Database quality assessment for interactive learning: Application to occupancy estimation

Manar Amayri, Stéphane Ploix, Nizar Bouguila, Frederic Wurtz

Database quality assessment for interactive learning: application to occupancy estimation

Manar Amayri[a], Stephane Ploix[b], Nizar Bouguila[c], Frederic Wurtz[d]

[a]*G-SCOP lab / Grenoble Institute of Technology, Grenoble, France(e-mail: manar.amayri@grenoble-inp.fr)*
[b]*G-SCOP lab / Grenoble Institute of Technology, Grenoble, France (e-mail: stephane.ploix@grenoble-inp.fr)*
[c]*Concordia Institute for Information Systems Engineering (CIISE), Concordia University, Montreal, QC, Canada, (e-mail: nizar.bouguila@concordia.ca)*
[d]*G2ELAB/ Grenoble Institute of Technology, Grenoble, France, (e-mail: frederic.wurtz@g2elab.grenoble-inp.fr)*

Abstract

Data quality assesment is a key component for many real applications, since it can drive better modelling. In this work a methodology to asses data quality (Qscore) is proposed and discussed. The validation of Qscore is performed via an interactive learning experiment related to occupancy estimation. Interactive learning has been shown to be crucial to consider and integrate occupant behavior in smart buildings. Indeed, valuable feedback and information can be collected from the occupants by involving them and by improving their consciousness about energy management systems. Users should feel involved to keep developing highly energy-efficient buildings. To reach this goal, occupants should be aware of the building features to feel more in control. This paper proposes a framework to interact with occupants to estimate building occupancy. This framework is based on an enhanced supervised learning approach that involves interaction with occupants, when necessary, to keep collecting training data. The training data consist of the measurements (i.e. features) collected from common sensors, for instance, motion detection, power consumption, and CO2 concentration, and the label (i.e. number of occupants) provided by the occupants during interactions. The considered learning machine in our experiments is the Multi-layer Perceptron regressor (MLP), although other approaches could be easily integrated within the proposed framework. In order to avoid useless interaction with users a new concept is introduced, called spread rate, to measure the quality of the data to decide if an interaction with the user is necessary or not. Extensive simulations have shown

the merits of the proposed approach.

Keywords:
Data quality, machine leaning, data mining, human behavior, building performance, activities recognition, office buildings.

---

1. Introduction

Future building energy management systems may cover a large set of applications. It should support retrospective analyses of past periods by estimating and correlating actions and variables such as occupancy, usage of appliances, and heat flows through windows for instance. Using simulation, it is possible to extrapolate future states depending on hypothetical controls or by replaying past situations. To develop different applications in smart buildings, energy management systems have to embed knowledge and data models of the living area system; they have to be equipped with learning, estimation, simulation, and optimization capabilities. Because living zones are both related to physics and to occupants, the state characterizing a zone at a given time is also related to occupants such as location of occupants, their activities, actions performed on the envelope configuration (e.g. windows, doors, shuttles), actions performed on the HVAC (Heating, Ventilation and Air-Conditioning) system, and actions performed on other appliances. The human part of the state can be helpful with different regards. For instance, different key performance indicators could be calculated for mirroring analysis. Examples of these indicators include comfort during presence, waste/spare of energy, consumption linked to a specific device, and consumption per activity or person. Usage analysis measuring and estimating what cannot be measured can help occupants to discover costly routines. It can be managed by replaying a past period, displaying both energy impacts and human past behaviors. Occupant behavior modeling can ease the tuning of reactive human behavior. The resulting models can then be co-simulated with physical models to better represent human-physical zone systems. The model should be updated accordingly without losing its flexibility. One of the approaches that could be used to reach this goal is online learning which supports life-long learning (*i.e.* the models could be improved each time new data are added). Unlike supervised learning, online learning, also called incremental learning (Perner, 2003; Duda et al., 2012), is a challenging unsupervised task that has to be done in an online fashion, which imposes constraints on both strategy and efficiency (Zhang et al., 2005). Online learning techniques provide solutions addressing real-time occupancy estimation where the ground truth

is required to build the initial model that we will continously update. Generally, video cameras are used to determine the actual occupancy required for supervised learning (Milenkovic and Amft, 2013), which limits highly the application implementation because of privacy issues. Consequently, an interactive learning approach has been investigated for estimating occupancy with a set of sensors and self-labeling by occupants (Amayri et al., 2016b).

Interactive learning estimates the number of occupants by questioning occupants when relevant, by limiting the number of interactions and maximizing the information usefulness about the actual occupancy. Occupancy estimation algorithms use information collected from occupants together with common sensors. Interactive learning approach depends mainly on the interaction methodology to define when it is necessary to ask occupants. The *ask* is a question displayed on the screen with its order, date and time i.e. (Question1, 05/09/2019 15:42:12 How many occupants in last 30 minutes? (0...7)), while in a response area, there are different options to answer, defined according to a minimum and a maximum possible number of occupants with a timeout of 3 hours for each question.

Three criteria have been taken into account to determine the interaction time. The first one is the density of the neighborhood which is defined as the number of existing records (i.e. vectors of sensor features) in the neighborhood of a potential *ask* (i.e. interaction with the occupant). The second criterion is the classifier estimation error in the neighborhood of the potential *ask* which leads to the concept of neighborhood quality that will be defined later in this paper via a novel concept called spread rate. This methodology is based on the following. If the classifier estimation error is too high for a record, this record is removed from the neighborhood. However, an acceptable estimation error leads to updating the training set with the new record. The third criterion is based on the minimum class weight which consists of the minimum acceptable number of records for each class (see figure 1). Spread rate replaces the density of neighborhood. It moves from a local criterion to a global one (instead of counting the records, it checks how records are globally distributed).

The main goal of this paper is to investigate further the concept of interactive learning by introducing the concept of spread-rate which is basically a global measure of data quality. The primary objective of the introduced spread-rate technique is reducing the number of interactions with occupants by considering the whole database space instead of a local neighborhood.

The rest of this paper is organized as follows. Section 2 presents a review of the research about occupancy estimation. Section 3 investigates the proposed spread rate methodology. Section 4 discusses interactive learning for occupancy esti-

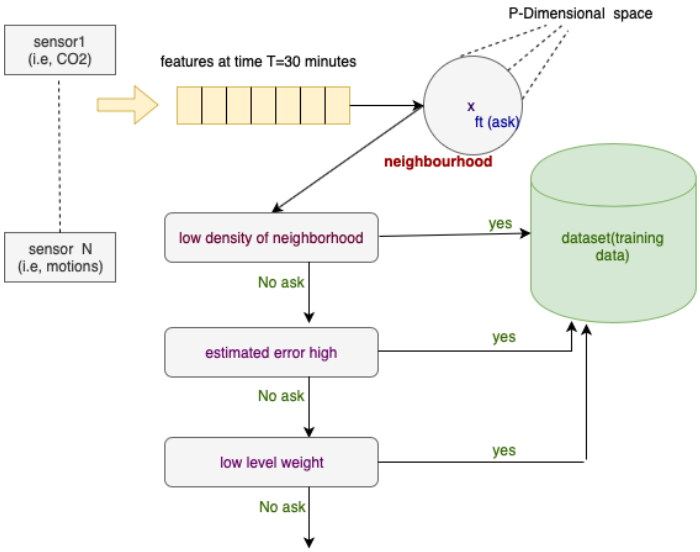

Figure 1: Interactive learning process with neighborhood approach

mation. Section 5 points out an implementation and experimentation with the interactive learning application and section 6 concludes the paper.

2. State of the art

Existing methods for occupancy estimation depend on the data sources. (Jin et al., 2018) proposed an occupancy detection using sensing by proxy, where the inference depends on proxy measurements such as CO2 concentration, and indoor temperature. In (Minor et al., 2017) the authors investigated a solution for occupancy and activity prediction (eating, sleeping, and taking medicine) by imitation learning and reduce it to a simple regression problem. Moreover, data coming from sensors have been used in this work (i.e. location sensors, window, and position). Numerous studies in smart homes have used sound processing for activities recognition such as (Sehili et al., 2012) which used Gaussian Mixtures Model (GMM) and Support Vector Machine (SVM), in order to classify sound data sequences in order to be used in elderly people monitoring systems. In (Valle, 2016) the author has proposed an algorithm for audio-based occupancy analysis, which depends on GMM and Hidden Markov Model (HMM). Moreover, (Ordonez and Roggen, 2016) proposed an action recognition approach based on Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). This model

is suitable in case of using wearable sensors. Besides, it does not require expert knowledge to design the required features but still suffers from the drawback that deep learning algorithms need a large quantity of training data. In (Amayri et al., 2016a) supervised learning approach is investigated, it starts by determining the common sensors that shall be used to estimate and classify the approximate number of people (within a range) in a room and their activities. Means to estimate occupancy include motion detection, power consumption, CO2 concentration sensors, microphone or door/window positions. It starts by determining the most useful measurements in calculating the information gains and removes the ones which, when added to the classification algorithm, make no difference to the overall output. Then, estimation algorithms are proposed: they rely on decision tree and random forest learning algorithms because they yield decision rules readable by humans, which corresponds to nested if-then-else rules, where thresholds can be adjusted depending on the considered living areas. An office has been used for testing and 2 video cameras have been deployed in this approach. This limits highly the application implementation because of privacy issues. Moreover, in (Amayri et al., 2019b) a knowledge-based approach using sensor data and knowledge coming respectively from observations and questionnaires has been proposed. It relies on a hidden Markov model and Bayesian network algorithms to model human behavior with probabilistic cause-effect relations and states based on knowledge and questionnaire. Different applications have been studied for validation: an office, an apartment and a house with different levels of complexity according to their context, available sensors, occupancy or activities feedbacks, the complexity of the environment, etc. Better results have been obtained from the Bayesian network as compared with Hidden Markov model by taking into account the relations between the state's model. Using knowledge domain and questionnaire with data sensors in unsupervised learning method is more flexible and open for different types of applications, with an acceptable average error for occupancy estimation. In addition, avoiding the use of video cameras has been achieved. This knowledge-based approach can be used widely in different contexts but may lead to poor performance during some periods (Wang and Liu, 2011; Wang and Shi, 2009).

In (Amayri et al., 2019a), an interactive learning approach is proposed to estimate the number of occupants in a room by questioning occupants when relevant, by limiting the number of interactions and maximizing the information gains, about the actual occupancy. The density of the neighborhood, average error estimation, and the weight of each class are used to define the valuable time to interact with the end users. The results lead to the conclusion that the interactive approach is more

efficient for occupancy estimation than the other methods taking into account the context. Active learning is an important tool for many real-time applications. The main idea behind active learning is that a machine learning algorithm can perform higher accuracy with fewer training labels if it is permitted to determine the data from which it learns, (Tong and Chang, 2001), (Quionero-Candela et al., 2009). The main difference between active learning and our approach is that interactive learning has no ground truth to build the model before interacting with the user. In general, the accuracy of the training database is the main important factor to build an accurate model. The problem of taking into account the quality of the training data was rarely discussed. In fact, as the available information is dynamic and changes over time, the structure of the training data should be readjusted to deal with such dynamic aspects. In (Ardgna et al., 2018), the authors have evaluated big data quality using different factors: accuracy, completeness, consistency, distinctness, precision, timeliness, and volume. An existing data quality method has been proposed in the case of classification in order to avoid having overlapping classes (Wang et al., 2009) and (Wang and Shi, 2009). This concept of data quality is different than the one which is proposed in this paper. Spread rate considers the global space of the data and does not look at each class alone.

3. Spread rate principle

3.1. Problem statement

Let's define an $ask(t)$, called simply point in the following, by a list of feature values coming from various sensors $ask(t) = P(t) = (f_1(t), \ldots, f_p(t))$ related to a living place at the same time t, where $f_i(t) \in (\check{f}_i, \hat{f}_i)$. Let's define a label $l(t)$ as a user feedback corresponding to a number or a text label. A record is defined as an ask and a label related to the same time: $r(t) = (ask(t), l(t))$. An ask $ask(t)$ for which there is no corresponding label $l(t)$ is called a candidate ask because no label has yet been collected from user feedback. Conversely, when $l(t)$ exists, $ask(t)$ is called a "recorded" ask. A database $\mathcal{P}_r$ is a collection of records collected at different times whereas an ask database $\mathcal{P}$ is a collection of recorded asks. In the next, "database" will refer to recorded ask-database and a point will refer to a p-dimensional ask having value in a p-dimensional feature subspace $\mathcal{F}$. Generally speaking, let $\mathcal{P} = \{P_1, \ldots, P_n\}$ be a normalized ask-database of n p-dimensional points (asks) distributed in a p-dimensional normalized space $\mathcal{S} = (0, 1)^p$.

Consider a set of asks from a database $\mathcal{P}$ distributed in a non-normalized feature subspace $\mathcal{F} = [\check{f}_1, \hat{f}_1] \times \cdots \times [\check{f}_p, \hat{f}_p]$. The corresponding normalized database is

given by $\mathcal{P}'$ with $\forall i \in \{1,\ldots,n\}, \forall j \in \{1,\ldots,p\}, (f'_j)_i = \frac{(f_j)_i - \check{f}_j}{\hat{f}_j - \check{f}_j}$.
The main problem in interactive learning is to determine when a candidate ask P should be considered for collecting occupant feedback considering an existing database $\mathcal{P}$. Let's assume that $score(\mathcal{P})$ assesses the quality of the database $\mathcal{P}$. A new point P should be considered for an ask if $score(\mathcal{P} \cup \{P\}) > score(\mathcal{P})$ where $score(.) \in (0,1)$; 0 for the lowest quality and 1 for the highest. Therefore, the main question of the paper raises: how to assess the quality of a database i.e. how should be the score function?

3.2. Intuitive approach for the quality assessment of a database

Intuitively speaking, it can be said that a database $\mathcal{P}$ of *good* quality means that its points are regularly spread all over the normalized feature subspace $\mathcal{S} = [0,1]^p$. For instance, when all the points are same, the poorest quality is met ($score(\mathcal{P}) = 0$), (see figure 2), but what corresponds to the highest quality? The 2D-patterns in figure 3 should correspond to the highest quality because spreading cannot be improved further.

3.2.1. Perfect spreading

Generally speaking, a perfect spreading $\mathcal{P}^*_{p,\lambda}; \lambda \in \mathbb{N}$ in a p-dimensional space is met when: 1) the number of points $n = (1+\lambda)^p$, and 2) the infinite distance between each point and its closest neighbor is $\frac{1}{\lambda}$.
As a reminder, the infinite-distance is based on the infinite-norm, which is defined by: $\|P\|_\infty = max\left(|P_1|,\ldots,|P_p|\right)$. It is used in this work because it satisfies: $\forall P_i \in (0,1)^p, \forall P_j \in (0,1)^p, \|P_i - P_j\|_\infty \leq 1$.

3.2.2. Definition of spread rate

The spreadrate score (Sscore) measures how much the points P_i of a normalized database regularly cover the space $[0,1]^p$. $\forall p \in \mathbb{N}^*$ and $\forall n \in \mathbb{N}^* \backslash \{1\}$, it is defined as:

$$Sscore(\mathcal{P}_n) = \left(n^{1/p} - 1\right) \times \frac{\sum_{i=1}^{n} \left(min_{j \in \{1,..,n\}} \|P_i - P_j\|_\infty\right)}{n} \qquad (1)$$

The spreadrate score aims at assessing the quality of a $n \times p$ table of features values. It corresponds to the average distance (according to infinity norm to avoid exceeding one) per dimension separeting each point of a normalized database to its closest neighbor (see figure 3). It satisfies: a) Sscore $\in [0,1]$, b) Sscore = 0 if all the points in the normalized database are the same (null distance between all points), c) Sscore = 1 if all the points are perfectly distributed over the normalized

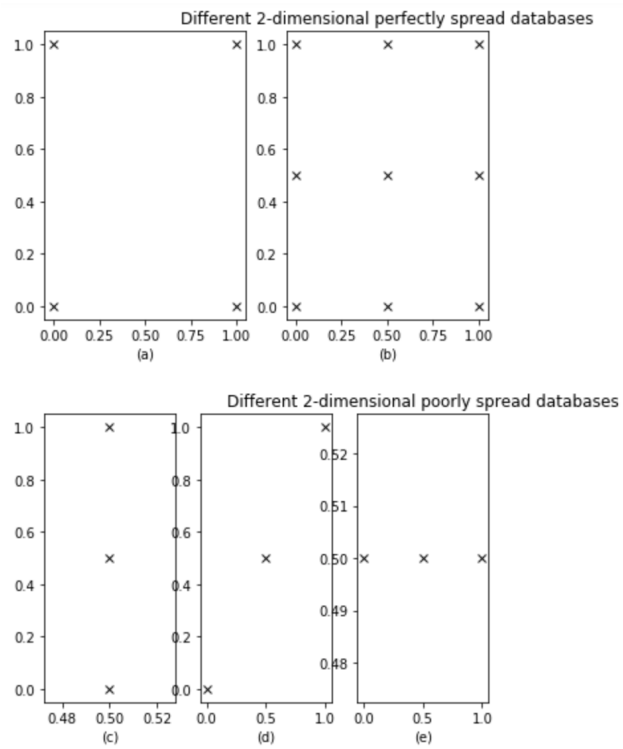

Figure 2: Different 2-dimensional spread databases

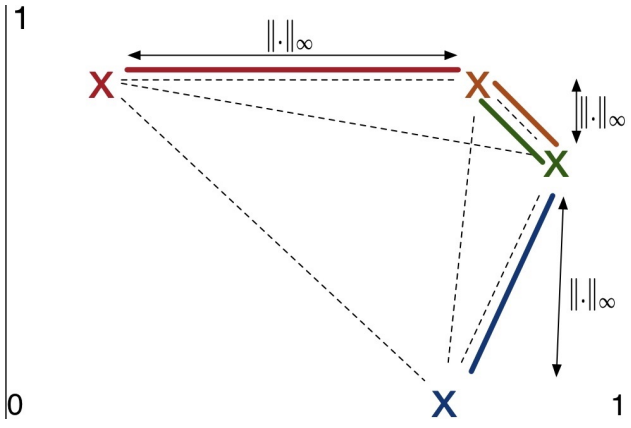

Figure 3: Spread rate definition

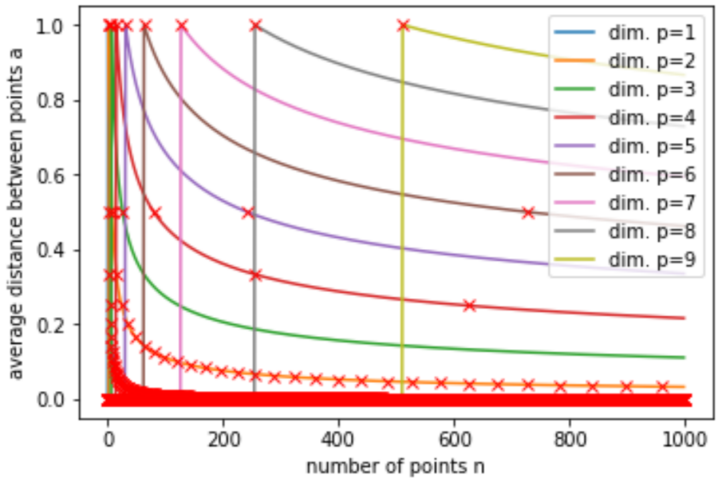

Figure 4: Perfect spreading and interpolated best distances for different dimensions

space $(0,1)^p$. The last property is straightforward to show using the definition of a perfect spreading $\mathcal{P}^*_{p,q}$ presented above.
Because of the infinite distance, the spreading (c) to (e) yields the same spread rate (0.366), (2). Indeed, infinite distance is isotropic. The spread rate score can also be seen as a ratio of:

$$Sscore = \frac{\text{average of each point minimum distances between itself and its closest neighbor}(=a)}{\text{best theoretical distance between 2 points for a perfect distribution } (=a^*)}$$

with $a = \frac{\sum_{i=1}^{n}\left(min_{j\in\{1,..,n\}}\|P_i - P_j\|_\infty\right)}{n}$ and $a^* = \frac{1}{n^{1/p}-1}$.

Let's calculate the best theoretical distance between 2 points for a perfect spreading. As seen before, perfect spreadings are known only when $n^* = (1+\lambda)^p$ with $\lambda \in \mathbb{N}^*$ with at least 2 points per dimension. λ stands for the number of points per dimension. The distance between 2 points in a perfect distribution is equal to $a^* = \frac{1}{\lambda}$. For a given n, the interpolated best theoretical distance between 2 points is given by $a^* = \frac{1}{n^{1/p}-1}$. This distance is exact for n^* only. Figure 4 shows the distance a as a function of n and p ($a(n,p)$).
In order to see how the average distance $a(n,p)$ interpolates the distance between points in perfect spreadings, let's look at different examples in figure 5. According to this figure, it is clear that spreadrate score is not enough because it doesn't take into account the density of points. Let's introduce the expected frequency of

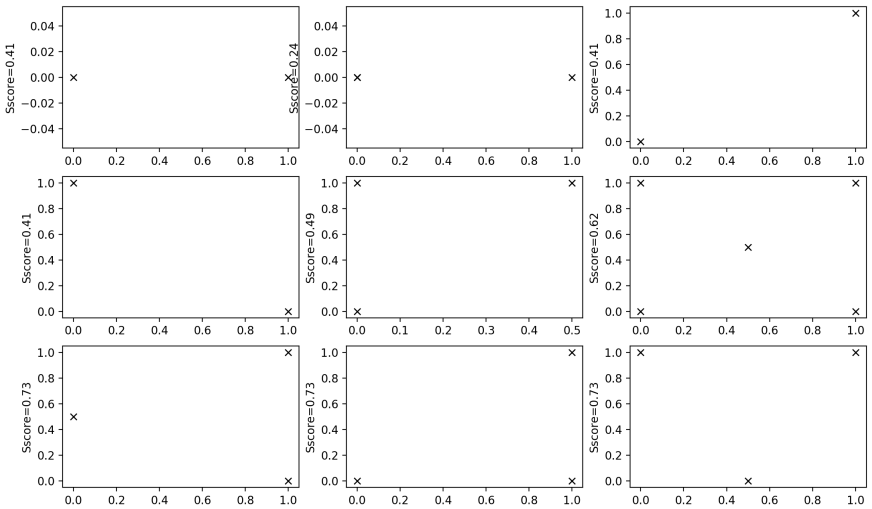

Figure 5: 2-dimensional databases examples

points per dimension f. For a p-dimensional database $\mathcal{P}_n$, the expected number of points is p^f. The accomplishment is: $A = 1 - e^{\frac{-3n}{p^f}}$ with $A \in (0, 1)$; $A = 1$ means expected resolution is met and 0, not at all.
For a normalized database $\mathcal{P}'_n$, the Qscore (in $(0, 1)$) is defined as the product of spreadrate and resolution accomplishment:

$$Qscore(\mathcal{P}'_n, f) = \frac{\left(n^{1/p}-1\right)\left(1-e^{\frac{-3n}{p^f}}\right)}{n} \times \sum_{i=1}^{n} \left(min_{j\in\{1,..,n\}} \|P_i - P_j\|_\infty\right)$$

Best database quality is obtained when Qscore=1, and worst when Qscore=0.

For illustration purposes and ease of representation, let's now focus on a 2-dimensional database with 8 random points and compute the spreadrate scores, see figure 6. The spread rate score changes by changing the location and the number of the points. A genetic algorithm (differential evolution) is used to maximize the spread rate score by adjusting the locations of the 8 points, see figure 7. Let's analyze experimentally the maximum spread rate values depending on the number of points in a database in a 2-dimensional space. Figure 8 shows the location of the points

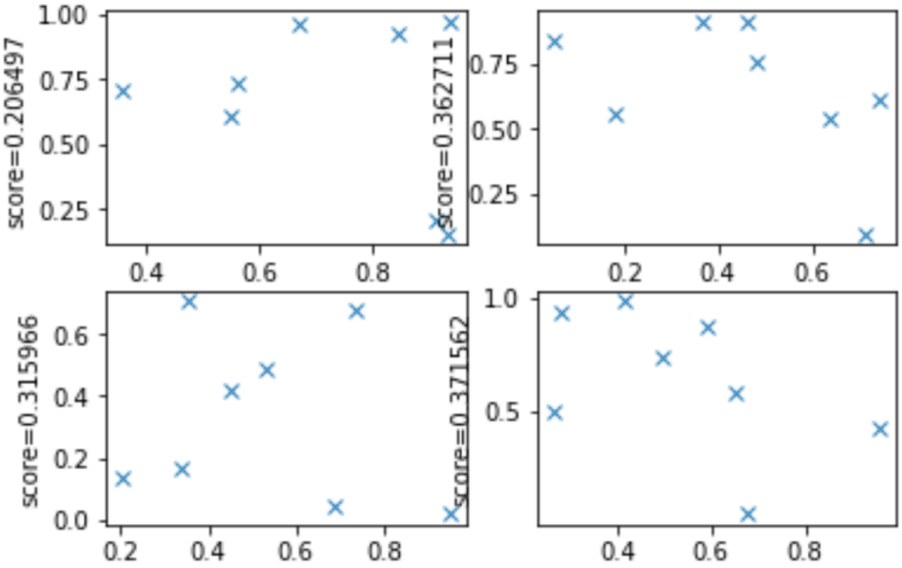

Figure 6: spreadrate for 2-dimensional database with 8 random points

with the best spread rate score (i.e, 4 points give a Sscore=1 while we can not acheieve this value with 7 points). Note that the minimum number of points in the database is: $n_{min} = 2^p$.

3.2.3. Complexity

Complexity is related to the combinations of distances i.e. $O(n) = \binom{n}{2} = \frac{n!}{2(n-2)!} = \frac{n(n-1)}{2} = \propto n^2$

Complexity of the spread rate score calculation does not depend on the dimension p but only on the number of points n. This proposed score about the increase of the database's quality fits well the interactive learning approach, where the main issue in this methodology is when to interact with the end user and how many asks to collect in order to build a good training database.

4. Interactive learning for occupancy estimation

Interactive learning is a process involving an exchange of information with the users in order to collect some important data according to the problem context.

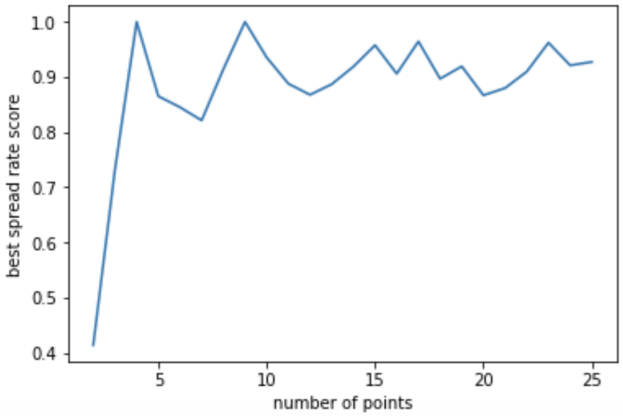

Figure 7: The spread rate score by adjusting the locations of the 8 points using a genetic algorithm

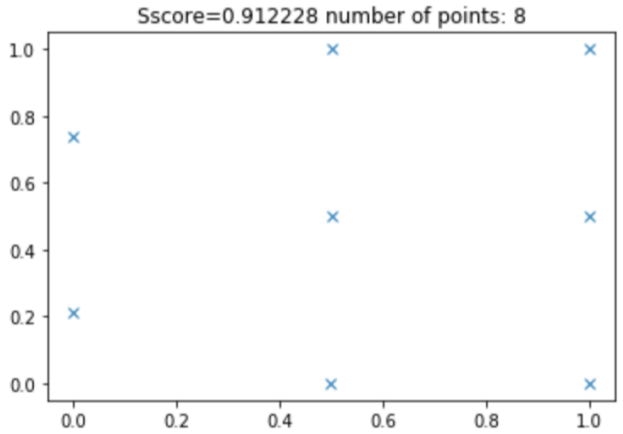

Figure 8: The points location for maximum spread rate values

In supervised learning methods, which are widely used in a lot of applications, the problem of the required target arises in the determination of the number of occupants i.e. the labeling issue is usually tackled using video cameras. Using cameras is generally not acceptable in many places to respect the privacy of occupants. Interactive learning is an adaptation of supervised learning that determines the occupancy by collecting the required labels from the occupants themselves. The problem statement of occupancy estimation has been explained in (Amayri et al., 2019a).

4.1. Multi-layer Perceptron regressor (MLP) with interactive learning

In order to evaluate the interactive approach, Multi-layer Perceptron regressor is deployed. MLP is trained using backpropagation with no activation function in the output layer, which can also be seen as using the identity function as activation function. Therefore, it uses the square error as the loss function, and the output is a set of continuous values. Supervised algorithms need a training period (labeling) which is usually obtained from video cameras in the case of occupancy estimation. In regression algorithms, the better the data are distributed (spread all over the space) the better the prediction that will be achieved. Because of that MLP regressor is chosen to validate the occupancy estimation problem. Indeed, the training is based on the stochastic gradient descent with mini batches which fits exactly the interactive learning with spread rate approach since the data are collected one by one.

4.2. Spread rate score with interactive learning for occupancy estimation

Let $Qscore(\mathcal{P}'_{n+1})$ be the new spread rate when adding one point to a normalized database $Sscore(\mathcal{P}'_n)$ of recorded asks. Considering a new candidate ask in interactive learning depends either on:

- The spread rate improvement i.e. how well points are distributed.
- The number of points n and $n+1$: a decrease in the spread rate can be compensated by a relative increase of the number of points i.e. a higher density of points.

4.3. Case study for one zone office

In order to evaluate the interactive learning experiments with Qscore, a one zone office context is used. This office is equipped with 30 sensors i.e, temperature, relative humidity (RH), motions, CO_2 concentration, power consumption, door and

window positions. Besides, there is a centralized database with a web application for continuously collecting data from different sources. The data cover 10 days from 04-May-2015 to 13- May-2015. During these days a simulation has been done to evaluate the proposed approach. At this step, Human Machine Interface (HMI) interaction with end users in the office is simulated, while the answers of asks are coming from the data labels are obtained from video cameras. Multi-layer Perceptron regressor (MLP) has been applied with the interactive learning process. According to (Amayri et al., 2016a) the set of interesting features to work on are motion counting, acoustic pressure, and occupancy from power. Results presented in this work are based on a period of time $\mathbf{T}_s$ **= 30 minutes**.

4.4. Results

The first step for the validation starts by applying interactive learning with a spread rate concept. The results show a limitation in interactive process because it stopped after 4 asks which causes a high occupancy estimation error equal to 0.42 people, see figure 9. Obviously, the number of collected asks is not sufficient for building a good training database and generate an accurate model. These results lead to conclude that spread rate is not enough because it doesn't take into account the density of points and it will reach a maximum value, smaller than the best possible one, before collecting all required data.

In the following experiment Qscore is used, table (1) illustrates how the 23 asks are distributed along the days by applying Multi-layer Perceptron regressor. Four days are needed to collect the 23 answers, in case the user answers directly to each question. Indeed the number of days may change due to the interest of the occupants.

For this reason, we proceed randomly with a reply probability equal to 50%, the collecting process needs 14 days. In both experiments, an average error equal to 0.04 person has been achieved which is clearly better than the result achieved with spread rate.

The blue curve in figure 10 shows how the spread-rate increases with each record added to the dataset. It reaches 0.14 with the whole dataset.

The estimation process starts with a high average error on the first day which is equal to 0.65 person for 480 samples. After collecting the required training data, the estimation results are improved with an average error equal to 0.1 person, this can be seen on the fourth day (for the same number of samples while the training dataset contains 23 samples). Obtaining all the required answers (training data) decreases the average error of occupancy estimation and it starts to be almost stable with an average error equal to 0.012 person, see figure 11.

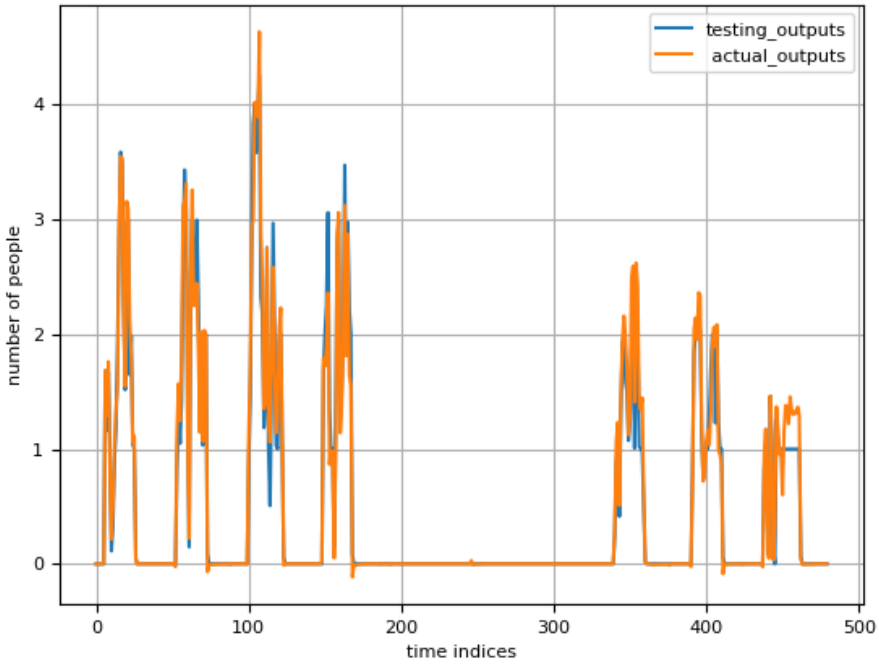

Figure 9: Occupancy estimation with interactive learning by applying MLP and spread rate

Day	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Number of *asks* with 100% replies	18	1	2	1	0	0	0		0	0	0	0	0	0	0	0
Number of *asks* with 50% replies	8	0	0	2	1	1	3	2	1	0	2	0	1	2	0	0

Table 1: Number of asks each day

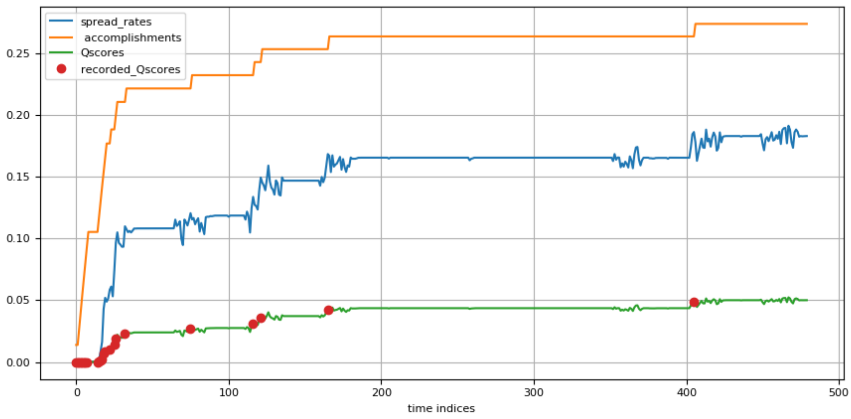

Figure 10: Interactiv learning process

The average error has been calculated for 480 samples, based on a period of time 30 minutes during the 10 days.
Figure 12 presents the results obtained from the learned Multi-layer Perceptron regressor (MLP) considering 4 features as input to the model (motion detection, power consumption, door position and acoustic pressure from a microphone), where both actual and estimated occupancy profiles are plotted with relation to time (quantum time was 30 min). The accuracy achieved from MLP was 93%, and the average error was 0.15 persons.
Additionally, density approach with MLP for occupancy estimation is applied to compare it with spread rate algorithm, the asking process leads to almost the same number of asks as the Qscore, while the average error is increased to 0.27 person, see figure 13.
Applying the interactive learning approach with density increased the complexity of the estimation process. Several parameters should be adjusted with each new context in order to minimize the number of asks and the estimation average error. Contrary, the spread rate is not restricted to any parameter. It depends on the global improvement on the quality of the database in the whole normalized space while density relies directly on the defined neighborhood. For deeper analyses other factors have been investigated for both algorithms (Qscore and density of the neighborhood) with interactive learning: sensitivity to the initialization time, and sensitivity to the features bounds. These factors cause a small effect on the

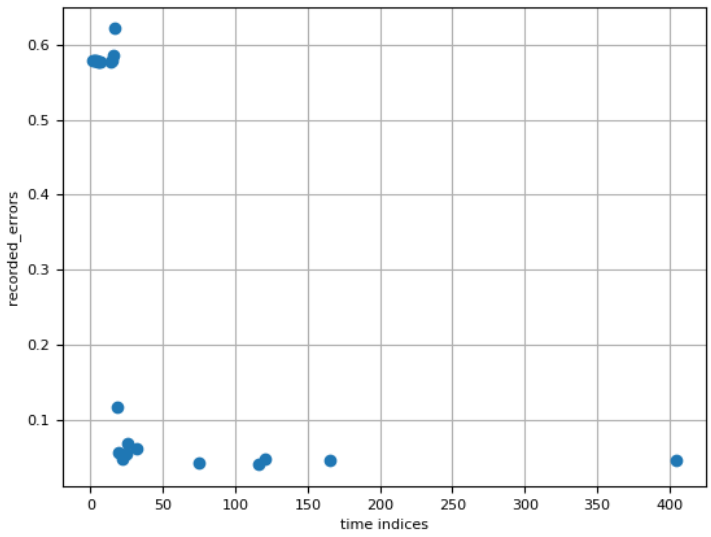

Figure 11: MLP estimation error with each new ask using 480 samples

results of interactive learning with Qscore, which can be ignored. While with density approach the error increased to 0.36 person when changing the initial time from the midnight to working time where the occupant's number is more than one. The MLP model doesn't have the required dataset in order to give good estimation results. In addition, the model is sensitive to the bounds of the features because changing the bounds will affect directly determining the neighborhood, which is the main factor in the ask process. The occupancy estimation results suggest that the Qscore of improved spread rate can be simply generalized to any context.

5. Conclusion

In this paper, a novel approach to assess the quality of training data has been proposed and successfully applied to occupancy estimation via an interactive learning approach, which allows model training via continuous interactions with the users. In particular, two scores (Sscore and Qscore) are calculated depending on a new concept, called spread rate, in order to assess the quality of training data when collected continuously. Extensive simulations have shown that the developed quality assessment approach improves significantly the estimation results while reducing

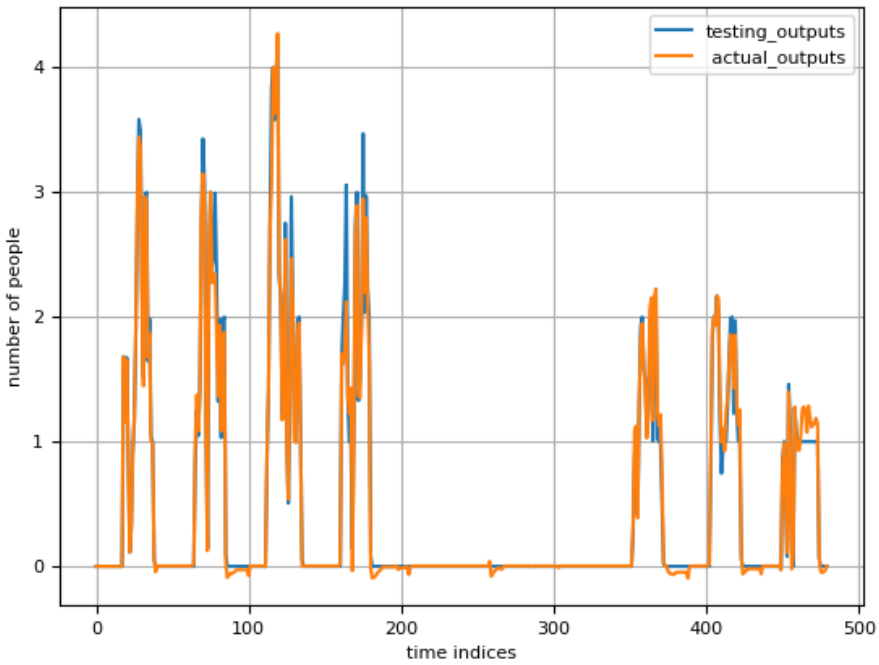

Figure 12: Occupancy estimation with interactive learning by applying MLP and Qscore

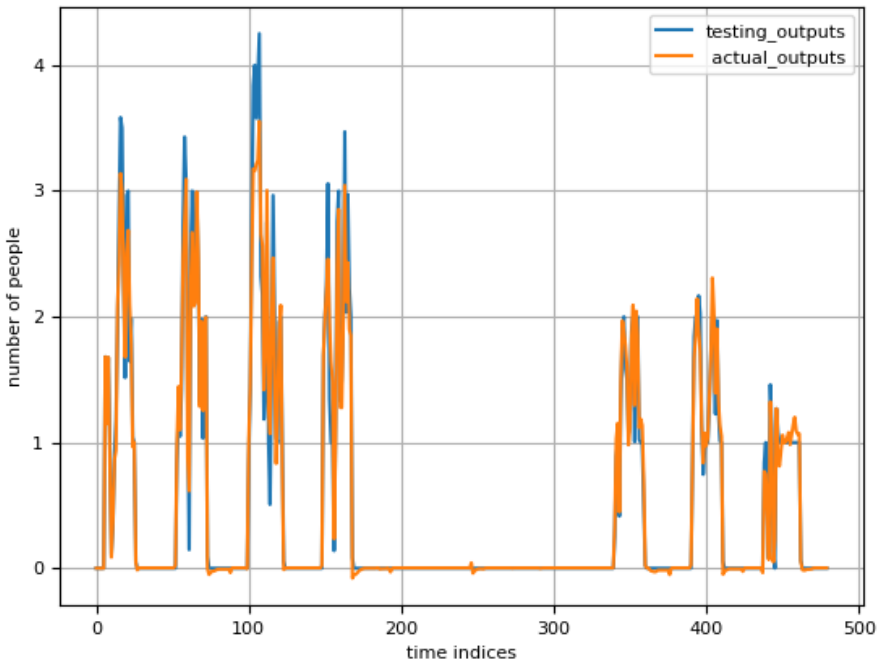

Figure 13: Occupancy estimation from MLP and density of the neighbourhood

the number of interactions with the occupants (around 16 asks). Although, only MLP has been considered as a supervised learning approach, it is obvious that the proposed approach could integrate easily other learning machines (e.g. SVM, decision trees, etc.). Additionally, the comparison between the current approach of interactive learning using spread rate and the previous one using local neighborhood density shows improvement in the occupancy estimation results, besides simplifying the model because of the less number of parameters. Indeed, the spread rate concept can be used in many different machine learning and automatic control application where the quality of the data is of crucial importance. There are several interesting potential future works. For instance, the research presented in this paper could be extended further to consider online learning where the parameters of the learning machine could be updated online each time new training data are introduced. In this context, making the overall framework robust to outliers would be a potential work in order to avoid compromising the overall training model. Other potential future works could be devoted to integrate streaming feature selection in case new sensors are introduced. The overall goal would be to propose a unified occupancy estimation approach that takes into account simultaneously the quality of the data and the relevance of the features.

Acknowledgment

This work is supported by the French National Research Agency in the framework of the " Investissements dávenir program (ANR-15-IDEX-02) ECO-SESA, COMEPOS projects. The authors would like to thank the associate editor and the reviewers for their helpful comments.

REFERENCES

Amayri, M., Arora, A., Ploix, S., Bandhyopadyayc, S., Ngod, Q.-D., Badarla, V. R., October 2016a. Estimating occupancy in heterogeneous sensor environment estimating occupancy in heterogeneous sensor environment. Energy and Buildings 129, 46–58.

Amayri, M., Ploix, S., Bouguila, N., Wurtz, F., 2019a. Estimating occupancy using interactive learning with a sensor environment: Real-time experiments. IEEE Access 7, 53932–53944.

Amayri, M., Ploix, S., Kazimi, H., Ngo, Q., Safadi, A., 2019b. Estimating occupancy from measurements and knowledge using bayesian network for energy management. Sensor 7, 53932–53944.

Amayri, M., Ploix, S., Reignier, P., Bandyopadhyay, S., 2016b. Towards Interactive Learning for Occupancy Estimation. In: ICAI'16 - International Conference on Artificial Intelligence (as part of WORLDCOMP'16 - World Congress in Computer Science, Computer Engineering and Applied Computing). Las Vegas, United States.
URL https://hal.archives-ouvertes.fr/hal-01407401

Duda, R. O., Hart, P. E., Stork, D. G., 2012. Pattern classification. John Wiley & Sons.

Helmbold, D. P., Long, P. M., 1994. Tracking drifting concepts by minimizing disagreements. Machine learning 14 (1), 27–45.

Jin, M., Bekiaris-Liberis, N., Weekly, K., Spanos, C. J., Bayen, A. M., April 2018. Occupancy detection via environmental sensing. IEEE Transactions on Automation Science and Engineering 15 (2), 443–455.

Kuh, A., Petsche, T., Rivest, R. L., 1991. Learning time-varying concepts. In: Advances in Neural Information Processing Systems. pp. 183–189.

Lanquillon, C., 2001. Enhancing text classification to improve information filtering.

Milenkovic, M., Amft, O., 2013. Recognizing energy-related activities using sensors commonly installed in office buildings. Procedia Computer Science 19, 669–677.

Minor, B. D., Doppa, J. R., Cook, D. J., Dec 2017. Learning activity predictors from sensor data: Algorithms, evaluation, and applications. IEEE Transactions on Knowledge and Data Engineering 29 (12), 2744–2757.

Ordonez, F. J., Roggen, D., 2016. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. Sensors 16 (1).
URL http://www.mdpi.com/1424-8220/16/1/115

Perner, P., 2003. Incremental learning of retrieval knowledge in a case-based reasoning system. In: International Conference on Case-Based Reasoning. Springer, pp. 422–436.

Quionero-Candela, J., Sugiyama, M., Schwaighofer, A., Lawrence, N. D., 2009. Dataset Shift in Machine Learning. The MIT Press.

Sehili, M. A., Istrate, D., Dorizzi, B., Boudy, J., Aug 2012. Daily sound recognition using a combination of gmm and svm for home automation. In: 2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO). pp. 1673–1677.

Tong, S., Chang, E., 2001. Support vector machine active learning for image retrieval. In: Proceedings of the Ninth ACM International Conference on Multimedia. MULTIMEDIA '01. ACM, New York, NY, USA, pp. 107–118.
URL http://doi.acm.org/10.1145/500141.500159

Valle, R., Dec 2016. Abroa: Audio-based room-occupancy analysis using gaussian mixtures and hidden markov models. In: 2016 Future Technologies Conference (FTC). pp. 1270–1273.

Wang, J.-D., Liu, H.-C., 2009. Evaluating the ambiguities between two classes via euclidean distance. Asian Journal of Health and Information Sciences 4 (1), 21–35.

Wang, J.-D., Liu, H.-C., 2011. An approach to evaluate the fitness of one class structure via dynamic centroids. Expert Systems with Applications 38 (11), 13764–13772.

Wang, J.-D., Liu, H.-C., Shi, Y.-C., 2009. A novel approach for evaluating class structure ambiguity. In: 2009 International Conference on Machine Learning and Cybernetics. Vol. 3. IEEE, pp. 1550–1555.

Wang, J.-D., Shi, Y.-C., 2009. Evaluating the ambiguity of non-linear separable class structure via instance neighbor entropy. In: The 20th Workshop on Object-Oriented Technology and Applications. p. 54.

Zhang, J., Ghahramani, Z., Yang, Y., 2005. A probabilistic model for online document clustering with application to novelty detection. In: Advances in Neural Information Processing Systems. pp. 1617–1624.