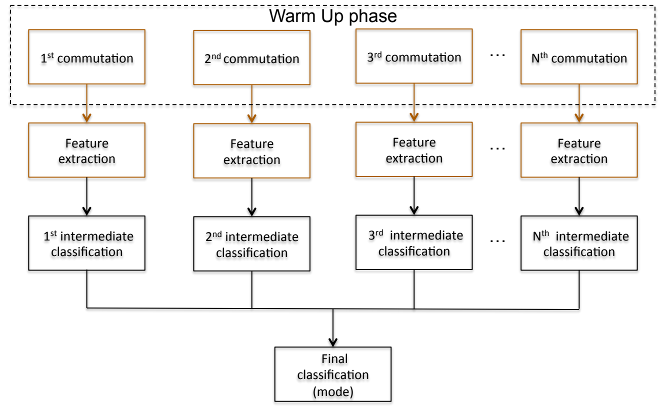


Fig. 3. Classification scheme for the Warm Up model using single commutations. *WU Single*.

## B. Agitation model

As explained before, once the Warm Up phase is completed the WM proceeds with the water loading and the agitation phases. As said, since the water loading phase cannot be used for this modeling task, we extracted a set of hand-designed features from the agitation phase that have then be used, in combination with the ones of the Warm Up, in order to develop a 'late' weight classifier with the aim of sharpening the previous prediction. Due of the increased size of the features set, it is even more important than before to perform a feature selection exploiting the LASSO regularization.

Similarly to what has been done for the Warm Up model, two alternative approaches are considered:

- *AG Stats* - where statistics over the four commutations are employed (the same statistics used in WU Stats) in conjunction with the features extracted from the Agitation phase.
- *AG Single* - where, similarly to what was don in the this WU Single approach, 4 different estimations are provided at commutation level based on the features computed in that commutation and on the agitation phase, that is common for all 4 commutations. Also in this case, a global estimate is provided as the mode over the 4 estimations and some quantitative considerations are applied to handle ties.

The two solutions are represented in Fig. 5 and Fig. 4.

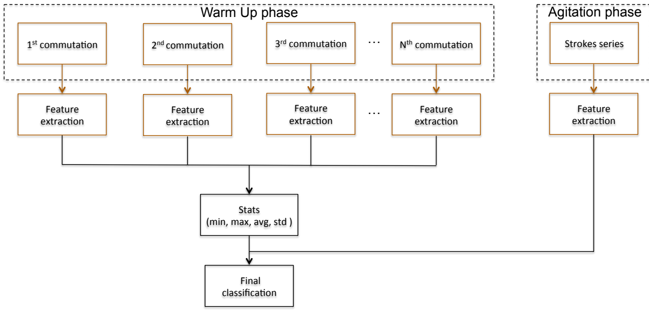


Fig. 4. Classification scheme for the Agitation model using statistics. *AG Stats*

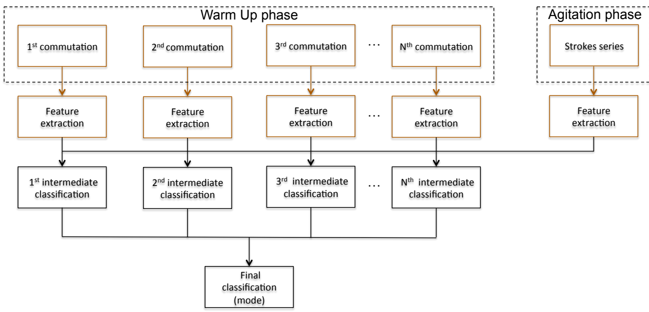


Fig. 5. Classification scheme for the Agitation model using single commutations. *AG Single*

## V. Results

In this Section, the experimental results obtained with the models described in the previous Section are reported. The performances have been estimated using a Monte Carlo (MC) cross validation [15] with $k_{\text{OUT}} = 100$ outer cycle iterations (for performance evaluation) and $k_{\text{IN}} = 100$ nested cycle iterations (for hyper-parameter tuning) in the inner validation loop. As a measure of the quality of the models we use the classification rate which is defined as follows: given a test set $\{(x_i, y_i)\ i = 1, \dots, n_{test}\}$, we compute the estimated output $\hat{y}_i = f(x_i)$ for every $i = 1, \dots, n_{test}$ using a trained model. Let $n_{match}$ be the number of input values where $\hat{y}_i = y_i$; the classification rate can be computed as:

$$CR = \frac{n_{match}}{n_{test}} \cdot 100 \tag{5}$$

A confusion matrix is also reported for each approach where, on the main diagonal of the matrix, the classification rate is indicated for each weight class.

In Fig. 6 the confusion matrices for the *WU Stats* and *WU Single* classification schemes are reported. The *WU Single* scheme outperforms the *WU Stats* one with classification rates equal to 63,49% on average, while the *WU Stats* solution achieves only the 42,92% on average classification rate. Moreover, the *WU Single* solution provides more precise estimations as it can be clearly appreciated, as the colored squares are better centered around the main diagonal of the matrix than the ones obtained with the *WU Stats* solution.

The performance of the Agitation models are reported in Fig. 7, with, respectively, the confusion matrix of the *AG Stats* model reported on the left panel and the confusion matrix of the *AG Single* model in the right panel. The Agitation model is fed with a higher number of features that provide additional information; this allows both the classification schemes to perform better compared to the Warm Up model. The scheme based on single commutation classification, *AG Single*, outperforms the one based on commutations statistics, *AG Stats*. Both confusion matrices present a decreasing trend of the performance w.r.t. the weight, confirming the intuition presented in Section IV that the classification task is more difficult at high loads. In Table I the performances of the implemented classification schemes are summarized. The most important performance indicators are the classification rate and the percentage of correctly classified observations. It can be appreciated that the AG Single approach outperforms all the other methods in terms of classification accuracy and the difference between the AG Single and AG Stats is negligible in terms of the other performance indicators. Consistently, for faster SS solutions, the WU Single outperforms the WU Stats approach. To complete the performance analysis, we compare the model sizes of the AG Single approach for the Ridge and LASSO models in Table II. It is possible to notice how the LASSO reduces of more than a half the complexity of the model.

TABLE I

Performance comparison of the implemented classification schemes Averaged results over $k_{\text{OUT}} = 100$ MC simulations.

| Classification scheme | correctly classified tries [%] | % of tries within a 1kg error | % of tries within a 2kg error | Max Error [kg] |
|---|---|---|---|---|
| *WU Stats* | 42,92 | 88,59 | 98,62 | 6 |
| *WU Single* | 63,49 | 96,79 | 98,40 | 6 |
| *AG Stats* | 63,29 | **99,22** | **100,00** | **2** |
| *AG Single* | **73,61** | 99,00 | 99,98 | 3 |

TABLE II

Model size comparison for the two regularized methods. Averaged results over $k_{\text{OUT}} = 100$ MC simulations.

| Method | Average model size |
|---|---|
| LASSO | 23,32 |
| Ridge | 50 |

## VI. Conclusion

In this paper, a novel Soft Sensing approach for load weight estimation in VA-WMs is proposed. The algorithms have been tested on real industrial data, obtaining satisfactory results despite the simplicity of the solution.

Agitation models exhibit better performance with respect to Warm Up models with both the classification schemes thanks to the availability of more information to perform the prediction. However, Warm Up models still achieve acceptable performance, therefore, they may be employed for a first weight estimation after a short amount of time from the beginning of the washing cycle, before water is added to the

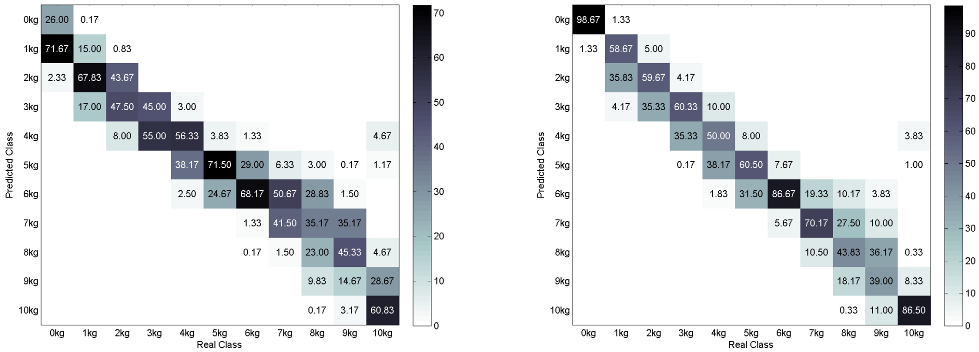


Fig. 6. Confusion matrices for the *WU Stats* model (left panel) and for the *WU Single* model (right panel). Averaged results over 100 MC simulations.

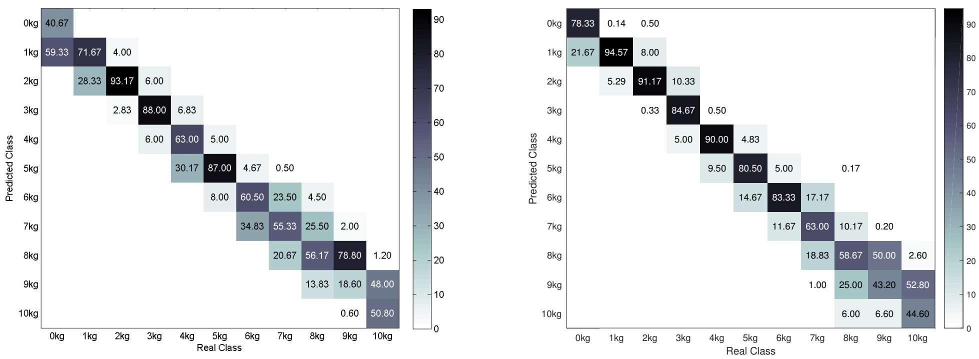


Fig. 7. Confusion matrices for the *AG Stats* model (left panel) and for the *AG Single* model (right panel). Averaged results over 100 MC simulations.

process. Such information may be used to load a suitable water level for the given laundry load, thus limiting the environmental impact of the product while still guaranteeing a proper washing cycle. Moreover, the choice of using each single WU commutation as a test outperforms the usage of statistical metrics of the features.

The proposed algorithms exploit LASSO sparsity to provide a simple and interpretable solution, that can be implemented using the limited computational resources available on the WM. If more performing processors and RAM are available, more sophisticated ML techniques could be used, that can automatically extract some useful features from the available signals and can be directly fed with the raw signals. Such models could avoid the problems caused by manually selected features that are subject to errors in the extraction phase and do not provide a scalable solution.

## References


[1] Electrolux, "Annual report 2016," 2016. [Online]. Available: http://www.electroluxgroup.com/en/electrolux-annual-report-2016-23326/

[2] P. Kadlec, B. Gabrys, and S. Strandt, "Data-driven soft sensors in the process industry," *Computers & Chemical Engineering*, vol. 33, no. 4, pp. 795 – 814, 2009.

[3] G. A. Susto and A. Beghi, "A virtual metrology system based on least angle regression and statistical clustering," *Applied Stochastic Models in Business and Industry*, vol. 29, no. 4, pp. 362–376, 2013.

[4] G. A. Susto, A. Schirru, S. Pampuri, G. De Nicolao, and A. Beghi, "An information-theory and virtual metrology-based approach to run-to-run semiconductor manufacturing control," in *Automation Science and Engineering (CASE), 2012 IEEE International Conference on*. IEEE, 2012, pp. 358–363.

[5] A. J. de Assis and R. Maciel Filho, "Soft sensors development for on-line bioreactor state estimation," *Computers & Chemical Engineering*, vol. 24, no. 2-7, pp. 1099–1103, 2000.

[6] F. Gustafsson, M. Drevo, U. Forssell, M. Lofgr̈en, N. Persson, and H. Quicklund, "Virtual sensors of tire pressure and road friction," in *SAE Technical Paper*. SAE International, 03 2001.

[7] M. Bruschetta, F. Maran, and A. Beghi, "A nonlinear, mpc-based motion cueing algorithm for a high-performance, nine-dof dynamic simulator platform," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 2, pp. 686–694, 2017.

[8] V. Badam, M. Dausch, W. Whipple, R. Hornung, and D. Dickerson, "Energy efficient washer with inertia based method for determining load," Nov. 26 1996, US Patent 5,577,283.

[9] A. Yorukoglu and E. Altug, "Determining the mass and angular position of the unbalanced load in horizontal washing machines," in *2009 IEEE/ASME*, July 2009, pp. 118–123.

[10] B. Schölkopf and A. J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*, 2002.

[11] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2009.

[12] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, pp. 55–67, 1970.

[13] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[14] G. A. Susto, A. Schirru, S. Pampuri, and S. McLoone, "Supervised aggregative feature extraction for big data time series regression," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 3, pp. 1243–1252, 2016.

[15] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Ijcai*. Stanford, CA, 1995, pp. 1137–1145.