The Initialization Factor: Understanding its Impact on Active Learning for Analog Circuit Design

Sezin Kircali Ata*, Zhi Hui Kong†, Anusha James*, Lile Cai*, Kiat Seng Yeo†,
Khin Mi Mi Aung*, Chuan Sheng Foo*§ and Ashish James*
*Institute for Infocomm Research (I^2R), A*STAR, Singapore
§Centre for Frontier AI Research (CFAR), A*STAR, Singapore
†Singapore University of Technology and Design (SUTD)

***Abstract*—Active learning, which aims to enhance modeling efficiency, precision, and cost effectiveness through selective labeling, is emerging as a promising strategy for analog circuit modeling. However, analog circuits are constrained by strict functional and technological limitations, resulting in scarcity of data for modeling, and additional data acquisition involves expensive and time-consuming simulations. For efficient and effective active learning for analog circuit modeling, our research analyzes data-driven initial sampling techniques which lays the foundation for the active learning process. Our experiments reveal that these initialization strategies expedite the learning process, decrease the demand for extensive simulations, and produces more accurate models. Furthermore, the results demonstrate that active learning techniques, which uniformly sample the design space, tend to benefit from distance-based initialization technique.**

***Index Terms*—Regression, Active Learning, Analog Circuits, Operational Amplifier, Folded Cascode Operational Amplifier**

I. Introduction

Active learning (AL) is a machine learning paradigm that focuses on optimizing the learning process by actively selecting and labeling the most informative data points for modeling [1]. In contrast to traditional supervised learning [2], that uses a fixed labeled dataset, active learning works iteratively and dynamically to identify and label the most informative samples, aiming to reduce labeling costs while maintaining or surpassing model's performance compared to traditional approaches. This is particularly attractive for analog/radio frequency (RF) circuit modeling due to its capability to efficiently capture intricate circuit behavior by reducing labeling costs and flexibly adapting to diverse design scenarios [3]–[5].

The growing complexity of analog circuit design due to continuous advancements in integrated circuit technology has led to increased design iterations and reduced productivity, sparking significant interest in automated analog circuit sizing [6]–[8]. In this context, the primary objective is to create an accurate predictive model that estimates circuit performance metrics based on the circuit configurations, which can essentially be formulated as a regression problem. However, while some prior studies have characterized this as a classification problem [4], [9], [10], our primary focus here is to minimize the number of simulations required for training by treating it as a regression problem. Such a problem formulation enhances a circuit designer's capacity to navigate complex design spaces more efficiently and also offers the potential to significantly reduce the need for extensive and costly design iterations.

Active learning is generally carried out iteratively, commencing with the training of the model on an initially labeled set of samples. It then focuses on selecting unlabeled data points that have the most impact on improving the model's performance through a query strategy. The selected points are labeled and subsequently used to retrain the model together with previously labeled samples. This iterative process continues until a predefined stopping criterion (e.g., labeling budget, model performance threshold) is satisfied. Since the initial set of labeled samples establishes the foundation for the entire active learning process (as observed in image classification [11]), their selection is of paramount importance on multiple fronts. Firstly, it is important to carefully consider the representativeness and quality of the initial samples as it impacts the accuracy of the initial model, which in turn steers subsequent active learning iterations. Secondly, since the initial samples essentially serve as seeds within the available input space, ensuring a comprehensive coverage of input space through these seeds can provide better guidance during the query stage in the active learning process.

To the best of our knowledge, there has been no prior investigation into the impact of initial samples in a regression setting. In this work, we analyze the impact of initial samples in active learning within the context of analog circuit design. Specifically, by adopting a data-driven approach to initial sample generation, we showcase their influence to accelerate the active learning process while modeling analog circuits. Furthermore, this study compares multiple active learning techniques and highlights the significance of well-chosen initial samples for modeling analog circuits.

II. Literature Review

In analog circuit design, active learning has been employed primarily to achieve diverse goals by formulating it as a classification problem. In [12], an active learning approach to identify feasible design spaces based on a committee of support vector machines (SVMs) is introduced. In [13], an SVM based active learning approach that efficiently reduces the training dataset size by addressing uncertainty in the learning process is introduced. In [4], Bayesian deep active learning (BDAL) is introduced as a solution to minimize data requirements for analog circuit performance classification.

Active learning techniques have also been applied in analog circuit design by framing it as a regression problem.

In [14], authors introduce an active learning algorithm for modeling integrated circuit behavior using multiple neural network regressions models. In [15], the reliability analysis of large analog and mixed-signal systems, with considerations of process variations and transistor aging effects, is discussed. In [16], the effectiveness of neural networks and other modeling techniques in active learning for global surrogate models in complex computer simulations is discussed.

The concept of selecting initial samples for labeling, known as *initialization*, has garnered increased attention in the field of active learning. In a recent study [17], the effects of various initialization techniques within the active learning framework for classification are investigated. These techniques encompass random sampling and entropy-based uncertainty sampling method called MaxE [18].

Notably, the impact of initialization in regression settings remains relatively unexplored, with majority of studies employing conventional initialization methods such as random [19] and K-Means clustering [20], [21].

III. Active Learning for Modeling Analog Circuit Performance

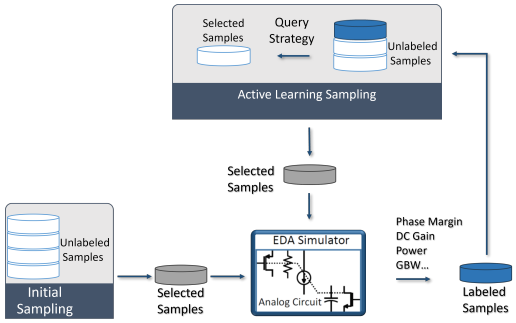


Fig. 1: Block diagram of active learning framework with initialization for analog circuit design

We propose an active learning framework for analog circuit design with an initialization sampling phase designed for regression tasks, as depicted in Fig. 1. For a given circuit $\mathbf{C}$, we consider D design parameters as $x_C \in \mathbb{R}^D$ and the corresponding figure of merit (FOM), $y_C \in \mathbb{R}^P$ characterized by P performance metrics such as Phase Margin, DC Gain, Power, and Gain Bandwidth Product (GBW).

In the context of analog circuit design, the design parameters x_C have predefined ranges $[x_C^{min}, x_C^{max}]$ with known step sizes, enabling the systematic generation of all potential circuit parameters. This unlabeled set U consisting of all possible circuit configurations can serve as the starting point for an active learning process. The initial sampling technique selects N circuit configurations x_C^i where i varies from 1 to N, from this unlabeled pool U for simulations by an EDA simulator. The corresponding FOM y_C^i obtained from this simulations will create a labeled set of samples $\{(x_C^i, y_C^i)_{i=1}^N\}$. Following this initialization phase, active learning query strategy is iteratively applied on the remaining unlabeled circuit configurations to identify the most informative samples for labeling by the EDA simulator. This iterative process of active learning sampling and labeling by the EDA simulator continues till a predefined stopping criteria like the labeling budget and/or model performance thresholds are satisfied.

Initialization Factor - Initial Sampling

In this paper, we explore the initialization phase of active learning by examining four distinct data-driven techniques, namely the traditional random and K-Means clustering [22] as well as K-Center Greedy [23] and latin hypercube sampling (LHS) [24]. Brief description of each methods is given below.

1) Random Sampling: In this paper, we consider random sampling as selecting an initial set of N samples from all the possible circuit configurations U where each configuration has an equal probability of being chosen. This ensures that the initial sample is a representative and unbiased subset of the possible circuit configurations. However, due to its lack of control in intricate parameter spaces, there is a possibility of selecting outliers or redundant points closely clustered in local regions, potentially leading to sub-optimal solutions and slowing down the active learning process.

2) K-Means Sampling: This involves exploiting the underlying structure of the possible circuit configurations and grouping them with similar characteristics into $K(= N)$ clusters [22]. The objective of K-Means clustering is to minimize the sum of squared distances represented by:

$$J = \sum_{i=1}^{|U|} \sum_{j=1}^{K} ||x^i - c^j||^2 \tag{1}$$

where K denotes the number of clusters, x^i signifies the ith data point in the multi-dimensional circuit configuration space, and c^j the jth cluster centroid. While K-Means clustering typically exhibits better representativeness when compared to random sampling, its effectiveness depends on the choice of the value for K and suitability of partitioning the data into meaningful clusters.

3) K-Center Greedy Sampling: This aims to maximize the minimum distance between selected samples to promote diversity and coverage. Initially, we include the mean point of x_C into an empty set of selected points S, while U contains all the remaining possible circuit configurations. This approach iteratively selects $K(= N)$ points to be added into S based on the following equation:

$$x^* = \arg\max_{x \in U} \min_{z \in S} |x - z| \tag{2}$$

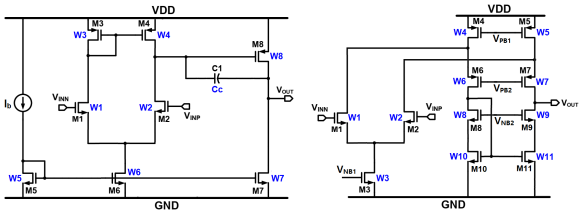
where x^* is the selected data point. The focus here is on maximizing the minimum distance between U and S thereby ensuring the initial samples to cover a broad range of data space, thus promoting diversity [23]. While this algorithm targets to promote diversity, it tends to favor samples situated at the boundaries of the design space.

4) Latin Hypercube Sampling (LHS): The previous initialization techniques require the generation of all possible circuit configurations U for sampling, a computationally expensive and at times infeasible as the number of circuit parameters increases. In contrast, LHS divides the circuit configuration space into stratified and evenly spaced intervals along each of the D dimensions [24]. It ensures that each interval contains at most one sample, and the samples are distributed uniformly

across the entire circuit design space. LHS allows for a well-structured selection of initial samples, addressing the need for diversity and thorough coverage of the design space by generating them from a uniform distribution within the predefined range of $[x_C^{min}, x_C^{max})$.

IV. Results & Discussion

Initialization's impact on active learning in two analog circuit topologies of varying complexity, as depicted in Fig. 2, is analyzed in this section.



(a) Two-stage OpAmp (b) Folded Cascode OpAmp

Fig. 2: Schematics of analog circuits

1) Two-Stage Miller-Compensated Operational Amplifier (OpAmp): Fig. 2(a) depicts the schematic of a two-stage Miller-compensated OpAmp design [25] implemented in 45nm TSMC CMOS technology. In this example, there are 9 design parameters ($x_C \in \mathbb{R}^9$), which are dedicated to the sizing of individual transistors. These design parameters, indicated in blue in Fig. 2, can be adjusted within predefined ranges and step sizes as given in Table I, resulting in ∼27 million possible circuit configurations. The biasing conditions remain constant throughout the experiment to ensure reasonable predictability in circuit behaviour and consistency in performance analysis. The FOM is defined by Phase Margin (PM), DC Gain, Gain Bandwidth Product (GBW) and Power, i.e. $y_C \in \mathbb{R}^4$.

TABLE I: Range and step sizes for Two-Stage OpAmp

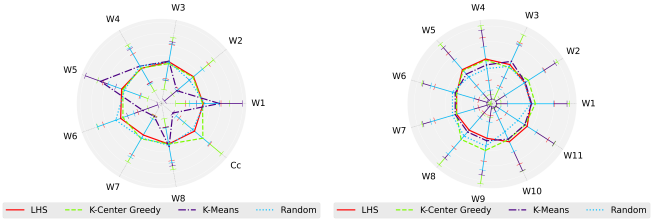
	W1(m)	W2(m)	W3(m)	W4(m)	W5(m)	W6(m)	W7(m)	W8(m)	Cc(F)
Min.	6μ	6μ	9μ	9μ	6μ	6μ	15μ	9μ	3p
Max.	21μ	21μ	42μ	42μ	21μ	21μ	21μ	42μ	6p
Step	3μ	3μ	3μ	3μ	3μ	3μ	3μ	3μ	1p

2) Folded Cascode Operational Amplifier (OpAmp): The schematic diagram of the folded cascode OpAmp [26] [27] adopted in our analysis is portrayed in Fig. 2(b). In this circuit, we employ 11 design parameters with symmetries across transistor widths (W1=W2, W4=W5, W6=W7, W8=W9, W10=W11), with each parameter varying across a larger range of 1μm to 50μm in steps of 0.5μm ($x_C \in \mathbb{R}^{11}$). This leads to ∼900 billion possible circuit configurations. Similar to its Miller-compensated counterpart, biasing conditions are also kept unchanged over the course of the experiment. FOMs for this circuit are identical to that of the previous circuit ($y_C \in \mathbb{R}^4$). We note that the experiments involving this circuit topology are considerably more complex than the previous circuit in view of the higher number of design parameters.

A. Analysis of Initial Samples

In this paper, our emphasis is on modeling the circuit with minimal data and thereby the initialization phase in the active learning framework commences with a relatively small sample set size of 16 for both circuits. For two-stage OpAmp, with relatively small unlabeled pool size $|U|$ (27 million configurations), the initial samples are selected from the entire U. However, due to the significantly larger number of possible configurations for folded cascode OpAmp (900 billion), we create a random sample U of 30 million circuit configurations. We would like to highlight that LHS operates on the defined range $[x_C^{min}, x_C^{max})$ for the circuits as opposed to U, thereby leading to lower computational complexity.

Fig. 3 depicts the distribution of mean values of the initial samples for both circuits using the initialization techniques discussed in Section III. The vertical bars indicate the standard deviation among the initial 16 samples. It is evident that clustering approaches like K-Means do not evenly explore the design space, unlike LHS, which inherently aim for more representativeness and uniformity among samples. The higher standard deviation in Fig. 3 for K-Center Greedy signifies its inclination for sampling from the design space boundaries. Further, random sampling leads to non-uniform coverage of the circuit design space.



(a) Two-stage OpAmp (b) Folded Cascode OpAmp

Fig. 3: Initial sample distribution for active learning

In active learning, the selection of diverse samples is crucial for improving the model's ability to generalize across different scenarios, enhancing its robustness and accuracy. To assess diversity, we analyze the average inter-sample Euclidean distance for the different initialization techniques, as shown in Table II. K-Center Greedy sampling, designed to maximize the distance between samples, demonstrates the highest diversity with the largest inter-sample distance, followed by LHS and random sampling, which provide a more uniform distribution. K-Means sampling, which results in local clusters, yields the lowest diversity with short inter-sample distance.

TABLE II: Average Euclidean distance across initializations

$d = \|x_{ci} - x_{cj}\|_{x_c}$	Random	K-Means	K-Center Greedy	LHS
Two-Stage OpAmp	3.00E-05	2.30E-05	4.40E-05	2.90E-05
Folded Cascode OpAmp	6.30E-05	5.30E-05	10.8E-05	6.10E-05

B. Analysis of Initialization on AL Techniques

Following the initialization phase, multiple AL iterations are performed, with each iteration involving sampling and simulation of 16 samples. In our experiments we employ 4 different AL techniques based on their query strategy, to select 16 samples from the unlabeled set U. These samples are ranked based on their scores computed using the corresponding acquisition function. Details are provided below:

1) Random: Baseline approach where samples are selected randomly from U.

2) Diversity: K-Center Greedy: This seeks to enhance sample diversity by selecting data points from U that has the maximum distance from the labeled samples S [23]. This is based on the acquisition function:

$$A_{KCG}(x) = \min_{x \in U} d(x, S) \tag{3}$$

where $d(x, S)$ computes the Euclidean distance from a point x to the nearest labeled point in S. This function is maximized to select the data point x_i that is farthest from S.

3) Uncertainty: Query by Committee (QBC): This selects samples that an ensemble of models find most uncertain [28]. The corresponding acquisition function is

$$A_{QBC}(x) = \sum_{j=1}^{P} \left(\frac{1}{M} \sum_{i=1}^{M} (f_{ij}(x) - \bar{f}_j(x))^2 \right) \tag{4}$$

where M denotes the number of models in ensemble, $f_{ij}(x)$ is the prediction made by model i for FOM j, and $\bar{f}_j(x)$ is the mean prediction of models for FOM j for sample x in U. This function is maximized to select the data point x that has the maximum uncertainty.

4) Combination of diversity and uncertainty: Clustering uncertainty-weighted embeddings (CLUE): This selects samples by considering both ensemble model uncertainty and diversity by utilizing uncertainty-weighted embeddings [29]. The acquisition function is defined as:

$$A_{CLUE}(x) = A_{KCG}(\mathbf{\Phi(x)} \circ \mathbf{A_{QBC}(x)}) \tag{5}$$

where $\Phi(x)$ denotes average embeddings over M models as $\Phi(x) = \frac{1}{M} \sum_{i=1}^{M} \Phi_i(x)$, operation $\circ$ represents elementwise product, and A_{QBC} and A_{KCG} computes model uncertainty and diversity, respectively. In CLUE, we opted for K-Center Greedy over K-Means++ [30] to maximize diversity.

Considering the resource intensive circuit design simulations, we have set a labeling budget of 16 AL iterations as the stopping criteria. The entire process is repeated across 2 random seeds to validate the impact of initialization on active learning. The performance is evaluated using a fully connected neural network that trains on independently standardized x_C and y_C. The model performance reported in this paper is on a test set consisting of 320 samples, evenly generated from the 4 distinct initialization techniques described in Section III.

Fig. 4 shows the average RMSE on the test set for the two-stage OpAmp, across 4 FOMs and various AL techniques iterations. For clarity, only a selection from the final 16 AL iterations are shown. Consistently, K-Means initialization yields the worst performance across the AL techniques. In most AL techniques, K-Center Greedy and LHS initializations outperform others, with the latter being the best for QBC. CLUE AL technique that combines diversity and uncertainity sampling appears to be less responsive to initializations.

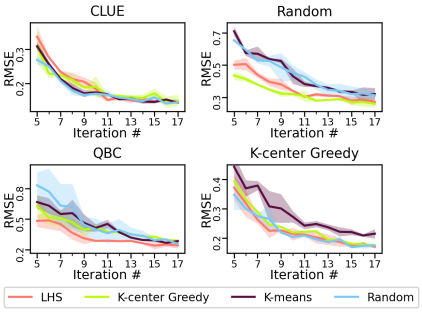


Fig. 4: Prediction Performance for Two-Stage OpAmp

For folded cascode OpAmp, the RMSE prediction performance on the test set for the FOMs is depicted in Fig. 5. Here, the uncertainty-based AL method QBC, prone to simulation errors, is excluded from comparison. Like the prior case, K-Means initialization performs the worst for the 4 FOMs. Remarkably, K-center greedy initialization excels over random and CLUE AL methods, with LHS being the best performer for K-Center greedy AL technique.

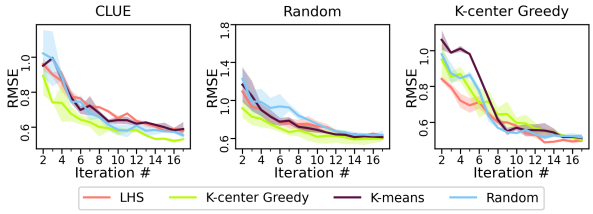
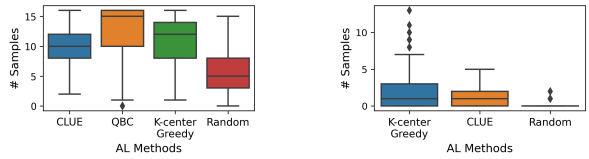


Fig. 5: Prediction Performance for Folded Cascode OpAmp

To emphasize active learning techniques that venture into boundaries of the circuit design space, their respective distributions across initializations are shown in Fig. 6. Boundaries of design space correspond to selecting x_C^{min} and/or x_C^{max} during AL iterations. Techniques prioritizing diversity (e.g., K-Center Greedy) or uncertainty (e.g., QBC) are more inclined to select boundary points compared to others.



(a) Two-Stage OpAmp (b) Folded Cascode OpAmp

Fig. 6: Boundary selection across iterations for AL techniques

Overall, K-Center Greedy, which yielded most diverse initial samples (Table II), often led to faster learning with fewer AL iterations. Nevertheless, it tends to explore the design space boundaries and encounters challenges with growing circuit design parameters. In contrast, LHS's well-distributed initial samples maintain consistent performance across AL methods. Additionally, distance-based initializers like K-Center Greedy improve AL techniques that uniformly sample the design space, whereas boundary-sampling AL techniques benefit from uniform initializers like LHS.

V. Conclusions

This study examines how initialization impacts active learning in analog circuit design. We demonstrate that data-driven initialization methods, selecting informative and diverse samples, accelerate learning, lessen simulation needs, and improve complex behavior modeling. K-Center Greedy initialization speeds up learning but faces challenges with boundary exploration and increased computational complexity with more design parameters. Conversely, LHS provides stable performance with representative sampling. The findings enhance active learning, suggesting ways to reduce data constraints and boost modeling efficiency and accuracy, with wider implications.

Acknowledgement

This research is supported by Agency for Science, Technology and Research (A*STAR) under its AME Programmatic Funds (Grant No. A19E3b0099 and A20H6b0151).

REFERENCES

[1] A. Tharwat and W. Schenck, "A survey on active learning: State-of-the-art, practical challenges and research directions," *Mathematics*, vol. 11, no. 4, 2023.

[2] P. Cunningham, C. Matthieu, and S. J. Delany, "Supervised learning," in *Machine Learning Techniques for Multimedia: Case Studies on Organization and Retrieval*. Springer Berlin Heidelberg, 2008, pp. 21–49.

[3] S. Noureen, I. H. Syed, A. A. Abdellatif, M. Q. Mehmood, and Y. Massoud, "Reducing complexity and data-set-size through physics inspired tandem neural network," in *IEEE Intl. Symp. Circuits and Systems (ISCAS)*, 2023, pp. 1–5.

[4] L. Zhang, S. Raju, A. James, R. Dutta, G. Fournier, D. Lancry, K. C. T. Chuan, V. R. Chandrasekhar, and C. S. Foo, "Bayesian deep active learning for analog circuit performance classification," in *IEEE Intl. Symp. Circuits and Systems (ISCAS)*, 2022, pp. 3018–3022.

[5] E. Afacan, N. Lourenço, R. Martins, and G. Dündar, "Review: Machine learning techniques in analog/rf integrated circuit design, synthesis, layout, and test," *Integration*, vol. 77, pp. 113–130, 2021.

[6] R. A. Rutenbar, G. G. E. Gielen, and J. Roychowdhury, "Hierarchical modeling, optimization, and synthesis for system-level analog and rf designs," *Proceedings of the IEEE*, vol. 95, no. 3, pp. 640–669, 2007.

[7] Z. Zhao and L. Zhang, "Deep reinforcement learning for analog circuit sizing," in *IEEE Intl. Symp. Circuits and Systems (ISCAS)*, 2020, pp. 1–5.

[8] R. Dutta, A. James, S. Raju, Y.-J. Jeon, C. S. Foo, and K. Tshun Chuan Chai, "Automated deep learning platform for accelerated analog circuit design," in *IEEE 35th Intl. System-on-Chip Conf. (SOCC)*, 2022, pp. 1–5.

[9] F. De Bernardinis, M. Jordan, and A. SangiovanniVincentelli, "Support vector machines for analog circuit performance representation," in *Design Automation Conf. (DAC)*, 2003, pp. 964–969.

[10] F. D. Bernardinis, P. Nuzzo, and A. S. Vincentelli, "Robust system-level design with analog platforms," in *IEEE/ACM Intl. Conf. Computer Aided Design*, 2006, pp. 334–341.

[11] Q. Jin, M. Yuan, S. Li, H. Wang, and M. W. andZhijian Song, "Cold-start active learning for image classification," *Information Sciences*, vol. 616, pp. 16–36, 2022.

[12] M. Ding and R. Vemur, "An active learning scheme using support vector machines for analog circuit feasibility classification," in *18th Intl. Conf. VLSI Design held jointly with 4th Intl. Conf. Embedded Systems Design*, 2005, pp. 528–534.

[13] H. Lin and P. Li, "Circuit performance classification with active learning guided sampling for support vector machines," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 9, pp. 1467–1480, 2015.

[14] V. Grosu, L. Goras, E. David, and G. Pelz, "Using neural network based active learning for modelling integrated circuits behavior," in *Intl. Symp. Signals, Circuits and Systems (ISSCS)*, 2023, pp. 1–4.

[15] E. Maricau, D. De Jonghe, and G. Gielen, "Hierarchical analog circuit reliability analysis using multivariate nonlinear regression and active learning sample selection," in *Design, Automation & Test Europe Conf. & Exhibition (DATE)*, 2012, pp. 745–750.

[16] D. Gorissen, L. De Tommasi, K. Crombecq, and T. Dhaene, "Sequential modeling of a low noise amplifier with neural networks and active learning," *Neural Computing and Applications*, vol. 18, no. 5, pp. 485–494, Jun 2009.

[17] Y. Yang and M. Loog, "To actively initialize active learning," *Pattern Recognition*, vol. 131, p. 108836, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S003132032200317X

[18] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," in *SIGIR*, B. W. Croft and C. J. van Rijsbergen, Eds. London: Springer London, 1994, pp. 3–12.

[19] G. Citovsky, G. DeSalvo, C. Gentile, L. Karydas, A. Rajagopalan, A. Rostamizadeh, and S. Kumar, "Batch active learning at scale," *ArXiv*, vol. abs/2107.14263, 2021.

[20] D. Wu, "Pool-based sequential active learning for regression," *IEEE Trans. Neural Networks and Learning Systems*, vol. 30, no. 5, pp. 1348–1359, 2019.

[21] Z. Liu and D. Wu, "Integrating informativeness, representativeness and diversity in pool-based sequential active learning for regression," in *Intl. Joint Conf. Neural Networks (IJCNN)*, 2020, pp. 1–7.

[22] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Comput. Surv.*, vol. 31, no. 3, p. 264–323, Sep 1999.

[23] O. Sener and S. Savarese, "A geometric approach to active learning for convolutional neural networks," *ArXiv*, vol. abs/1708.00489, 2017.

[24] M. Mckay, R. Beckman, and W. Conover, "A comparison of three methods for selecting vales of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, pp. 239–245, 05 1979.

[25] G. Palmisano and G. Palumbo, "A compensation strategy for two-stage cmos op-amps based on current buffer," in *IEEE Trans. Circuits and Systems (part I)*, vol. 44, no. 3, 1997, p. 257–262.

[26] S. Mallya and J. Nevin, "Design procedures for a fully differential folded-cascode cmos operational amplifier," *IEEE Journal of Solid-State Circuits*, vol. 24, no. 6, pp. 1737–1740, 1989.

[27] D. B. Ribner, M. A. Copeland, and M. Milkovic, "80MHz low offset fully-differential and single-ended opamps," in *IEEE Custom Integrated Circuits Conf.*, 1983, pp. 74–75.

[28] H. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Fifth Annual ACM Workshop on Computational Learning Theory*. Publ by ACM, 1992, pp. 287–294.

[29] V. Prabhu, A. Chandrasekaran, K. Saenko, and J. Hoffman, "Active domain adaptation via clustering uncertainty-weighted embeddings," in *IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*. Los Alamitos, CA, USA: IEEE Computer Society, oct 2021, pp. 8485–8494. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/ICCV48922.2021.00839

[30] D. Arthur and S. Vassilvitskii, "K-means++ the advantages of careful seeding," in *Eighteenth Annual ACM-SIAM Symp. Discrete Algorithms*, 2007, pp. 1027–1035.