International Journal of Energy Systems and Mechanical Engineering

Volume1, Issue 2 2025

Machine Learning Model Towards Evaluating Data gathering methods in Manufacturing and Mechanical Engineering

Li, Guangzhong Hu, School of Mechanical Engineering,vietnam

Research Article

Date of Submission: 10-04-2025 Date of Acceptance: 19-05-2025 Date of Publication: 1-06-2025

ABSTRACT

To accurately simulate the behavior of intricate mechanical systems and processes, supervised machine learning (ML) models need a large amount of training data. Since real-world experiments or suitable simulations are costly, time-consuming, or incident-related, it is imperative that sample data be acquired efficiently. Data is often gathered in mechanical engineering and production using recognized Design of Experiments (DOE) techniques. At the same time, Active Learning (AL), which promises to reduce data, is becoming more and more popular in the academic community but is seldom used in business. In order to get better predictive outcomes with fewer samples on regression problems, we compare the most popular data sampling techniques with AL in this work. Regardless of their disparate criteria, we provide a unique assessment framework that enables the controlled and objective comparison of diverse sampling techniques. By examining the sample efficiency, stability, and prediction accuracy of the resultant machine learning models, we assess whether AL or DOE approaches should be used for the job of data creation using three illustrative use cases (UCs). For engineers and data scientists who need very effective data collecting for subsequent ML application, this article offers helpful advice.

KEYWORDS: additive manufacturing, machine learning, design of experiments, data generation

INTRODUCTION

The availability of open source frameworks has made training and implementing ML (Machine Learning) models much simpler. As a consequence, sophisticated machine learning algorithms are actively used in a variety of fields. The availability of data is still one of the major obstacles to entrance in the field of machine learning. Finding domain-specific data sets may be quite challenging. Numerous top businesses, like Google and Facebook, focus on gathering and making money from the data [1–5]. In order to get a competitive advantage, businesses in the mechanical engineering and manufacturing sectors work to enhance their product design and production procedures by using accessible data and cutting-edge analytics technologies. Numerous real-world applications, including approximating complicated systems, accelerating and improving simulations, topology-optimization, and energy management, are covered by currently available machine learning methods [6–11]. However, solving a particular issue often requires gathering training data first, which entails conducting several expensive and time-consuming experiments on actual machinery or simulated surroundings. Furthermore, because the data must be provided in a certain quality and variability, a huge amount of data does not always ensure the project's success. The degree of adoption among manufacturing organizations is still low since it is difficult to foresee the results, associated expenses, and necessary capabilities of a real-world

experiment [12–17]. All of this makes it difficult to adopt ML solutions in the setting of intricate production systems and procedures. To guarantee the success of the whole project, it is crucial to choose an ideal data creation approach that is particularly appropriate for the planned ML solution. Making an informed choice on how to produce the necessary data for a particular UC (Use Case) might be difficult. It is uncertain if and to what extent AL (Active Learning) may improve performance over a conventional well-planned DOE since the majority of AL (Active Learning) research use random sampling as the performance baseline [18–24]. To the best of our knowledge, no research has been done that explicitly compares DOE with AL approaches in the context of mechanical engineering and production. In this work, we directly compare the random data generation, Query-by-Committee (QBC) sampling from the AL region, and Latin Hypercube Sampling (LHS), a popular DOE technique, to fill these research gaps. In order to do this, we create a unique assessment framework that enables a controlled, bias-free direct comparison of DOE and AL approaches [25–33]. We make sure the outcomes are relevant for
mechanical engineering and production jobs by using suitable UCs with varying degrees of complexity. Sample efficiency, result stability, and attained prediction accuracy of machine learning models trained on the produced data are used to assess the applicability of all data production techniques [34–41]. LGBM (Light Gradient Boosting Machine), which is based on regression trees, and bagging, a technique to improve prediction performance by merging several ML models, are used in the assessment. Both methods are well-known and often used in several fields of machine learning. When choosing data generating methodologies for machine learning projects in mechanical engineering and manufacturing, engineers and data scientists may use the findings as a useful guidance [42–49].

2.0 BACKGROUND

The most recent approaches to data creation and AL techniques as a branch of machine learning are reviewed and summarized in this section.

Data Generation Approaches

1)n [57–63].
It takes a lot of samples with enough variation to use supervised machine learning models for complex system prediction. A actual procedure or a realistic simulation must be carried out and observed in order to generate a labeled sample [50–56]. DOE research suggests a number of techniques for selecting samples to be labeled, which are popular because of their ease of use or adaptability and are explained as follows:

Completely randomized designs: These designs randomly allocate a value to every feature. It was introduced in 1964 and is a simple method. While random sampling performs well for high sample counts, small sample sizes are not always well-spread. As one performance benchmark for our evaluation, Figure 1 shows sample

distributions for random sampling, which is simple to execute.
2) Block designs that are random: Completely randomized designs are expanded upon by randomized block designs. The fundamental idea is to divide each feature's value space into homogenous blocks and then randomly sample each block. A random-ordered uniform distribution with extra noise results from the uniform distribution of the selected samples among the designated blocks [64–71]. Latin Hypercube Sampling (LHS) is a popular method that combines the homogenous space-filling of classical designs with the scalability of random sampling. Examples of LHS distributions are shown in Figure 1. LHS produced n cubes of equal probability if we divided each dimension of a k-dimensional experimental area D into n uniform blocks [72–76].

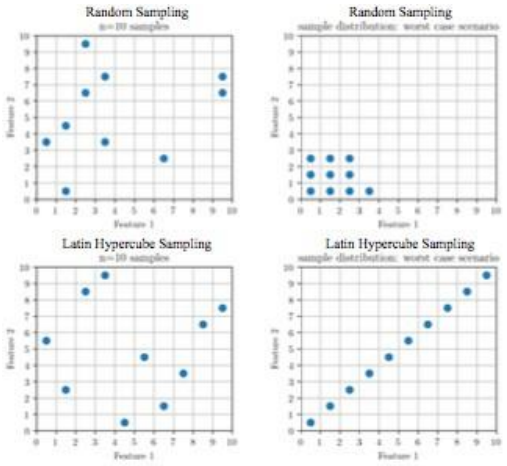


Fig. 1: Two-dimensional random and LHS, n=10.

Full factorial designs: Full factorial designs are com- mon sampling designs where all features are

each set to a predetermined number of l levels. Two-level designs (l = 2), where each feature may have a high or low value, are popular options, at least in engineering. For each of the k characteristics, a complete factorial design generates every conceivable combination of the available values. This results in l experiment runs for n features with l levels each. Keep in mind that complete factorial designs produce 2187 samples even for seven attributes with only three levels (low, mid, and high). Experiments with more characteristics are not appropriate due to the exponential increase, particularly when the cost per trial is considerable [77–81].

2) Fractional factorial designs: "A factorial experiment in which only an adequately chosen fraction of the treatment combinations required for the complete factorial experiment is selected to be run" is the definition of a fractional factorial design. By using just a portion of them, the trial runs produced by a complete factorial design are decreased [1–7]. The choice of which components of the experiment to retain will determine the method's success. Simple methods choose fractions like, etc. and usually provide satisfactory findings 24 when the selected trial runs are balanced and orthogonal. A fractional design lowers the number of runs by the selected denominator d to 1 ∗ 1*n for the complete factorial example above, which has n features with l levels each [8–13].

3) Response surface designs: According to projects, traditional designs such as Central Composite Design (CCD) and Box-Behnken Design (BBD) [18] are mostly used for physical investigations because they choose a limited number of evenly distributed samples from the feature space. A sample selection of BBD and CCD for three characteristics is shown in Figure 2. As shown in, the number of samples in these traditional designs is set and directly correlated with the experiment's feature count (n). BBD generates 13 samples organized in a sphere given three (n = 3) dimensions. Every sample is located on the boundaries of the experimental area, with the exception of the center point. For n = 3, CCD generates 15 samples that are arranged as a collection of "star points" around a single center point (see Figure 2) [14–19].

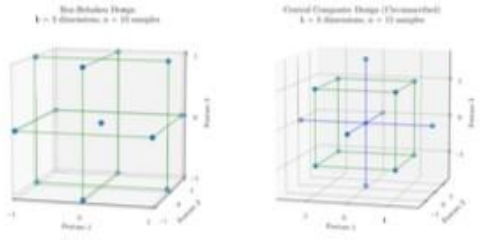


Fig. 2: Left: BBD with one center point. Right: CCD with star points aligned around the center.

Active Learning

AL algorithms are used in cremental query learning, which is a subfield of machine learning that consists of two parts: a training algorithm, usually an ML model, and a query algorithm. The primary idea is that if the training algorithm can choose the samples using the query algorithm, it will be able to learn more with fewer examples. A typical AL loop is shown in Figure 3, where the primary objective is to train an ML model as efficiently as possible with the fewest feasible data. From the unlabeled train set U, the AL algorithm chooses one or more samples (step size), and an oracle provides the corresponding label for further training the ML model or models. The collection of all feasible feature value combinations for n features is the feature space X: X = {($\vec{x}$) | $\vec{x}$ is an acceptable feature combination}. Keep in mind that X may have an endless number of entries since it is continuous. In the context of AL, we also refer to the finite subset P ÷ X as a pool. Additionally, the subset of P with known labels is called the labeled train set L: L={p •P| labelofpisknown}.U = P \ L is the definition of the unlabeled train set U. The oracle shown in Figure 3 may be a simulation (e.g., of a production process), a real-world process, or a human annotator [20]. An AL approach actively expands its train set, which is the primary distinction from the statistical DOE discussed in Section II-A [20–28]. It is anticipated that each subsequent sample added to the train set would optimize the learning impact. This method chooses the instance from U where the prediction made by the underlying machine learning model is least confident. There are several different query algorithms, the two most

Uncertainty sampling and Query-By-Committee (QBC) are popular and widely utilized techniques [21]. Since uncertainty sampling is only useful when a probabilistic machine learning model is used, we do not include it into our work. The committee C = {θ1,... , θC } in the QBC technique was trained on the labeled data set L. The maximum disagreement principle is used to choose the subsequent sample. For each available x ∈ U, every committee member θi ∈ C predicts a label; the sample with the greatest discrepancy in expected labels is chosen. As shown in Figure 4 [29–34], this method necessitates that the committee members θi have distinct hypotheses. QBC concentrates on contentious areas in the unlabeled train set U, questions that sample x ∈ U with the most disagreement, and enhances all committee C members by including x while re-fitting these. When compared to other methods, QBC is said to regularly provide positive outcomes [23]. Furthermore, QBC has several Python implementations and is now one of the best-documented and implemented AL algorithms [35–41].



Fig. 3: The AL algorithm repeatedly selects samples, an oracle labels these, and its results improve the ML model(s).

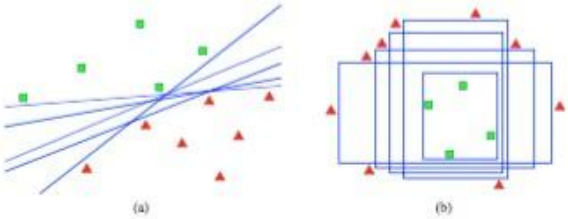


Fig. 4: Different hypotheses of (a) linear and (b) axis-parallel binary classifiers, which are all consistent with the labeled training data $\mathcal{L}$ [20].

3.0 RELATED WORK

DOE Applications in Machine Learning

Numerous computationally costly methods can be used more easily thanks to DOE. The use of DOE for algorithm-tuning is a major focus of the study. Using an example of a block relocation job in container terminal logistics, Caserta and Voss refine a suggested Corridor Method algorithm using Response Surface Methodology (RSM). Finding parameter values for algorithms whose performance relies on employing non-obvious combinations of values may be accomplished effectively by combining Factorial Experimental Design with a local search technique. The DOE appropriateness for optimizing algorithms on an ant colony system for the traveling salesman issue example was shown using Response Surface Experimental Design. Numerous studies demonstrate that using DOE in the parameter-tuning process of machine learning algorithms may provide outstanding results at affordable computing costs [1–17].

From feature selection for training predictive models to behavior studies of neural networks in the context of reinforcement learning, DOE turns out to be an appropriate tool for addressing a variety of ML application difficulties. Consequently, DOE is a customized tool for producing training data while building machine learning models [18–27].

Data Generation in Machine Learning

Data generation for ML model training may be done in a number of ways. Simple random sampling is often used to produce data. For example, Moiz et al. create data for a behavior approximation of a heavy-duty engine simulation using an ML model using random sampling techniques. Nonetheless, it is generally believed that DOE and active learning techniques can provide improved performance with less training data needed for practical machine learning applications [28–36].

In order to train deep neural networks for the optimization of the composite textile draping process, Pfrommer et al. anticipate that the amount of training data needed may be reduced with the use of DOE techniques. Probst et al. train predictive models on a complicated computational fluid dynamics (CFD) simulation by mapping its input-output parameters using optimum Latin hypercube DOE. To sum up, there are now a number of approaches to generate data for later ML model training, but there isn't much practical advice on which approach to use in what circumstances [37–43].

4.0 DEVELOPING AN EVALUATION FRAMEWORK

Our studies' main objective is to compare AL with two traditional DOE techniques: random sampling and LHS. Only when an AL approach regularly beats well-designed DOE methods without previous tweaking and trial runs can it be deemed production-ready [44–49].

TABLE I: Data requirements of the compared data generation strategies and solutions proposed in our evaluation framework.

Strategy	Required input data	Framework solution
Random sampling	Continuous	GTM
LHS	Continuous	GTM
QBC	Discrete and finite	Create pool $\mathcal{P}$ via GTM

Input Data Requirements

The data requirements for random sampling, LHS, and QBC are listed in Table I. Since real-world data sets don't fit these criteria for LHS or random sampling, we provide an evaluation framework with fixes. This structure, which is explained in the following [50–54], also offers a suitable data base for QBC.

Since both LHS and random sampling choose samples straight from a continuous feature space, they cannot be used directly to preexisting data sets. It is necessary to have an oracle that can label data live, encompassing not just the current data points in a particular data set but also continually all potential samples within that data set's range. For this stage, actual experiments or an appropriate simulation may

usually be employed. However, it is uncommon to find simulations with sufficient complexity and performance right out of the box, and utilizing a real machine severely restricts the speed of validation and greatly increases the work and expense [55–61].

In order to get around this problem, we use the input real-world data set to fit a ground truth model (GTM), which then acts as an oracle. Within the training observation boundaries, a well-trained GTM may generalize throughout continuous feature space and approximate relationships in the original data set. The use of random sampling and LHS is made possible by this GTM, which not only replicates labels for samples from the original data set but also labels any sample in the whole continuous feature space of the original data. Any sampling technique that calls for a continuous feature space is satisfied by this method [62–69].

Because the third approach, QBC, needs a limited pool of unlabeled samples to iterate over in order to choose the next training samples, it cannot be utilized directly with GTM (cf. Section II-B). The input data set used for GTM training may be utilized by any technique that samples from a discrete finite pool. Due to the following disadvantages, we do not adopt this strategy. First, the number of possible train samples is greatly decreased by dividing the data set into train and test sets [70–76].

However, using the whole pool U for testing biases the outcome since U eventually becomes smaller and only includes samples with the least amount of information gain. Secondly, no claims can be made for any set of input data. It may (i) have very unevenly distributed samples (bias), (ii) be excessively large, making it difficult to choose a relevant subsampling, or (iii) be complicated, making it difficult to select a bootstrap train set. Third, there is often little to no previous data available when doing tests on an actual physical system. An essential component of using AL in practical applications is understanding how to initiate learning for AL techniques in the most effective manner. As a result, a strong validation schema for active learning should provide a high degree of flexibility and control over the creation of test and training data sets. The aforementioned issues are avoided by our validation method, which also permits flexible data collecting [77–81].

Incremental Evaluation

The comparison of data generating techniques is shown in Figure 5. We break it up into four stages, which are described below:

1) GTM fitting: This stage entails using a selected input data set to train an ML model. Data preparation, hyperopia remoter tuning of the ML model, prediction accuracy validation, and storing the trained GTM for later use as an oracle are all included in this stage, which calls for a certain level of data science expertise [1-4].

2) Data generation: Selecting the sample with the greatest information gain ensures a better learning curve, therefore each iteration step from m = m + 1 is critical. By

choosing samples to be labeled, Figure 6 shows how the various sampling techniques expand their train set. The input consists of a collection of size m samples that represent the samples that are currently known, together with their labels. Within the predetermined feature space X, the random technique produces a single random sample that is statistically independent of any previous choices. As a consequence, a set of m + 1 samples is produced, which includes the newly chosen sample together with all of the previous m samples [5–9].

From the feature space X, LHS creates a set of a specified that cannot be expanded or modified in the future. As a result, in our trials, LHS separately generates a new sample of size m + 1 and discards the input of m samples. This implies that it is not feasible to extend an existing data set created using LHS iteratively for real-world applications, and the number of trials to be carried out must be determined in advance. Despite this, there are application scenarios where the number of tests is predetermined due to financial or time restrictions, which makes it pertinent to our research. Since only the ultimate performance matters in this situation, random sampling, LHS, and AL are seen as equally valid possibilities in the process. If an LHS performs well for the predetermined number of trials, then choosing it over alternative extensible data production procedures is justifiable [10–16].

Using the input of m samples, the third technique (AL) chooses the next sample from unlabeled set U that its committee members disagree on the most. Every committee member $\theta \in C$ is allowed to make a prediction for each of the remaining unlabeled data $u \in U$. The standard deviation of these predictions is computed by the function $s(\theta 1(u), \theta 2(u),...$ The sample u with the highest standard deviation is chosen because it shows the greatest degree of disagreement among committee members. The outcome is a collection of m+1 samples that includes one extra sample based on greatest disagreement in addition to the previous m samples [17–26].

3) Develop and verify machine learning models: To cover the whole input feature space and accurately assess the performance of predictive models built during the assessment, LHS creates a single test set of suitable size. The prediction models are trained using train data sets that are labeled by the GTM and sampled using random sampling, LHS, and AL. For all comparative sampling techniques, the impact of each increase in train set size is verified on the test set. This makes it possible to compare the sample efficiency, stability, and prediction accuracy of trained models directly between random sampling, LHS, and AL [27–33].

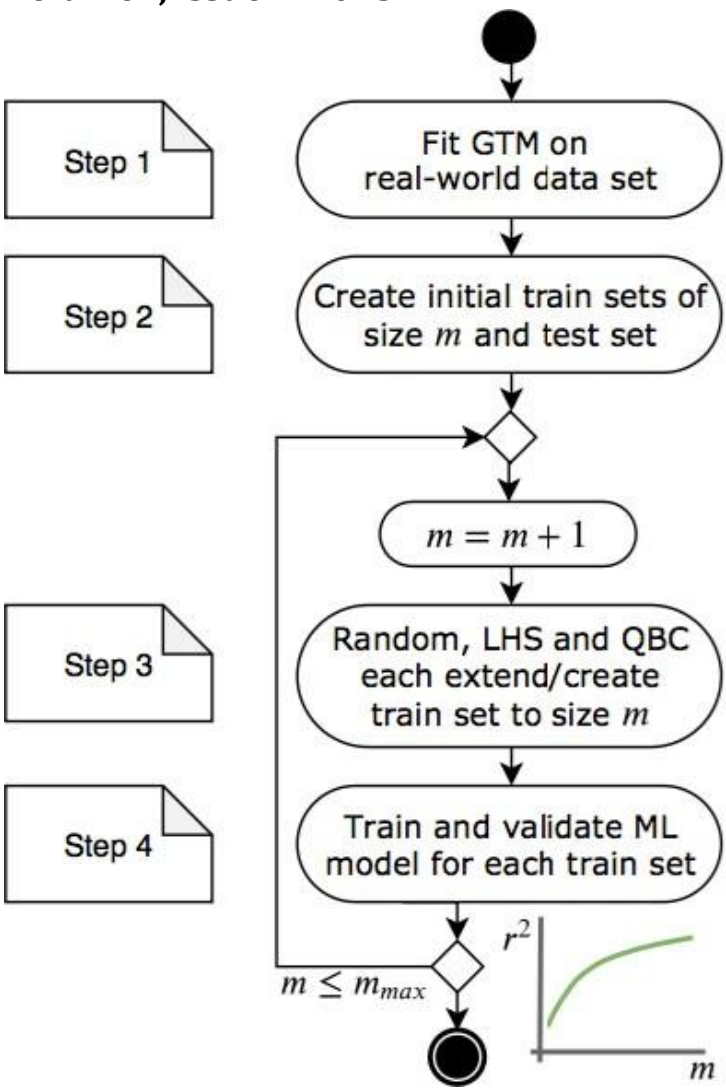


Fig. 5: We fit a GTM on a data set and use it to label samples requested by QBC, random and LHS. These grow with an increasing m and yield a validation plot. The entire procedure can be repeated multiple times (cross-validation) [67-73].

EXPERIMENTS

The assessment framework we created in the previous part is expanded upon in this section, which also carries out the required actions and analyzes the outcomes.

A. Evaluation data sets

Validation UCs are chosen based on two primary criteria: on the one hand, the assessment framework needs a labeled data set for GTM fitting. In contrast, this work focuses on different sampling strategies for regression problems while modeling intricate mechanical engineering systems and processes. The chosen UCs

are based on data sets regarding complicated mechanical systems that are accessible to the public.

1) The Pumadyn series of data sets is created by modeling the dynamics of a Puma 560 robot arm. A robotic manipulator is confidently described by using simulation. Pumadyn data sets range in complexity from linear to non-linear interactions with different noise levels.

in the signals that were measured. The data set makes it possible to forecast the angular acceleration of one of the robot's arm links and learn the dynamic model of the robotic arm. This study uses the Pumadn-8nm and Pumadn-32nm data sets from the specified family. Pumadyn-8nm has one goal variable (angle acceleration of joint 3) and eight input characteristics (angle locations of three links, angle velocities of three links, torque at the first two joints). There are 8192 distinct observations in all. With the same number of observations and angle acceleration of joint 6 as a target variable, Pumadyn-32nm provides 32 input features (angle positions of links 1 to 6, angel velocities of links 1 to 6, torques at joints 1 to 5, change in mass of links 1 to 5, change in length of links 1 to 5, change on viscous friction of links 1 to 5). There is a moderate amount of noise in both extremely non-linear data sets.

2) The Electric Motor Temperature data set provides 36475 distinct sensor readings from a permanent magnet synchronous motor produced on a test bench. The gathered information is used to forecast the permanent magnet's surface temperature using the ambient temperature, coolant temperature, motor voltage, current, speed, and torque.

Machine Learning Models

In this study, we use two well-known ML gradient boosting algorithms: LightGBM (LGBM) and Extreme Gradient Boosting (XGBoost). Gradient boosting is a potent machine learning approach that achieves state-of-the-art outcomes in a range of real-world tasks while being sample-efficient and quite simple to use. The gradient boosting class of machine tree-based learning algorithms includes both XGBoost and LGBM. To address the shortcomings of each individual learning model, the gradient boosting approach uses the data set to train several combined learning models. These are referred to as "weak" learning models since they often fall short of achieving more accuracy on their own. Every new learning model concentrates on target values that were previously mispredicted. Several "weak" learning models' predictions serve as the foundation for the final forecast. Accuracy may rise dramatically as a result. We also examine how bagging several predictive models enhances overall performance for various sample procedures. By repeatedly executing an existing regression model on a batch of resampled data and averaging the results, bagging may enhance performance. Bagging is a useful technique for improving decision-tree-based algorithms' performance from one angle and can provide some benefits when combined with QBC sampling techniques, making it a suitable complement to our experimental setup.

Training and Validation of GTMs

Because of the great prediction accuracy attained across all three data sets, XGBoost models [46] are employed for GTM. The TPOT library [47], which uses evolutionary algorithms to obtain near-optimal model parameters within a user-defined range, is used to do model tweaking for each data set. Ten-fold cross-validation (CV) is used to validate each model. The data collection is divided into 10 equal portions and scrambled during validation. The remaining nine parts are utilized for training, and each component is used once to validate prediction accuracy against actual labels. Ten iterations, known as CV folds, are carried out in total. For each CV fold, the explained variance score is used to assess prediction accuracy. The ultimate performance parameter of a trained GTM is the average explained variance score over all 10 CV folds. A 95.6% explained variance score is attained by a tuned GTM trained on the Pumadyn-8nm data set. Figure 7a shows its forecast accuracy. The y-axis shows anticipated values for the identical input parameters as determined by CV, whereas the x-axis shows actual values. Since it denotes a perfect match between the expected and actual values, the ideal prediction is located on a line that is slanted 45 degrees toward both axes. Figure 7b shows the prediction accuracy of the tuned GTM trained on the Pumadyn-32nm data set. It receives a 91.7% explained variance score. Figure 7c shows the tuned GTM trained on the Electric Motor Temperature data set, which has an explained variance score of 96.4%. High explained variance scores and visualized prediction performances demonstrate how effectively tailored GTMs approximate relationships in original data sets. Full-sized data sets with thousands of observations are used to train all three GTM models. The next stage is to look at data sampling techniques that make it possible to use machine learning to discover dependencies in the data in the most sample-efficient manner.

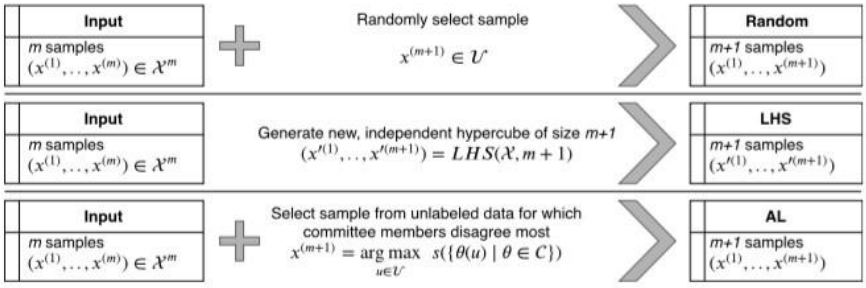


Fig. 6: Given m samples, the sampling strategies create $m+1$ samples differently. Random extends the given set by one random sample. LHS creates, independent from the given samples, a fresh set of $m+1$. QBC lets all committee members predict targets for all still unlabeled samples $\mathcal{U}$ and selects that with highest disagreement to extend the given m samples.

Comparison of Data Generation Strategies

In order to compare data creation techniques, we establish the following crucial criteria:

• Prediction accuracy of machine learning models trained on the provided data; all assessment scenarios include regression tasks. Since explained variance (R2) score intuitively illustrates how effectively a particular machine learning model captures underlying dependencies in the data (100% = absolute correctness, 0% = no

relationships learnt), we have chosen it as a performance indicator.
•Sample efficiency: when the size of the train set increases, the rate of prediction accuracy increases.
The reproducibility of the observed prediction accuracy for the specified number of samples throughout several iterations is known as stability of results. The standard deviation of the observed prediction accuracy for the specified train size, which is obtained across several iterations, is used to reverse the findings' stability. On the three selected UCs, random sampling, LHS, and QBC approaches are contrasted. Two data creation procedures are used to build an initial training data set L: LHS generates an experimental design of the same size, and random sampling selects m random points inside the valid feature space. A GTM labels those spots. Both initial data sets are used to train identical LGBM models. A different testing data set T of size t is used to assess the performance of trained models. It spans the same predetermined feature space range and integrates LHS design. To guarantee accurate and consistent assessment of the prediction performance for every experiment, the same testing data set is utilized for every validation run of the UC under consideration. In addition to an initial train set, the QBC sampling strategy needs an unlabeled pool of data U of size l from which fresh observations may be chosen. LHS design is used to produce pool set U and first QBC train set L. Bootstrapped portions of the training data are used to train a committee made up of three LGBM models. The validation cycle is then repeated once the original data sets are expanded with an extra k data points. The validation configuration for each validation UC is detailed in Table II.

TABLE II: Configuration of Training Setups

Validation UC	Initial train set size m	Pool set size l	Step size k	Test set size
Pumadyn-32nm	100	400	3	2000
Pumadyn-8nm	50	400	3	2000
ETM UC	50	400	3	2000

The main difference between experiments on different sampling techniques is the way new points are chosen. For random sampling corresponding training data set is repeatedly augmented with three randomly generated points within the defined feature space. It is important to note, that LHS excludes the possibility to subsequently refine an existing data set. For every required data size, a completely new train set is generated. It provides a solid baseline towards what prediction performance can be achieved with more sophisticated sampling methods, provided the training size is chosen and data is generated in one iteration only. In the case of QBC, a trained committee decides at each step which next observation points from the pool data set to label with GTM and include to the training data set. Apart from the training a LGBM model, we use a bagging predictive model consisting of three LGBM
models. A bagging predictive model can work particularly well while combined with a QBC sampling strategy based on a committee with equal compositions [45]. The experiment results are averaged over 30 independent runs for every setup (GTMs for three predictive tasks, three sampling techniques and two predictive

models) including iterations from initial train set size to the final training size.



(a) Predicted vs. Real angle acceleration of joint 3, Pumadyn-8nm data set.

(b) Predicted vs. Real angle acceleration of joint 6, Pumadyn-32nm data set.

(c) Predicted vs. Real surface temperature of permanent magnet, Electric Motor Temperature data set.

Fig. 7: CV Results for GTMs.

RESULTS

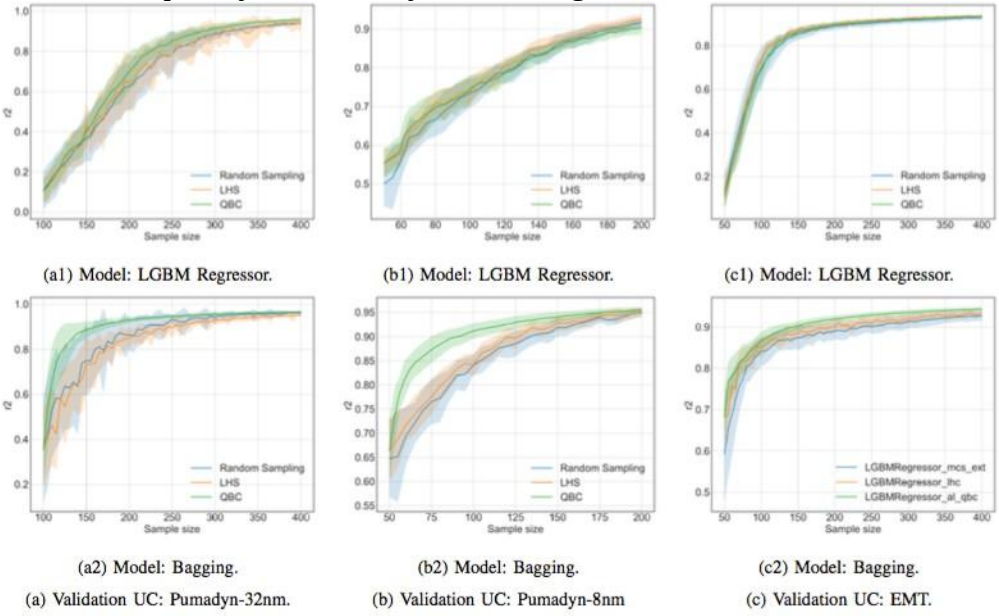
Figure 8 displays the assessment results for two ML models and all three validation UCs. These graphics make it possible to evaluate different sampling techniques using three predetermined comparison criteria:

• Prediction accuracy is represented by points on learning curves that are shown as solid lines.

• The steepness of a particular learning curve is known as sample efficiency.
The standard deviation of the observed accuracy values for the specified number of training samples, shown as the filled area around the specified learning curve, may be used to assess the stability of the findings. Random sampling has the largest deviation range across experimental runs for both the single LGBM model and the bagging model for the Pumadyn-32nm validation UC (Figures 8a1 and 8a2). If a single LGBM model is trained, the QBC sampling strategy performs on par with or somewhat better than both random sampling and LHS. Additionally, the bagging model in conjunction with QBC clearly outperforms all other methods of data sampling. It has a better final explained variance score, smaller prediction performance variations across validation runs, and a sharper gain in prediction accuracy with increasing train set size. The effectiveness of three data sampling techniques is then examined on a Pumadyn-8nm UC.

Figures 8b1 and 8b2 are almost identical to the earlier UC learning patterns. Both QBC and LHS outperform random sampling when using a single LGBM model in terms of prediction performance and result stability at modest training sample numbers. However, as compared to other sampling techniques, larger QBC-sampled training data results in a little decline in prediction performance. On the other hand, the bagging model in conjunction with the QBC sampling technique significantly beats all other approaches. Similar results are obtained from the experiment using the Electric Motor Temperature data set. Figures 8c1 and 8c2 show related learning curves. While QBC sampling may provide more consistent prediction performance

than random sampling alone when using a single LGBM model, bagging models learn more quickly and steadily when using QBC.



(a1) Model: LGBM Regressor. (b1) Model: LGBM Regressor. (c1) Model: LGBM Regressor.

(a2) Model: Bagging. (b2) Model: Bagging. (c2) Model: Bagging.

(a) Validation UC: Pumadyn-32nm. (b) Validation UC: Pumadyn-8nm (c) Validation UC: EMT.

Fig. 8: Performance of random, LHS and AL Samplings over Three Validation UC

CONCLUSION

While planning data collection experiments in the fields of mechanical engineering and manufacturing, we took into consideration the issue of AL preparedness for practical applications on an example of QBC sampling. We examined how QBC techniques may support data-driven approximations of complicated mechanical systems in relation to both an established DOE approach, LHS, and the traditional AL baseline, random sampling. Given that DOE is an industry-wide standard for organizing and carrying out data-collection experiments, this is especially pertinent for industrial applications. This document provides useful guidance for selecting data sampling techniques for certain investigations. We noted that due of possible bias induced during the initial data collection, utilizing existing data sets for the validation of new AL algorithms might provide findings that are deceptive. Additionally, it prevents direct comparisons with other sampling methods like DOE, which primarily put data points in the observation space on their own. By introducing GTMs, which are trained on validation data sets, we suggested a low-effort, sampling-agnostic validation method devoid of the aforementioned biases. To examine the effectiveness and stability of the random, LHS, and QBC sampling techniques, three validation UCs were created. We demonstrated that the ultimate outcomes for all three approaches are equivalent when a large number of trials are conducted. If the expenses of the experiment are very minimal, then simple random sampling is justifiable. LHS or AL turn out to be a superior option when there are fewer experiments available. For every prediction model and sample size taken into consideration, LHS produces strong outcomes in terms of prediction accuracy and

stability. The number of trials to conduct must be decided very early in the planning process since LHS necessitates establishing the sample count a priori. If specific requirements are satisfied, QBC has been shown to be the best option and provides an incremental expansion of the train set. QBC is not appropriate for applications with large data collecting costs since it requires tweaking and trial rounds to provide satisfactory results for single prediction models. QBC is the most effective and reliable data sample methodology without the need for tweaking if the machine learning models meant to be applied to the gathered data are compatible with ensembling methods like bagging, such as LGBM.

REFRENCES

[1] Vartak, Manasi, et al. "ModelDB: a system for machine learning model management." Workshop on Human-In-the-Loop Data Analytics Proceedings. 2016.

[2] Sebastian Schelter et al. "On challenges in machine learning model management." (2015).

[3] Measurement 119 (2018): 117-128 Abdulaziz, Ahmed, et al. "A machine learning model for improving healthcare services on cloud computing environment."

[4] Wang, Chengcheng, et al. "Machine learning in additive manufacturing: State-of-the-art and perspectives." Additive Manufacturing 36 (2020): 101538.

[5] Zhu, Zuowei, et al. "Machine learning in tolerancing for additive manufacturing." CIRP Annals 67.1 (2018): 157-160.

[6] Shafaie, Majid, Hongyan Zhang, Mohsen Sarparast, and Maziar Khademi. The International Journal of Advanced Manufacturing Technology 123, no. 11 (2022): 4385-4398. "Modified GTN parameters calibration in additive manufacturing of Ti-6Al-4 V alloy: a hybrid ANN-PSO approach."

[7] Meng, Lingbin, et al. "Machine learning in additive manufacturing: a review." Jom 72.6 (2020): 2363-2377.

[8] Sarparast, Vahid Tahmasbi, Mohsen, Majid Ghoreishi, and Tohid Jahangirpoor. Journal of the Brazilian Society of Mechanical Sciences and Engineering 42, no. 6 (2020): 1–9. "Experimental and finite element investigation of high-speed bone drilling: evaluation of force and temperature."

[9] Li, Zhixiong, et al. "Prediction of surface roughness in extrusion-based additive manufacturing with machine learning." Robotics and Computer-Integrated printing 57 (2019): 488-495.

[10] Jin, Zeqing, et al. "Machine learning for advanced additive manufacturing." Matter 3.5 (2020): 1541-1556.

[11]Jiang, Jingchao, et al. "Machine learning integrated design for additive manufacturing." Journal of Intelligent Manufacturing (2020): 1–14.

[12] Sarparast, Vahid Tahmasbi, Mohsen, Majid Ghoreishi, and Tohid Jahangirpoor. Biotechnology & Biotechnological Equipment 33, no. 1 (2019): 1616-1625. "Modelling and optimisation of temperature and force behaviour in high-speed bone drilling."

[13] International Journal of Rapid Manufacturing 7.4 (2018): 310-336 Baumann, Felix W., et al. "Trends of machine learning in additive manufacturing."

[14] Gu, Grace X., et al. "Bioinspired hierarchical composite design using machine

learning: simulation, additive manufacturing, and experiment." Materials Horizons 5.5 (2018): 939-945.S
[15] Bahrami, Javad, Sylvain Guilley, Mohammad Ebrahimabadi, Sofiane Takarabt, Jean-luc Danger, and Naghmeh Karimi. "On the Practicality of Relying on Simulations in Different Abstraction Levels for Pre-Silicon Side-Channel Analysis."
[16] Qi, Xinbo, et al. "Applying neural-network-based machine learning to additive manufacturing: current applications, challenges, and future perspectives." Engineering 5.4 (2019): 721-729.
[17] Johnson, N. Additive Manufacturing 36 (2020): 101641 S., et al. "Invited review: Machine learning for materials developments in metals additive manufacturing."
[18] Sadi, Mehdi, Yi He, Yanjing Li, Javad Bahrami, Satwik Kundu, Swaroop Ghosh, Mahabubul Alam, and Naghmeh Karimi. 2022 IEEE 40th VLSI Test Symposium (VTS), pp. 1–12, "Special Session: On the Reliability of Conventional and Quantum Neural Network Hardware." IEEE, 2022.
[19] Qin, Jian, et al. "Research and application of machine learning for additive manufacturing." Additive Manufacturing (2022): 102691.
[20]Caggiano, Alessandra, et al. "Machine learning-based image processing for on-line defect recognition in additive manufacturing." CIRP annals 68.1 (2019): 451-454.
[21] Bahrami, Javad, Sylvain Guilley, Jean-Luc Danger, Mohammad Ebrahimabadi, and Naghmeh Karimi. 2022 Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 1263-1268, "Leakage power analysis in different S-box masking protection schemes." IEEE, 2022.
[22] Yao, Guijun Bi, Xiling, and Seung Ki Moon. Rapid Prototyping Journal (2017) "A hybrid machine learning approach for additive manufacturing design feature recommendation."
[23] Bahrami, Javad, Khaled N. Khasawneh, Jens-Peter Kaps, Abubakr Abdulgadir, Viet B. Dang, and Kris Gaj. Proceedings of the 4th ACM Workshop on Attacks and Solutions in Hardware Security, pp. 45-56. "Lightweight implementation of the lowmc block cipher protected against side-channel attacks." 2020.
[24] Sing, S. L., et al. "Perspectives of using machine learning in laser powder bed fusion for metal additive manufacturing." Virtual and Physical Prototyping 16.3 (2021): 372-386.
[25] Mohammed Bagher Menhaj, Javad Bahrami, Farzad, Ahmadinejad, and Saeed Shiry Ghidary. 2018 4th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS), pages. 217-223, "Autonomous Flight of Quadcopters in the Presence of Ground Effect." IEEE, 2018.
[26]Ko, Hyunwoong, et al. "Machine learning and knowledge graph based design rule construction for additive manufacturing." Additive Manufacturing 37 (2021): 101620.
[27] Negar Jahanbakhsh Javidd, Hasan Hosseini Nasabc, Amir Mohammad Golmohammadib, Hengameh, and Hadiana. "Time Parameter Estimation Using Statistical Distribution of Weibull to Improve Reliability." (2017).
[28] Jiang, Jingchao, et al. "Achieving better connections between deposited lines in additive manufacturing via machine learning." Math. Biosci. 3382-3394 in Eng 17.4

(2020).
[29] Hossein Foroozan, Meysam Gheisarnejad, Mohammad-Hassan Khooban, Zavareh, and Bozorgasl. Naval Engineers Journal 133, no. 3 (2021): 115-135. "New trends on digital twin-based blockchain technology in zero-emission ship applications."
[30] Vincent C. Paquit, Li, Rui, and Mingzhou Jin. Materials & Design 206 (2021): 109726. "Geometrical defect detection for additive manufacturing with machine learning models."
[31] Kristian Martinsen, Oleksandr Semeniuta, Ivanna, and Baturynska. "Process parameter optimization for

A conceptual framework for powder bed fusion additive manufacturing using the finite element technique and machine learning. Procedia Cirp 67 (2018): 227-232.
[32] Zavareh Bozorgasl and Mohammad J. Dehghani. 2014 4th International Conference on Computer and Knowledge Engineering (ICCKE), pp. 86-89. "2-D DOA estimation in wireless location system via sparse representation." IEEE, 2014.
[33] Christian Gobert et al. "Application of supervised machine learning for defect detection during metallic powder bed fusion additive manufacturing using high resolution imaging." Additive Manufacturing 21 (2018): 517-528.
[34] Emily Sullivan. The British Journal for the Philosophy of Science, "Understanding from machine learning models." (2022).
[35] Paul, Arindam, et al. 2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA) "A real-time iterative machine learning approach for temperature profile prediction in additive manufacturing processes." IEEE, 2019.
[36] Aryati Bakri, Mahyar, and Amini. Journal of Information Technology & Information Systems Research (JITISR) 9.2 (2015): 121-135. "Cloud computing adoption by SMEs in the Malaysia: A multi-perspective framework based on DOI theory and TOE framework."
[37] Applied Sciences 11.24 (2021): 11910 Mahmoud, Dalia, et al. "Applications of Machine Learning in Process Monitoring and Controls of L-PBF Additive Manufacturing: A Review."
[38] Mahyar Amini. "The factors that influence on adoption of cloud computing for small and medium enterprises." (2014).
[39] Khanzadeh, Mojtaba, et al. Journal of Manufacturing Science and Engineering 140.3 (2018) "Quantifying geometric accuracy with unsupervised machine learning: Using self-organizing map on fused filament fabrication additive manufacturing parts."
[40]Amini, Mahyar, et al. "MAHAMGOSTAR.COM AS A CASE STUDY FOR ADOPTING LARAVEL FRAMEWORK AS THE BEST PROGRAMMING TOOLS FOR PHP BASED WEB DEVELOPMENT"
Journal of Innovation & Knowledge, ISSN (2021): 100-110, "FOR SMALL AND MEDIUM ENTERPRISES."
[41] Xiaoli Zhang, Liu, Rui, and Sen Liu. The International Journal of Advanced Manufacturing Technology 113.7 (2021): 1943-1958. "A physics-informed machine learning model for porosity analysis in laser powder bed fusion additive manufacturing."
[42]Amini, Mahyar, et al. "Development of an instrument for assessing the impact

of environmental context on adoption of cloud computing for small and medium enterprises." Australian Journal of Basic and Applied Sciences (AJBAS) 8.10 (2014): 129-135.
[43] Nojeem, Lolade, et al. "Machine Learning assisted response circles for additive manufacturing concerning metal dust bed fusion." International Journal of Management System and Applied Science 17.11 (2023): 491-501.
[44]Lily Poursoltan, Hamid Esmaeeli, Negar Jahanbakhsh Javid, Golmohammadi, and Amir-Mohammad. Industrial Engineering and Management Systems 15, no. 4 (2016): 385-395. "Modeling and analyzing one vendor-multiple retailers VMI SC using Stackelberg game theory."
[45]Motalo, Kubura, et al. "evaluating artificial intelligence effects on additive manufacturing by machine learning procedure." Journal of Basis Applied Science and Management System 13.6 (2023): 3196-3205.
[46]Fu, Yanzhou, et al. Journal of Manufacturing Processes 75 (2022): 693-710 "Machine learning algorithms for defect detection in metal laser-based additive manufacturing: a review."
[47]Amini, Mahyar, et al. "The role of top manager behaviours on adoption of cloud computing for small and medium enterprises." Australian Journal of Basic and Applied Sciences (AJBAS) 8.1 (2014): 490-498.
[48] George-Christopher Vosniakos and Emmanuel Stathatos. The International Journal of Advanced Manufacturing Technology 104.5 (2019): 1967-1984. "Real-time simulation for long paths in laser-based additive manufacturing: a machine learning approach."
[49] Mahyar, Nazli Sadat Safavi, and Amini. International Journal of Information Technology & Information Systems 5.15 (2013): 1–23. "Critical success factors for ERP implementation."
[50] Petrich, Jan. et al. "Multi-modal sensor fusion with machine learning for data-driven process monitoring for additive manufacturing." Additive Manufacturing 48 (2021): 102364.
[51]Amini, Mahyar, et al. "Agricultural development in IRAN base on cloud computing theory." International Journal of Engineering Research & Technology (IJERT) 2.6 (2013): 796-801.
[52] Snow, Zackary, et al. "Toward in-situ flaw detection in laser powder bed fusion additive manufacturing through layerwise imagery and machine learning." Journal of Manufacturing Systems 59 (2021): 12-26.
[53]Amini, Mahyar, et al. "Types of cloud computing (public and private) that transform the organization more effectively." International Journal of Engineering Research & Technology (IJERT) 2.5 (2013): 1263-1269.
[54] Akhil, V., et al. "Image data-based surface texture characterization and prediction using machine learning approaches for additive manufacturing." Journal of Computing and Information Science in Engineering 20.2 (2020): 021010.
[55] Mahyar, Nazli Sadat Safavi, and Amini. International Journal of Innovation & Management Science Research 1.61 (2013): 1–5. "Cloud Computing Transform the Way of IT Delivers Services to the Organizations."
[56] Douard, Amélina, et al. "An example of machine learning applied in additive manufacturing." IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), 2018. IEEE, 2018.

[57] Mahyar, Nazli Sadat Safavi, and Amini. International Journal of Computer Science and Information Technologies 6.11 (2014): 25–30. "A Dynamic SLA Aware Heuristic Solution For IaaS Cloud Placement Problem Without Migration."
[58] Ding, Donghong, et al. "The first step towards intelligent wire arc additive manufacturing: An automatic bead modelling system using machine learning through industrial information integration." Journal of Industrial Information Integration 23 (2021): 100218.
[59] Mahyar, Nazli Sadat Safavi, and Amini. International Journal of Computer Science and Information Technologies 6.11 (2014): 52-57. "A Dynamic SLA Aware Solution For IaaS Cloud Placement Problem Using Simulated Annealing."
[60] Zhang, Wentai, et al. "Machine learning enabled powder spreading process map for metal additive manufacturing (AM)." International Solid Freeform Fabrication Symposium, 2017. Austin University of Texas, 2017.
[61] Sharifani, Koosha, Amini, Mahyar, Akbari, Yaser, and Aghajanzadeh Godarzi, Javad. International Journal of Science and Information System Research 12.9 (2022): 20–44. "Operating Machine Learning across Natural Language Processing Techniques for Improvement of Fabricated News Model."
[62] Du, Y., T. Mukherjee, and T. DebRoy. Applied Materials Today 24 (2021): 101123. "Physics-informed machine learning and mechanistic modeling of additive manufacturing to reduce defects."
[63]Sadat Safavi, Nazli, et al. "An effective model for evaluating organizational risk and cost in ERP implementation by SME." IOSR Journal of Business and Management (IOSR-JBM) 10.6 (2013): 70-75.
[64] Chen, Lequn, et al. "Rapid surface defect identification for additive manufacturing with in-situ point cloud processing and machine learning." Virtual and Physical Prototyping 16.1 (2021): 50-67.
[65] Mahyar Amini, Nazli, Nor Hidayati Zakaria, and Sadat Safavi. World Applied Sciences Journal (WASJ) 31.9 (2014): 1669-1676. "The risk analysis of system selection and business process re-engineering towards the success of enterprise resource planning project for small and medium enterprise."
[66] Ahmed Yaseer and Heping Chen. Journal of Manufacturing Processes 70 (2021): 543-552. "Machine learning based layer roughness modeling in robotic additive manufacturing."
[67] Seyyed Amir Ali Javadinia, Nazli, Mahyar Amini, and Sadat Safavi. International Journal of Advanced Research in IT and Engineering (IJARIE) 3.1 (2014): 1–8. "The determinant of adoption of enterprise resource planning for small and medium enterprises in Iran."
[68] Chen, Lequn, et al. "Rapid surface defect identification for additive manufacturing with in-situ point cloud processing and machine learning." Virtual and Physical Prototyping 16.1 (2021): 50-67.
[69] Safavi, Nazli Sadat, et al. "An effective model for evaluating organizational risk and cost in ERP implementation by SME." IOSR Journal of Business and Management (IOSR-JBM) 10.6 (2013): 61-66.
[70] Guo, Shenghan, et al. "Machine learning for metal additive manufacturing: Towards a physics-informed data-driven paradigm." Journal of Manufacturing Systems 62 (2022): 145-163.

[71] International Journal of Computer Engineering & Sciences (IJCES) 1.2 (2011): 28-35 Khoshraftar, Alireza, et al. "Improving The CRM System In Healthcare Organization."
[72] Computer Methods in Applied Mechanics and Engineering 362 (2020): 112734. Ren, K., et al. "Thermal field prediction for laser scanning paths in laser aided additive manufacturing by physics-based machine learning."
[73] Abdollahzadegan, A., Moshfegh Gohary, M., Che Hussin, A. R., & Amini, M. (2013). the organizational success criteria that are essential for SMEs to embrace cloud computing. Information Systems Research and Innovation Journal (JISRI), 4(1), 67-74.
[74] Jinhui Yan, Zeliang Liu, Zhu, and Qiming. Computational Mechanics 67.2 (2021): 619-635. "Machine learning for metal additive manufacturing: predicting temperature and melt pool fluid dynamics using physics-informed neural networks."
[75] Ali Rahmani, Mahyar, and Amini. "Agricultural databases evaluation with machine learning procedure." Australian Journal of Engineering and Applied Science 8.6 (2023): 39–50.
[76]Newiduom, Ladson, et al. "Machine learning model improvement based on additive manufacturing process for melt pool warmth predicting." International Journal of Applied Science and Engineering Research 15.6 (2023): 240-250.
[77] Jackson, Keypi, et al. "Increasing substantial productivity in additive manufacturing procedures by expending machine learning method." International Journal of Mechanical Engineering and Science Study
10.8 (2023): 363-372.
[78] Opuiyo, Atora, et al. "Evaluating additive manufacturing procedure which was using artificial intelligence." International Journal of Science and Information System 13.5 (2023): 107-116.
[79] Ali Rahmani, Mahyar, and Amini. Journal 9.12 (2023): 240-250. "Machine learning process evaluating damage classification of composites."
[80] Yun, Chidi, et al. "Machine learning attitude towards temperature report forecast in additive manufacturing procedures." International Journal of Basis Applied Science and Study 9.12 (2023): 287-297.
[81]Shun, Miki, et al. "Incorporated design aimed at additive manufacturing through machine learning." International Journal of Information System and Innovation Adoption 16.10 (2023): 823-831.