Bayesian Optimization in Hybrid Data Envelopment Analysis and Stacking Approach for Optimizing Supplier Selection

Samaneh Azarakhsh*.Stefano Ferrari.**

*Department of Economics, Management and Quantitative Methods, University of Milan, Milan, Italy (e-mails: samaneh.azarakhsh@studenti.unimi.it)

**Department of Computer Science, University of Milan, Milan, Italy(e-mails: stefano.ferrari@unimi.it)

Abstract: This study addresses the prescriptive limitations of traditional Data Envelopment Analysis (DEA) by proposing a novel hybrid framework, the Hybrid DEA-Stacking-Bayesian Model, to enhance supply chain optimization. An input-oriented DEA model under Variable Returns to Scale (VRS) is employed to minimize inputs while maintaining output levels. The efficiency scores obtained from the DEA model serve as outputs for machine learning models trained to predict the efficiency of decision-making units based on their input-output profiles. The hybrid approach integrates DEA with Stacking algorithms and incorporates Bayesian optimization for hyperparameter tuning. Results demonstrate that Bayesian optimization significantly enhances performance across all meta-learners, with XGBoost achieving the highest accuracy, improvement in R^2, and yielding the lowest error metrics. Comparative analyses highlight XGBoost as the best-performing model, followed by kNN, RF, and DT, confirming the effectiveness of the proposed framework. The integration of DEA and Stacking improves supplier selection by evaluating efficiency and enhancing prediction accuracy through model combination. This work provides a robust decision-support tool for supply chain management, laying the groundwork for future research on supplier selection.

Keywords: Data Envelopment Analysis, Supplier Selection, Bayesian Optimization, Stacking Meta Learner

1. INTRODUCTION

Supply Chain Management (SCM) is both an art and science aimed at optimizing procurement, production, and distribution to enhance efficiency, reduce costs, and improve customer satisfaction. In the evolving landscape, businesses face challenges from shifting market demands and sustainability requirements. Traditional optimization methods have limitations in addressing the complex, stochastic nature of supply chain operations, highlighting the need for innovative approaches. The critical role that Supplier Selection (SS) plays in effective Logistics and Supply Chain Management (LSCM) has attracted a lot of attention. It is a critical process in logistics and supply chain management. It entails locating, assessing, and selecting vendors that can offer the products or services required to satisfy an organization's needs. The goal is to find suppliers that offer the best combination of quality, cost, reliability, and service. A non-parametric technique for assessing effectiveness, Data Envelopment Analysis (DEA), has emerged as a crucial instrument for SCM decision-making, especially when it comes to finding sustainable suppliers. (Vörösmarty & Dobos, 2020).

A review of the existing literature highlights various methods used for supplier assessment and selection, each offering distinct advantages. However, integrating and extending these approaches to meet specific goals is crucial. By presenting a novel hybrid model for supply chain optimization and addressing the prescriptive limits of traditional DEA, this paper makes a substantial contribution. The proposed model combines DEA with Stacking algorithms and incorporates Bayesian optimization for hyperparameter tuning, focusing on enhancing both accuracy and efficiency. This framework provides a more comprehensive solution than traditional methods, addressing critical challenges in supply chain management and offering valuable insights for improving operational performance.

This research offers several key contributions to the literature: 1- Introduces an input-oriented VRS DEA model aimed at reducing input usage while keeping output levels constant. 2- Combines DEA with Stacking algorithms to enhance prediction accuracy. 3- Implements Bayesian optimization for hyperparameter tuning to boost model performance. 4- Evaluates the impact of Bayesian optimization on hyperparameter tuning outcomes. 5- Compares the performance of four Stacking models. 6- Demonstrates the effectiveness of the hybrid DEA-Stacking-Bayesian approach for supply chain optimization. 7- Provides a decision-support framework to help optimize resource allocation and improve supply chain operations.

This paper is laid out as follows: A summary of literature is given in Section 2, the methodology is described in Section 3. The results of the implementation are presented in Section 4. Section 5 wraps up the conclusion and future directions.

2. LITERATURE REVIEW

2.3 Integrated contributions in Supplier Selection

DEA and ML integration in SCM provides a comprehensive solution, where DEA's efficiency analysis informs the strategic decision-making process, identifying areas for improvement and benchmarking. Concurrently, ML algorithms use this efficiency data, along with historical trends and patterns, to predict future supply chain behaviors, demand fluctuations, and potential inefficiencies (Ni et al., 2020). In study by Islam et al., (2024), the demands are using Deep Learning (DL),

generated suppliers' weights using hybrid Principal Component Analysis (PCA), and compared various forecasting models for allocating orders. Khan et al., (2024) have studied the contributions of many elements to economic resilience and created an output-oriented DEA model. The K-Means technique is then used for clustering and analysis. After that, clustering and analysis are done using the K-Means algorithm. A study by Boubaker et al., (2023), introduces a two-stage DEA model in conjunction with support vector regression (SVM), neural network regression, random forest regression (RFR), and gene expression programming (GEP). In order to enable ranking comparisons of Micro, Small, and Medium-Sized Enterprises (MSMEs), the work developed a technique to derive a common set of weights (CSW) through regression analysis for DEA evaluations. In a study by Abdulla et al., (2023), they proposed an integrated approach that uses machine learning models like Decision Tree (DT) and Random Forest (RF) in conjunction with the MARCOS (Measurement of Alternatives and Ranking according to Compromise Solution) method for evaluation purposes.

Moslemi et al., (2022) study took into account the pharmaceutical supply chain's entire structure and suggested a BSC-based structure as a network architecture to establish performance benchmarks for the chain as a whole. They proposed the Anderson-Peterson Neuro-DEA model to rank the most efficient units and assess the supply chain network's efficacy. To evaluate and choose the best service, Abdulla et al., (2019), combined AHP and decision trees. Specifically, the decision tree classifier was employed in this study to identify the most crucial criteria. Principal component analysis was taken into consideration in (Davoudabadi et al., 2020), in order to minimize dimensions, and a model based on DEA-Entropy was integrated to evaluate the weights assigned to every criterion. For the purpose of selecting and classifying suppliers with respect to sustainability and resilience, Tavassoli & Ghandehari, (2023) proposed a fuzzy super efficiency-DEA and fuzzy-MIP model.

3. METHODOLOGY

3.1 Data Envelopment Analysis

DEA is a method that uses linear programming techniques to evaluate the effectiveness of Decision Making Units (DMUs).

3.1.1 Categories of Data Envelopment Analysis

Based on their orientation and returns to scale, DEA models can be roughly divided into two types: The goal of input-oriented DEA is to reduce inputs while keeping outputs constant. It assesses how much input quantity may be decreased proportionately without affecting output levels. The goal of output-oriented DEA is to maximize outputs while maintaining constant inputs.

The Constant Returns to Scale (CRS) hypothesis states that an increase in inputs will cause outputs to rise proportionately. When every DMU is functioning at its ideal scale, this model can be used. The concept of variable returns to scale, or VRS, acknowledges that there may be fluctuations in the proportionality between inputs and outputs. When DMUs function at several scales, this model is helpful because it permits rising, falling, or constant returns to scale (Pradhan & Kamble, 2015).

3.1.2 Input-oriented Constant Returns to Scale

By comparing the inputs that decision-making units (DMUs) use to generate specific outputs, the Input-oriented Constant Returns to Scale (CRS) model assesses the relative efficiency of DMUs. The CRS assumption, which is appropriate for evaluating entities functioning at their optimal size, suggests that raising inputs by a specific percentage would raise outputs by the same percentage. Mathematically we have:

Given n DMUs, where every DMU_j generates s outputs Y_{rj}using m inputs X_{ij}. The input and output matrices are denoted by X and Y, respectively, with dimensions m × n for X and s × n for Y. Objective: Minimize the efficiency score θ of a test DMU_o subject to the following constraints. The following is a formalization of the CRS-DEA model:

$$\min \theta \tag{1}$$

subject to:

$$\sum_{j=1}^{n} \lambda_j X_{ij} \leq \theta X_{io}, \quad \forall i = 1, \dots, m \tag{2}$$

$$\sum_{j=1}^{n} \lambda_j Y_{rj} \leq Y_{ro}, \quad \forall r = 1, \dots, s \tag{3}$$

$$\lambda_j \geq 0, \quad \forall j = 1, \dots, n \tag{4}$$

where the model aims to reduce the efficiency score of the DMU under evaluation, denoted by θ. λ_j are the weights assigned to each DMU_j in constructing the virtual DMU that serves as the reference point or "composite" DMU. X_{io} and $Y_{ro}$are the inputs and outputs, respectively, of the DMU_o under evaluation. X_{ij} and Y_{rj} are the inputs and outputs, respectively, of the DMU_j in the dataset.

The equations (1) and (2) define a virtual DMU as a linear combination of the available DMUs. The efficiency of DMU_o is then evaluated challenging this virtual DMU to achieve the same output using less input resources. The efficiency score θ, which indicates the percentage that inputs may be decreased while maintaining the same output level as DMU_o is minimized by the objective function in this formulation.

The set of constraints (2) ensures that the weighted aggregate of inputs (using weights λ_j) for all DMUs are not more than the scaled down inputs of the DMU being evaluated. The constraint (3) ensure that the virtual DMU constructed from the weighted aggregation of all DMUs can produce at least as much output as the DMU under evaluation. The objective is to minimize the efficiency score θ of the target DMU by finding a set of λ_j weights that allow the target DMU's weighted sum of inputs to be as small as possible relative to its outputs, while ensuring that the constructed virtual DMU remains within the production-possibility set determined by all DMUs.

3.1.3 Input-oriented Variable Returns to Scale

The basic DEA model is extended by the input-oriented variable returns to scale (VRS) DEA model, which does not make the assumption that the scale of operations is fixed.

Unlike the CRS model which assumes that inputs and outputs change proportionally, the VRS model allows for non-proportional scaling, accommodating entities that might have increasing or decreasing returns to scale. The VRS model differs from the CRS model in that it incorporates an additional restriction to account for Variable Returns to Scale. It is possible to formulate the VRS DEA model for either input or output orientation. For illustration, we'll look at an input-oriented VRS model here.

The input-oriented VRS DEA model's mathematical formulation is identical to that of the CRS DEA model, (1)–(4), with the following extra restriction:

$$\sum_{j=1}^{n} \lambda_j = 1 \tag{5}$$

where (5) is introduced to model Variable Returns to Scale, distinguishing it from the CRS model. The constraint (5) imposes the VRS condition by allowing for non-proportional increases or decreases in the scale of operations. The non-negativity constraints (4) require that the weights assigned to each DMU be non-negative. In the VRS DEA model, efficiency scores will still range between 0 and 1 for an input-oriented model. A score of 1 indicates that the DMU is operating efficiently relative to others in the dataset, considering variable returns to scale. The VRS model is particularly useful when analyzing DMUs of different sizes or when the assumption of proportional scaling CRS is not suitable. It allows for more nuanced analysis and can indicate whether DMUs would benefit from scaling up or down their operations.

3.2 Stacking Meta Learner

Stacking is a versatile ensemble method, where a secondary model, often referred to as the meta-learner, is trained to integrate the predictions from differen base models, called first-level learners. In this approach, the first-level learners are trained on the original dataset, producing outputs that serve as features for the meta-learner. Meanwhile, the original labels are retained as the target values for the new dataset used to train the meta-learner (Zhou, 2012). Stacking in classification is a hierarchical ensemble learning method comprising two levels of modeling: base learners and a meta-learner. Consider a training dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$ where $x_i \in R^d$ indicates the i-th feature vector with d features, and y_i is the corresponding target label. In the first level, *M* base learners $f_1, f_2, \dots, f_M$ are trained on D to minimize their respective loss functions L_m. Mathematically, each base learner is optimized as (6).

$$f_m = \text{argmin}_{f \in F_m} L_m\big(y, f(x)\big) \text{ for } m \in \{1, 2, \dots, M\} \tag{6}$$

After training, the base learners generate predictions that form an output matrix P. Each row of P contains the predictions of all *M* base learners for a single instance, such that *P* is structured as (7):

$$P = \begin{bmatrix} f_1(x_1) & \cdots & f_M(x_1) \\ \vdots & \ddots & \vdots \\ f_1(x_N) & \cdots & f_M(x_N) \end{bmatrix} \tag{7}$$

In the second level, a new dataset $D_{meta} = \{(P_i, y_i)\}_{i=1}^{N}$ is constructed, where (8) represents the predictions of the base learners for the *i*-th instance, and y_i remains the original label. This new dataset is used to train the meta-learner g(·), which learns to combine the outputs of the base learners.

$$P_i = [f_1(x_i), f_2(x_i), \dots f_M(x_i)\,]^T \tag{8}$$

The meta-learner is trained to minimize a loss function L_g, formulated as (9).

$$g = \text{argmin}_{g \in G} L_g(y, g(P)) \tag{9}$$

The meta-learner effectively assigns weights or establishes relationships among the base learners' predictions to improve overall accuracy.

3.2.1 K-Nearest Neighbors

The foundation of the k-nearest neighbor (kNN) technique is the idea that objects in the input space with similar properties would yield results that are similar. In contrast to conventional learning techniques, it merely keeps the dataset in its original form without implementing a systematic training procedure. kNN finds the dataset's k most similar points while evaluating a new data point. The test point is allocated by the algorithm to the class that is most prevalent among these neighbors for categorization. In regression, the average of their values is determined and used as the expected result.
The output for the test instance is determined by aggregating the target score of the k-nearest neighbors. Typically, the prediction is the average of these values, and alternatively, in weighted kNN, closer neighbors are given higher importance, using weights inversely proportional to their distances:
$\hat{y} = \frac{\sum_{i \in N} w_i . y_i}{\sum_{i \in N} w_i}$, where $w_i = \frac{1}{d(x, x_i)}$.

3.2.2 Extreme Gradient Boosting

A gradient boosting framework for decision tree construction with excellent scalability is called Extreme Gradient Boosting (XGB). XGB iteratively creates an additive model of the target function by minimizing the loss function $L_{xgb} = \sum_{i=1}^{n} L\big(y_i, F(x_i)\big) + \sum_{m=1}^{M} \Omega(h_m)$, just like gradient boosting does. To control tree complexity, XGB uses a modified version of the loss function, $\Omega(h) = \gamma T + 12\ \lambda \|W\|^2$, and only uses decision trees as its base classifier.

3.2.3 Random Forest

An ensemble learning technique called Random Forest Regression (RFR) builds several decision trees using arbitrary selections of the data and attributes. The average of all the trees' predictions is the final prediction for regression problems. Given a dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$, T bootstrap samples D_t are drawn with replacement. Each tree t is trained on D_t using a random subset of features at each split. The prediction from tree t is: $\hat{y}^{(t)} = \frac{1}{|L_t|} \sum_{i \in L_t} y_i$, where L_t represents the data points in the leaf node of tree t. The final prediction for a test point x is the average of all tree predictions as $\hat{y} = \frac{1}{T} \sum_{t=1}^{T} \hat{y}^{(t)}$. This averaging reduces variance and overfitting, providing a more stable and accurate regression model.

3.2.4 Decision Tree

A Decision Tree (DT), which is used to make decisions or predictions, looks like a flowchart. It consists of nodes that symbolize choices or attribute tests, branches that display the outcomes of these choices, and leaf nodes that show forecasts or final results. Every internal node denotes an attribute test, every branch displays the test's outcome, and every leaf node represents a continuous value or a class label.

A DT splits the data recursively to minimize the variance in each partition. Given a dataset *D*, the algorithm selects splits based on minimizing the variance within the resulting subsets. The variance at a node is: $Var(D) = \frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{y})^2$, where $\bar{y}$ is the mean target value in the dataset. For each split, the goal is to minimize the weighted sum of variances in the left and right subsets as $Var_{\text{split}}(D) = \frac{|D_L|}{|D|}\, Var(D_L) + \frac{|D_R|}{|D|}\, Var(D_R)$, where D_L and D_R are the left and right subsets. Prediction for a test point is the mean target value in the leaf node: $\hat{y}(x_{\text{test}}) = \frac{1}{|D_{leaf}|}\sum_{i \in D_{leaf}} y_i$.

3.3 Bayesian Hyperparameter Tuning

Bayesian optimization is a powerful approach for tackling problems where determining the extrema of a function is computationally intensive. It is particularly useful for optimizing functions that lack a closed-form representation, are costly to compute, have derivatives that are challenging to evaluate, or are non-convex in nature. In contrast to grid or random search, which aimlessly search the parameter space, Bayesian optimization takes into account past data and iteratively chooses hyperparameter combinations that are most likely to enhance the objective function (J. Wu et al., 2019). The process of Bayesian hyperparameter tuning involve three main steps as following.

3.3.1 Surrogate Model

A surrogate model is used in Bayesian optimization to approximate the objective function *f(x)*, which assesses a model's performance given a collection of hyperparameters x. By serving as a stand-in for the real objective function and being computationally efficient, this surrogate model allows for fewer costly evaluations. A Tree-structured Parzen Estimator (TPE) is used as the surrogate model in the optimization process. The relationship between the objective function *f(x)* and hyperparameters x is roughly represented by the TPE. The core idea behind TPE is to estimate two separate probability distributions:

$p(x \mid y \leq y^*)$: The hyperparameter distribution, where the objective function value *y* is less than or equal to a threshold value y^*.

$p(x \mid y > y^*)$: The hyperparameter distribution where the objective function value y exceeds the threshold y^*.

Here, *x* indicates the hyperparameter vector, and *y* is the objective function value associated with a given set of hyperparameters.

3.3.2 Acquisition Function

Which combination of hyperparameters should be examined next is decided by the acquisition function. It seeks to strike a balance between exploitation and discovery. The Upper Confidence Bound (UCB) or Expected Improvement (EI) is a popular acquisition function, defined as (10):

$$\alpha(x) = \frac{p(x \mid y \leq y^*)}{p(x \mid y > y^*)} \quad (10)$$

This function will prefer areas of the hyperparameter space where the model predicts lower objective values (better performance) while exploring regions with high uncertainty.

3.3.3 Irritation and Optimization

In this step, the objective function is evaluated for a given hyperparameter set x_{next}, and the results are used to update the surrogate model. The goal is to minimize the objective function y(x) as $y_{\text{next}} = f(x_{\text{next}})$. Iteratively, the process is carried out until convergence or the maximum amount of trials is reached after the surrogate model is modified to incorporate the new information based on the evaluation.

4. IMPLEMENTATION AND RESULTS

This approach aims to increase the accuracy and stability of prediction models for efficiency assessment while creating a reliable and consistent process for choosing suppliers. A central aspect of this method is the integration of the VRS DEA model with four base learners (KNN, Bagging, SVM, GB), four stacking meta-learners (RF, XGB, KNN, and DT) and the application of Bayesian hyperparameter tuning. By employing these techniques, the method seeks to significantly enhance model accuracy, optimize predictive performance, and ensure consistent and reliable efficiency evaluations.

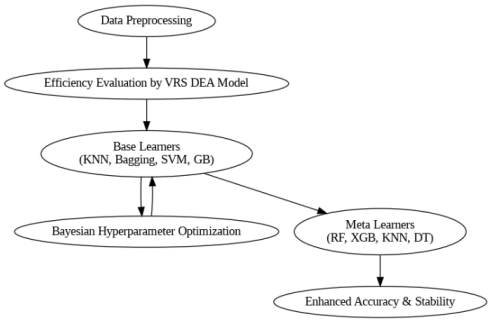

Figure 1. Workflow of methodology

We begin by preprocessing the data before building the models. Next, we develop a VRS DEA model to evaluate efficiency. Following this, each meta-learner algorithm is applied using default settings, without hyperparameter tuning, to establish baseline performance. To achieve better results, we optimize the parameters using Bayesian hyperparameter optimization, which effectively enhances model accuracy and stability. The models are then rebuilt with these optimized settings for final evaluation. Finally, the outputs from the base estimators are used as inputs for the stacking models, which further enhance overall prediction accuracy. The results demonstrate that hyperparameter tuning, coupled with Bayesian optimization, significantly improves the robustness and performance of the proposed methodology, as shown in the workflow in Figure 1.

4.1 Dataset

We implemented the model on a dataset of evaluated suppliers, containing 35 suppliers and 12 attributes, is curated to evaluate

and select suppliers based on criteria essential for maintaining high supply chain standards. Feature selection is key in DEA as it directly influences the efficiency scores of DMUs. Choosing appropriate inputs and outputs ensures meaningful comparisons and reliable results, depending on the analysis focus, such as cost-efficiency, resilience, or environmental effects. Metrics are normalized (0-10) for accurate comparison. Based on feature correlations, company priorities, and expert input, the selected DEA features include three inputs (employees, price, delivery time) and four outputs (quality, serviceability, reputation, flexibility). 80% of the data was used for training, and the remaining 20% was set aside for validation.

4.2 Evaluation Metrics

Evaluating machine learning (ML) models involves quantitatively measuring their performance using specific criteria and mathematical formulations. These criteria are essential for understanding how well a model predicts or categorizes data. The average prediction errors for regression models are frequently measured using Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). MAE provides the average absolute difference between observed actual results and predictions of the model, calculated as: MAE = $\frac{1}{n}\sum_{i=1}^{n}|y_i - \tilde{y}_i|$.

MSE quantifies the average of the squares of the errors, or the average squared difference between the estimated values and the actual value, calculated as: MSE = $\frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2$. RMSE is a common used metric to assess the accuracy of a predictive model, calculated as: RMSE = $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2}$.

The coefficient of determination, or R-squared (R^2), is the ratio of the dependent variable's variance that can be predicted from the independent variables, calculated as: $R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$

In the metrics mentioned above, n represents the number of observations, y_i represents the actual value for the i-th observation, $\tilde{y}_i$ represents the predicted value for the i-th observation, and $\bar{y}$ represents the observed data mean.

4.3 Results and Analysis

In the context of the input oriented VRS model, a score of 1 indicates that the supplier is on the frontier of the production potential set and is deemed fully efficient, meaning they are maximizing output from a given set of.

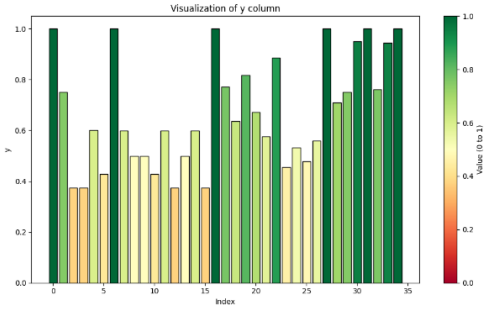

Figure 2. VSR DEA efficiency scores of suppliers

Table 1. Stacking Meta Learner Evaluation

Meta Learners	Before Bayesian Implementation				After Bayesian Implementation			
	R2	MSE	RMSE	MAE	R2	MSE	RMSE	MAE
KNN	0.76	0.010	0.10	0.10	0.91	0.004	0.06	0.05
XGB	0.56	0.019	0.14	0.13	0.92	0.003	0.06	0.05
RF	0.62	0.016	0.13	0.12	0.91	0.004	0.06	0.06
DT	0.53	0.020	0.14	0.12	0.89	0.004	0.07	0.06

Scores below 1 suggest inefficiency or lower efficiency, where the supplier could potentially increase output or reduce inputs without detriment to production levels. In the Figure 2, the efficiency scores of suppliers are demonstrated. The efficiency estimates obtained from the DEA model have been used as output dataset for the ML models which are then trained to predict the efficiency of the DMUs given their characteristics or input-output profiles.

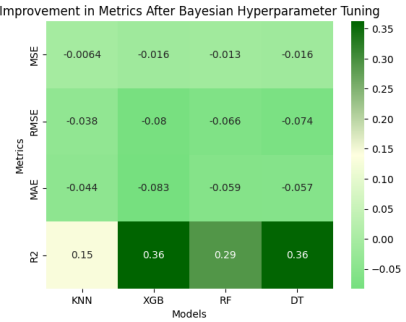

Figure 3. Improvements in metrics by Bayesian optimization

According to Table 1, and the heatmap in Figure 3 (showing the improvement across models), the results indicate that Bayesian hyperparameter optimization substantially improves the performance of all meta-learners considered in this study. Improved R^2 and lower errors enhance the accuracy of efficiency predictions, leading to better supplier differentiation and more informed selection decisions. The optimization process led to improvements across all metrics, with XGB showing the most significant increase in performance. Specifically, XGB saw a notable improvement in R^2, rising from 0.56 to 0.92, a 0.36 increase, demonstrating the greatest enhancement in model accuracy. Additionally, XGB achieved the best results in terms of R^2 (0.92), MSE (0.003), RMSE (0.06), and MAE (0.05), highlighting its superior performance after Bayesian optimization. Following XGB, DT exhibited the second most significant improvement, with R^2 increasing from 0.53 to 0.89, a 0.36 increase. Although its performance after tuning (R^2=0.89) was slightly lower than the other meta-learners, DT had a major boost in prediction accuracy.

After DT, RF showed a 0.29 improvement in results. Both RF and kNN achieved an R^2 of 0.91, indicating comparable performance in terms of explained variance. However, when comparing the models based on error metrics (MSE, RMSE and MAE), RF showed slightly higher values (MSE=0.004, RMSE = 0.07, MAE = 0.06) compared to kNN (MSE=0.004, RMSE = 0.06, MAE = 0.05). This suggests that kNN performed slightly better than RF in terms of prediction accuracy, despite both models exhibiting the same R^2 value.

Thus, kNN ranks as the second-best meta-learner in terms of overall performance and error metrics. XGBoost outperformed other models due to its robustness to noisy data, effective handling of outliers, built-in regularization, and scalability, making it less sensitive to human bias in data entry.

5. CONCLUSION AND FUTURE STUDIES

The study demonstrates the effectiveness of utilizing VRS DEA efficiency estimates as input for machine learning models to predict the efficiency of DMUs based on their characteristics. The integration of Bayesian hyperparameter optimization significantly enhanced the predictive performance of all meta-learners. The best-performing model in this study was XGB, which had the greatest improvement among them and continuously outperformed other models across all evaluation metrics. However, one of the key challenges encountered was dealing with noisy data, primarily caused by human bias in filling out information and providing data. These inconsistencies introduced variability that required careful preprocessing to maintain the reliability of the results.

For future research exploration of multi-stage DEA models to capture the interdependence of processes and provide a more detailed efficiency assessment is recommend. Integrating feature selection algorithms into the framework could enhance model interpretability and reduce computational complexity by identifying the most influential variables. Comparing the hybrid DEA-Stacking-Bayesian model to alternative decision-making methods would also be valuable to highlight its advantages and limitations in various scenarios.

ACKNOWLEDGMENT

This work was supported in part by project SERICS (PE00000014) under the NRRP MUR program funded by the EU - NGEU. Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the Italian MUR. Neither the European Union nor the Italian MUR can be held responsible for them.

REFERENCES

Abdulla, A., Baryannis, G., & Badi, I. (2019). *Weighting the Key Features Affecting Supplier Selection using Machine Learning Techniques*. MATHEMATICS & COMPUTER SCIENCE. https://doi.org/10.20944/preprints201912.0154.v1

Abdulla, A., Baryannis, G., & Badi, I. (2023). An integrated machine learning and MARCOS method for supplier evaluation and selection. *Decision Analytics Journal*, *9*, 100342. https://doi.org/10.1016/j.dajour.2023.100342

Boubaker, S., Le, T. D. Q., Ngo, T., & Manita, R. (2023). Predicting the performance of MSMEs: A hybrid DEA-machine learning approach. *Annals of Operations Research*. https://doi.org/10.1007/s10479-023-05230-8

Davoudabadi, R., Mousavi, S. M., & Sharifi, E. (2020). An integrated weighting and ranking model based on entropy, DEA and PCA considering two aggregation approaches for resilient supplier selection problem. *Journal of Computational Science*, *40*, 101074. https://doi.org/10.1016/j.jocs.2019.101074

Fotova Čiković, K., Martinčević, I., & Lozić, J. (2022). Application of Data Envelopment Analysis (DEA) in the Selection of Sustainable Suppliers: A Review and Bibliometric Analysis. *Sustainability*, *14*(11), 6672. https://doi.org/10.3390/su14116672

Islam, S., Amin, S. H., & Wardley, L. J. (2024). A supplier selection & order allocation planning framework by integrating deep learning, principal component analysis, and optimization techniques. *Expert Systems with Applications*, *235*, 121121. https://doi.org/10.1016/j.eswa.2023.121121

Khan, W., Islam, Md. S., Ullah Ibne Hossain, N., & Fazio, S. (2024). Measuring economic resilience of manufacturing organization leveraging integrated data envelopment analysis (DEA)-machine learning approach. *International Journal of Management Science and Engineering Management*, *19*(3), 228–238. https://doi.org/10.1080/17509653.2023.2267505

Moslemi, S., Mirzazadeh, A., Weber, G.-W., & Sobhanallahi, M. A. (2022). Integration of neural network and AP-NDEA model for performance evaluation of sustainable pharmaceutical supply chain. *OPSEARCH*, *59*(3), 1116–1157. https://doi.org/10.1007/s12597-021-00561-1

Ni, D., Xiao, Z., & Lim, M. K. (2020). A systematic review of the research trends of machine learning in supply chain management. *International Journal of Machine Learning and Cybernetics*, *11*(7), 1463–1482. https://doi.org/10.1007/s13042-019-01050-0

Pradhan, A. K., & Kamble, A. A. (2015). Efficiency Measurement and Benchmarking: An Application of Data Envelopment Analysis to Select Multi Brand Retail Firms in India. *Journal of Commerce and Management Thought*, *6*(2), 258. https://doi.org/10.5958/0976-478X.2015.00016.6

Tavassoli, M., & Ghandehari, M. (2023). Classification and forecasting of sustainable-resilience suppliers via developing a novel fuzzy MIP model and DEA in the presence of zero data. *Operations Management Research*. https://doi.org/10.1007/s12063-023-00401-z

Vörösmarty, G., & Dobos, I. (2020). A literature review of sustainable supplier evaluation with Data Envelopment Analysis. *Journal of Cleaner Production*, *264*, 121672. https://doi.org/10.1016/j.jclepro.2020.121672

Wu, J., Chen, X.-Y., Zhang, H., Xiong, L.-D., Lei, H., & Deng, S.-H. (2019). *Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimization*.

Zhou, Z.-H. (2012). Ensemble Methods. *Machine Learning*.