Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Machine Learning: Mathematical Framework, Practical Uses, and Future Research Paths

Rinki Singh[1], Deepti Goyal[1], Ritu Punhani[1], Nidhi Sharma[2], Anju Rani[2], Shalini Puri[3]

[1]*Department of Computer Science and Engineering, Satyug Darshan Institute of Engineering & Technology, Faridabad (Haryana), India*
[2]*(CAD) Computer Application Department, DPG institute of technology and management, Gurugram, India*
[3]*Department of Information Technology, Manipal University Jaipur, India*
Email: rinkisingh18@gmail.com

This paper offers a comprehensive overview of machine learning, covering its theoretical foundations, practical applications, and future research direc-tions. It explores the fundamental concepts, historical context, and mathe-matical underpinnings of machine learning, including linear algebra, calcu-lus, probability theory, and optimization The study grapples with crucial is-sues facing the discipline, including the reliability of datasets, the transparen-cy of models, and the moral implications of the research. It outlines a general framework for developing machine learning models and examines emerging research areas like causal AI, federated learning, and energy-efficient algo-rithms. This work highlights the field's ongoing evolution and its potential to address complex real-world problems.

Keywords: Machine Learning, Linear Algebra, Calculus, Probability and Statistics, Gra-dient, Interpretability, Data quality and quantity, Mathematical model.

1. Introduction

Artificial intelligence's branch, machine learning, has transformed the landscape of data analysis and computer science. Systems can accomplish objectives by discerning trends and drawing conclusions, rather than following specific directives, allowing them to operate autonomously. [1,13]. This approach marks a shift from traditional programming, allowing computers to learn, predict, and decide based on data. Machine learning mimics aspects of human cognition like pattern recognition and adaptation. AI systems excel at rapidly processing and examining enormous datasets, enabling them to extract valuable insights and forecast results in intricate, information-dense scenarios more efficiently than human counterparts. Fig. 1 illustrates the different terms associated with machine learning, from its historical conceptions to its current issues and concerns.

1.1 Historical Context

The mid-1900s saw the beginning of machine learning research. In 1959, Arthur Samuel created a self-improving checkers programme and came up with the phrase. The transition from theory to practice was facilitated by advances in processing power and data availability, with the emergence of complex models in the 1990s and 2000s propelling technical advancement [2].

1.2 Fundamental Concept

Machine learning efficiently handles complex, data-rich problems and adapts to new patterns, but requires substantial data and can be less interpretable [3]. Traditional programming offers more control but struggles with highly complex issues. Key differences are represented in Table 1.

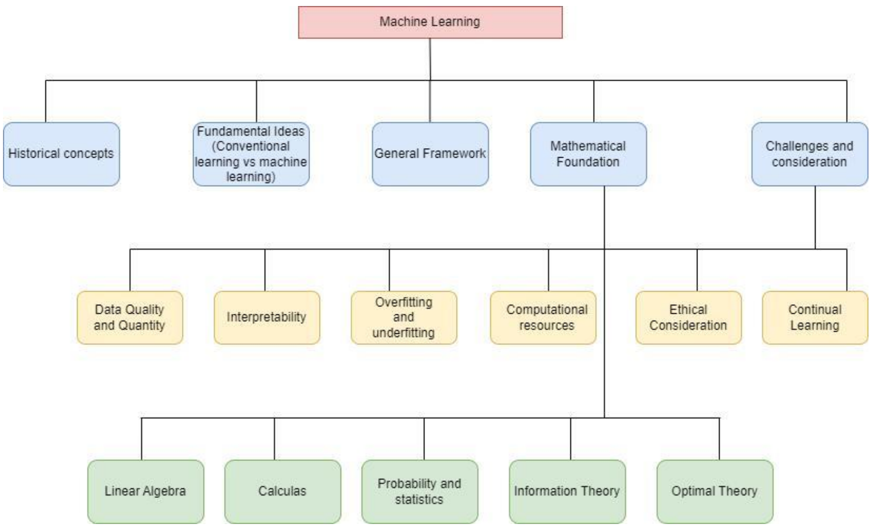

Fig. 1. Machine Learning Related Terms

Table 1. Conventional Programming vs Machine Learning.

Aspect	Conventional Programming	Machine Learning
Approach	Explicit instructions	Data-driven
Programming	Rules are manually coded	Algorithms learn patterns from data
Flexibility	Limited to predefined rules	Can adapt to new patterns in data
Input	Fixed set of parameters	Large amounts of training data
Output	Predictable based on rules	Probabilistic predictions
Maintenance	Regular updates to rules	Retraining with new data
Complexity	Increases with problem complexity	Can handle complex patterns more easily
Interpretability	Generally, more transparent	Often considered a "black box"
Development time	Faster for simple, well-defined problems	Longer initial development, faster for complex problems

Expertise required	Domain-specific knowledge	Data science and machine learning skills

1.3 Mathematical Foundation of Machine Learning

Machine learning relies on a robust mathematical foundation. It integrates linear algebra for data manipulation, calculus for optimization, probability theory for uncertainty modeling, information theory for data analysis, and optimization techniques for algorithm refinement [19]. These mathematical tools enable the creation, training, and evaluation of models that can learn patterns, make predictions, and solve complex problems across diverse domains.

Linear Algebra: Linear algebra is fundamental in machine learning for representing and manipulating data, as well as for performing computations efficiently [18].

Key concepts include:

1. Vectors: Ordered lists of numbers, often representing features of data points.

   Example: $x = [x_1, x_2, ..., x_n]$

2. Matrices: 2D arrays of numbers, used to represent collections of data points or linear transformations. Example: $A = [a_{ij}]$, in which j is the column and i is the row.

3. Matrix operations:

   - Addition: $X = Y + Z$, where $x_{ij} = y_{ij} + z_{ij}$
   - Multiplication: $X = YZ$, where $c_{ij} = \Sigma_k y_{ik}z_{kj}$
   - Transpose: Y^T, where $(Y^T)_{ij} = y_{ji}$

4. Eigenvalues and eigenvectors: $Av = \lambda v$, where A is a square matrix, v is an eigenvector, and λ is the corresponding eigenvalue.

Key techniques in the field, including PCA and a range of neural network models, are built upon these essential ideas.

Calculus: Calculus is essential for optimization in machine learning, particularly for training models using gradient-based methods [20].

Key concepts include:

1. Derivatives: A function's derivative represents its instantaneous rate of change. It's expressed as $f'(x) = \lim[h\to 0]\ (f(x+h) - f(x)) / h$ for a function f(x).

2. Partial derivatives: Calculate rate of change for a single variable, keeping all other variables fixed in a multivariable function.

   $\partial f/\partial x_i = \lim[h\to 0]\ (f(x_1, ..., x_i+h, ..., x_n) - f(x_1, ..., x_i, ..., x_n)) / h$

3. Gradient: Vector of partial derivatives.

   $\nabla f = [\partial f/\partial x_1, \partial f/\partial x_2, ..., \partial f/\partial x_n]$

4. Chain rule: $(f(g(x)))' = f'(g(x)) * g'(x)$

These concepts are used in algorithms like gradient descent for optimizing model parameters.

Probability Theory and Statistics: Probability and statistics are crucial for understanding data distributions, making inferences, and evaluating model performance [17].

Key concepts include:

1. Probability distributions: e.g., Normal distribution $N(\mu, \sigma^2)$ with probability density function:

$$f(x) = (1 / (\sigma\sqrt{(2\pi)})) * e^{(-(x-\mu)^2 / (2\sigma^2))}$$

2. Bayes' theorem: $P(A|B) = (P(B|A) * P(A)) / P(B)$

3. Maximum Likelihood Estimation (MLE):

$\theta_{MLE} = \text{argmax}[\theta]\ P(X|\theta)$, where X is the observed data and θ are the model parameters

These concepts are used in various machine learning algorithms, including Naive Bayes classifiers and probabilistic graphical models.

Information Theory: Information theory provides a foundation for understanding data compression, transmission, and the amount of information in data.

Key concepts include:

1. Entropy: The mean quantity of data contained within a stochastic variable X is quantified by this metric.

$$H(X) = -\Sigma[x]\ P(x) \log_2(P(x))$$

2. Mutual Information: The mutual impact of variables X and Y on one another serves as a measure of their interconnectedness.

$$I(X;Y) = \Sigma[x,y]\ P(x,y) \log_2(P(x,y) / (P(x)P(y)))$$

3. Kullback-Leibler (KL) Divergence: Quantifies the dissimilarity between probability distributions P and Q.$D_{KL}(P||Q) = \Sigma[x]\ P(x) \log_2(P(x) / Q(x))$

These concepts are used in feature selection, model evaluation, and in algorithms like decision trees.

Optimization Theory: Optimization theory is crucial for developing and understanding algorithms that efficiently find the best parameters for a given model.

Key concepts include:

1. Objective function: The function to be minimized or maximized.

Example: Mean Squared Error (MSE) $= (1/n)\ \Sigma_i (y_i - \hat{y}_i)^2$.

2. Gradient descent: An iterative optimization algorithm.

$\theta_{t+1} = \theta_t - \alpha\nabla J(\theta_t)$, where α is the learning rate and J is the objective function

3. Convex optimization:

Addresses issues characterized by convex optimization criteria.A function f exhibits convexity when the following inequality holds: $f(\alpha p + (1-\alpha) q) \leq \alpha f(p) + (1-\alpha) f(q)$, where p and q are arbitrary inputs, and α is any real number between 0 and 1, inclusive. Addresses issues characterized by convex optimization criteria.

4. Lagrange multipliers: Used for constrained optimization problems.

$L(x, \lambda) = f(x) - \lambda g(x)$, in which $f(x)$ is optimized under the condition that $g(x)$ equals zero.

1.4 Challenges and Considerations

Machine learning faces challenges in data quality, interpretability, overfitting, ethics, and resource management. Ongoing research addresses these issues, emphasizing the need for a multidisciplinary approach combining technical expertise with ethical considerations and domain knowledge.

Data Quantity and Quality: The Machine learning model effectiveness is largely determined by their training datasets. The quality and characteristics of input data significantly shape how well these systems perform. This encompasses both the quantity and quality of data [5].

1. Quantity: More data generally leads to better model performance, but this relationship isn't always linear. The concept of learning curves helps visualize this: Error = a + b/n^c, The coefficients 'a', 'b', and 'c' shift according to the unique challenge and model at hand, with 'n' denoting the complete set of training instances.es.

2. Quality: Data quality issues include noise, bias, and class imbalance. Biased data can lead to unfair models. For instance, if a recruitment model is trained on historically biased hiring data, it may perpetuate this bias: $P(\text{hire} \mid \text{qualified, minority}) \neq P(\text{hire} \mid \text{qualified, non-minority})$.

Interpretability: Many advanced models, especially deep neural networks, are hard to identify. This "black box" aspect may be problematic in fields like healthcare or finance where understanding the decision-making process is crucial techniques to improve interpretability include:

1. LIME (Local Interpretable Model-agnostic Explanations)

2. SHAP (SHapley Additive exPlanations)

3. Attention mechanisms in neural networks For instance, SHAP values φ_i for feature i are calculated as: $\varphi_i = \Sigma[S \subseteq N \setminus \{i\}]\ |S|!(|N|-|S|-1)! / |N|!\ [fx(S \cup \{i\}) - fx(S)]$ in which N is the collection of all features and fx is the model output.

4. Overfitting and Underfitting: Models that memorize training data, including errors, often fail on new examples due to overfitting. Overly simplistic models fail to grasp complex data relationships, resulting in underfitting.

Computational Resources: Training large models, especially deep neural networks, requires significant computational power. This has implications for cost, energy consumption, and accessibility [6]. The time complexity of training can often be expressed as: O(ndi), The variables n, d, and i represent training sample count, feature quantity, and iteration number respectively. Techniques to address this include:

1. Distributed computing
2. Model compression
3. Transfer learning

Ethical Considerations: Ethical issues in machine learning include privacy, fairness, accountability, and potential reinforcement of societal biases. Fairness can be mathematically formulated in various ways, such as demographic parity: $P(\hat{Y}=1 \mid X=0) = P(\hat{Y}=1 \mid X=1)$ Where $\hat{Y}$ is the model prediction and X is a protected feature.

Continual Learning: The goal of continual learning is to create systems capable of acquiring knowledge from continuous data streams while retaining previously learned information. Because of the "catastrophic forgetting" issue, this is difficult. Elastic Weight Consolidation technique, introducing a punitive element to the error function: $L(\theta) = L\_B(\theta) + \lambda/2 \ \Sigma_i \ F_i(\theta_i - \theta^*\_A,_i)^2$,Where L_B is the loss on the new task, θ*_A are the optimal parameters for the old task, and F is the Fisher information matrix.

1.5 General Framework

Machine learning analyzes large datasets to find patterns, modifying algorithms to improve predictions [2]. The process includes data collection, preprocessing, feature selection, and model training allowing algorithms to gain insights and modify responses autonomously [12]. The essential phases of a machine learning process are covered in this thorough procedure, from data preparation to model deployment and maintenance and the subsequent diagram is represented in Fig. 2. Essential steps of Machine Learning Model:

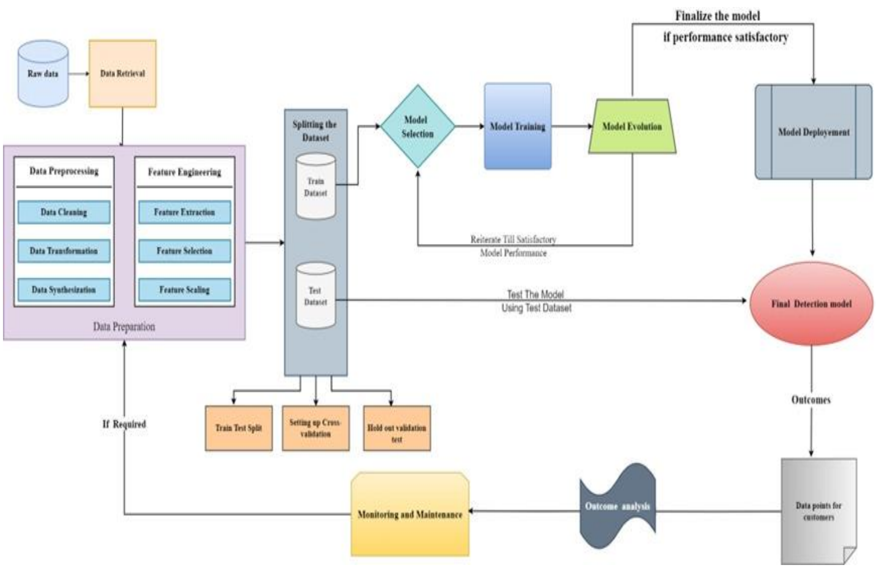

Fig. 2. Machine Learning Related Terms

1. Data Retrieval and Preprocessing:

a. Data Collection:

- Locate pertinent data sources, such as sensors, databases, and APIs.
- Compile a large dataset with examples of target variables or class labels that are known.
- Verify that the dataset accurately reflects the issue domain.

b. Data Cleaning:

- Use imputation or deletion to deal with missing values.
- Eliminate or adjust anomalies and outliers.
- Deal with inconsistent formatting of the data

c. Data Transformation:

- To guarantee uniform scale, normalize or standardize numerical characteristics.
- Use one-hot encoding or label encoding to encode data that is categorical.
- Manage data imbalance by utilizing approaches such as SMOTE (Synthetic Minority Over-sampling Technique).

d. Data Synthesization:

- Create fake data to balance classes or expand the size of the dataset.
- Use domain-specific data transformations already in place.

2. Feature Engineering and Selection:

a. Feature Extraction:

- From raw data, extract or build pertinent features.
- Use domain expertise to determine significant qualities.
- Reduce dimensionality by using methods like Principal Component Analysis (PCA).

b. Features Selection:

- Analyze the significance of each attribute using model-based techniques or statistical testing.
- To lower noise and enhance model performance, eliminate elements that are unnecessary or redundant.
- Use methods such as mutual information, recursive feature elimination, or correlation analysis.

c. Feature Scaling:

- For a fair comparison and stable models, scale characteristics to a common range.
- Data standardization methods like Z-score and Min-Max are commonly used preprocessing approaches for numerical features in machine learning pipelines.

3. Splitting the Dataset:

a. Train-Test split:

- Split the processed dataset into distinct set for model training and evaluation purposes.
- Split ratios of 70-30 or 80-20 (train-test) are typical

b. Setting Up Cross-Validation:

- For a more reliable model evaluation, use k-fold cross-validation.
- To ensure class distribution across folds, take stratified sampling into consideration.

c. Hold-out Validation Set:

- Make an additional validation set if you want to tweak the hyperparameters.

4. Model Selection and Training:

a. Selection of Algorithms:

- Select the right algorithms (clustering, regression, classification, etc.) for the type of problem.
- Think about things like feature dimensionality, dataset size, and interpretability needs.

b. Model Training:

- Supply the selected algorithm(s) with the prepared learning dataset.
- The system learns to correlate input characteristics with corresponding classifications or outcomes.
- In order to reduce a loss function, algorithms modify internal parameters.

c. Hyperparameter Tuning:

- Make use of methods like Bayesian optimization, random search, and grid search.
- To maximize performance, modify model-specific parameters.

5. Model Evaluation:

a. Performance Metrics:

- Classification tasks employ metrics like F1-score, precision, recall, accuracy, and ROC-AUC to evaluate model performance.
- Regression analysis employs key performance indicators including MAE, R-squared, and MSE to evaluate model accuracy and fit.k bacchA
- Choose metrics according to the needs of the business and the particular issue at hand.

b. Cross-Validation Results:

- Evaluate model stability by analyzing performance at various folds.

- Determine the measurements' standard deviation and mean.

c. Error Analysis:

- To evaluate the stability of the model, analyze performance at various folds.
- The metrics' mean and standard deviation should be calculated.

6. Final Model Selection and Deployment:

a. Model Comparison:

- Evaluate the performance of several models and choose the top performer.
- Think about the trade-offs between interpretability, complexity, and performance.

b. Model Deployment:

- To deploy the trained model, serialize it.
- Set up data pipelines for batch or real-time prediction processing.

c. Monitoring and Maintenance:

- In production, use logging and monitoring to track the performance of the model.
- Create alerts for shifts in the distribution of data or concepts.
- Schedule frequent updates and retraining of the model.

2 Mathematical Model of ML

The discipline of machine learning incorporates supervised, unsupervised, and semi-supervised strategies. The breakdown of learning algorithms into categories is illustrated in Fig. 3. Labeled data drives predictions in supervised learning [7], while unsupervised learning uncovers patterns in unlabeled data, and semi-supervised learning blends these approaches. Each method suits different problems, with model selection depending on data type, task requirements, interpretability, and computational efficiency [10].

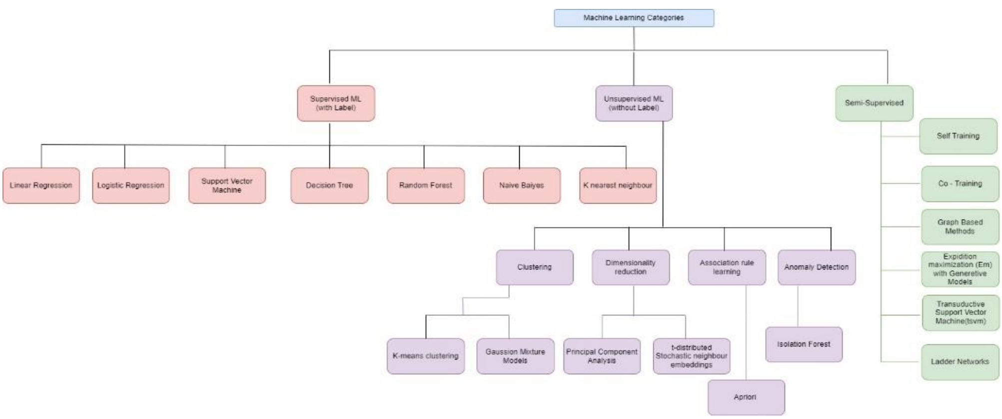

Fig. 3. Different Machine Learning Models

2.1 Supervised Mathematical Model

1. Linear Regression:

Model: $y = X\beta + \varepsilon$, where y: output vector, X: input matrix, β: coefficient vector, ε: error term. This method finds the best-fitting line by minimizing the sum of squared residuals.

2. Logistic Regression:

Model: $P(y=1|x) = 1 / (1 + e^{(-x^T\beta)})$. This model estimates the probability of binary outcomes using the sigmoid function.

3. Support Vector Machines (SVM):

Model: $f(x) = \text{sign}(w^Tx + b)$. SVM finds the hyperplane that best separates classes with maximum margin.

4. Decision Trees:

Model: Tree structure with features as nodes, decision rules as branches, outcomes as leaves. This method creates a tree-like model of decisions based on feature values [8].

5. Random Forests:

Model: Ensemble of decision trees, Random forests improve upon decision trees by reducing overfitting through ensemble learning.

6. Naive Bayes:

Model: The naive Bayes model, $P(y|x) \propto P(y)\ \Pi_i\ P(x_i|y)$, employs Bayesian inference while assuming features are conditionally independent given the class label.

7. K-Nearest Neighbors (KNN):

Model: No explicit training; prediction based on nearest neighbors in training set. KNN classifies or predicts based on nearest data points, using no fixed parameters. [9].

2.2 Unsupervised Mathematical Model:

1. Clustering Algorithms:

K-Means Clustering:

Objective: Minimize $J = \Sigma_{i=1}^k\ \Sigma_x \in S_i\ ||x - \mu_i||^2$, here k is cluster count, S_i is i-th cluster, μ_i is centroid, x is a point [13[.

Gaussian Mixture Models (GMM):

Likelihood: $L(\theta|X) = \Pi^n_{i=1}\ \Sigma^k_{j=1}\ \pi_j\ N(x_i|\mu_j, \Sigma_j)$. Uses EM algorithm to estimate parameters.

2. Dimensionality Reduction:

Principal Component Analysis (PCA):

Determines the covariance matrix $C = (1/n)XX^T$'s eigenvectors. Transforms data: $Y = W^TX$, where W contains top eigenvectors.

t-SNE:

Minimizes KL divergence: $KL(P||Q) = \Sigma_i \Sigma_j p_{ij} \log(p_{ij}/q_{ij})$. Where P, Q are high and low-dimensional similarity distributions.

3. Association Rule Learning:

a. Apriori Algorithm:

Support: $supp(I) = |\{t \in T \mid I \subseteq t\}| / |T|$ and Confidence: $conf(I \rightarrow J) = supp(I \cup J) / supp(I)$. Finds frequent itemset and generates rules.

4. Anomaly Detection:

a. Isolation Forest:

Anomaly score: $s(x, n) = 2\text{^}(-E(h(x))/c(n))$, where $E(h(x))$ represents the mean path length, and $c(n)$ denotes a normalizing factor [14].

2.3 Semi-Supervised Mathematical Models:

1. Self-Training:

Train: $f = \text{argmin}[\theta]\ \Sigma(x,y)\in L\ \text{loss}(f(x;\theta), y)$

Update: $L = L \cup \{(x, \hat{y}) \mid x \in U, \text{confidence}(\hat{y}) > \text{threshold}\}$

2. Co-Training:

Update: $L1 = L1 \cup \{(x, f1(x)) \mid x \in U, \text{confidence}(f1(x)) > \text{threshold1}\}$

$L2 = L2 \cup \{(x, f2(x)) \mid x \in U, \text{confidence}(f2(x)) > \text{threshold2}\}$

3. EM with Generative Models:

E-step: $\gamma(z_{ik}) = \pi_k N(x_i|\mu_k, \Sigma_k) / \Sigma_{j=1}^k \pi_j N(x_i|\mu_j, \Sigma_j)$

M-step: Update μ_k, Σ_k, π_k using L and U

4. Graph-Based Methods [15]:

Optimize: $\min[f]\ (1/2)\ \Sigma_{ij} w_{ij}(f(i) - f(j))^2 + \mu\ \Sigma_i\in L\ (f(i) - y_i)^2$

Solution: $f = (L + \mu I)^{-1}\mu y$

5. Transductive SVM:

$\min[w,b,y^*]\ (1/2)||w||^2 + C\ \Sigma_i\in L\ \xi_i + C^*\ \Sigma_i\in U\ \xi^*_i$

Subject to constraints for L and U

6. Ladder Networks:

Cost: $C = C_s + \lambda\ \Sigma_l\ C_u(l)$

Unsupervised cost: $C_u(l) = ||z(l) - \hat{z}(l)||^2$

3 Future Research Paths

The main goals of future machine learning research are to improve efficiency, robustness, and interpretability. Energy-efficient algorithms, methods for protecting privacy such as federated learning, AutoML [16], causal AI, few-shot learning, and AI for scientific discovery are important fields. Multimodal learning, human-AI cooperation, quantum machine learning, and ethical issues are also crucial. These initiatives address issues of justice, privacy, and environmental impact in the process of developing more robust, transparent, and adaptive AI systems [11].

4 Conclusion

This paper extensively explores machine learning's impact on data processing and decision-making across various fields, highlighting its mathematical foundations in probability theory, linear algebra, calculus, and optimization. It addresses ongoing challenges and research in data quality, interpretability, and ethics. The author emphasizes promising future developments, including quantum machine learning, causal AI, and privacy-preserving methods. These advancements are expected to yield more robust, transparent, and adaptable AI systems, overcoming current limitations and opening new possibilities across industries, potentially transforming scientific discovery and human-AI interaction.

Acknowledgments. All authors contributed equally to this research, sharing responsibility for its conception, execution, and documentation.

Disclosure of Interests. The researchers affirm absence of competing interests regarding their study, writing, and dissemination of this article, upholding scholarly integrity and openness in their academic endeavor.

References

1. Liu, L., Silva, E. A., Wu, C., & Wang, H. (2017). A machine learning-based method for the large-scale evaluation of the qualities of the urban environment. Computers, envi-ronment and urban systems, 65, 113-125, https://doi.org/10.1016/j.compenvurbsys.2017.06.003.
2. Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learn-ing. Electronic Markets, 31(3), 685-695, https://doi.org/10.1007/s12525-021-00475-2.

Application

3. Mishra, S., & Tyagi, A. K. (2022). The role of machine learning techniques in internet of things-based cloud applications. Artificial intelligence-based internet of things sys-tems, 105-135,
4. Mair, C., Kadoda, G., Lefley, M., Phalp, K., Schofield, C., Shepperd, M., & Webster, S. (2000). An investigation of machine learning based prediction systems. Journal of sys-tems and software, 53(1), 23-29, https://doi.org/10.1016/S0164-1212(00)00005-4.
5. Mahesh, B. (2020). Machine learning algorithms-a review. International Journal of Sci-ence and Research (IJSR).[Internet], 9(1), 381-386, DOI: 10.21275/ART20203995.
6. Singh, A., Thakur, N., & Sharma, A. (2016, March). A review of supervised machine learning algorithms. In 2016 3rd international conference on computing for sustainable global development (INDIACom) (pp. 1310-1315). Ieee,
7. Dhall, D., Kaur, R., & Juneja, M. (2020). Machine learning: a review of the algorithms and its applications. Proceedings of ICRIC 2019: Recent innovations in computing, 47-63.
8. Dhall, D., Kaur, R., & Juneja, M. (2020). Machine learning: a review of the algorithms and its applications. Proceedings of ICRIC 2019: Recent innovations in computing, 47-63.
9. Ray, S. (2019, February). A quick review of machine learning algorithms. In 2019 In-ternational

conference on machine learning, big data, cloud and parallel computing (COMITCon) (pp. 35-39). IEEE,https://doi.org/10.1109/COMITCon.2019.8862451.

10. Shrestha, A., & Mahmood, A. (2019). Review of deep learning algorithms and architec-tures. IEEE access, 7, 53040-53065.,https://doi.org/10.1109/ACCESS.2019.2912200.
11. Choudhary, R., & Gianey, H. K. (2017, December). Comprehensive review on super-vised machine learning algorithms. In 2017 International Conference on Machine Learning and Data Science (MLDS) (pp. 37-43). IEEE, https://doi.org/10.1109/MLDS.2017.11.
12. Crisci, C., Ghattas, B., & Perera, G. (2012). A review of supervised machine learning algorithms and their applications to ecological data. Ecological Modelling, 240, 113-122, https://doi.org/10.1016/j.ecolmodel.2012.03.001.
13. Naeem, S., Ali, A., Anam, S., & Ahmed, M. M. (2023). An unsupervised machine learning algorithms: Comprehensive review. International Journal of Computing and Digital Systems.,http://dx.doi.org/10.12785/ijcds/130172.
14. Alloghani, M., Al-Jumeily, D., Mustafina, J., Hussain, A., & Aljaaf, A. J. (2020). A systematic review on supervised and unsupervised machine learning algorithms for data science. Supervised and unsupervised learning for data science, 3-21.
15. Khanum, M., Mahboob, T., Imtiaz, W., Ghafoor, H. A., & Sehar, R. (2015). A survey on unsupervised machine learning algorithms for automation, classification and maintenance. International Journal of Computer Applications, 119(13).
16. Usama, M., Qadir, J., Raza, A., Arif, H., Yau, K. L. A., Elkhatib, Y., ... & Al-Fuqaha, A. (2019). Unsupervised machine learning for networking: Techniques, applications and research challenges. IEEE access, 7, 65579-65615, https://doi.org/10.1109/ACCESS.2019.2916648.
17. Jaffee, L. R. (1984). Of Probativity and Probability: Statistics, Scientific Evidence, and the Calculus of Chance at Trial. U. Pitt. L. Rev., 46, 925, https://doi.org/10.2307/2333573.
18. Vovk, V. G. (1993). A logic of probability, with application to the foundations of sta-tistics. Journal of the Royal Statistical Society Series B: Statistical Methodology, 55(2), 317-34, https://doi.org/10.1111/j.2517-6161.1993.tb01904.x.
19. Lefebvre, M. (2007). Applied probability and statistics. Springer Science & Business Media,
20. Di Nardo, E. (2015). Symbolic calculus in mathematical statistics: a review. arXiv pre-print arXiv:1512.08379, https://doi.org/10.48550/arXiv.1512.08379.