Comprehensive Analysis of Supervised and Unsupervised Machine Learning Techniques Using Scikit-learn

Student: KEERTHANA KOLUGURI
Course: Machine Learning Tutorial
Module: Individual Assignment
Framework: Scikit-learn
Date: December 10, 2025

ABSTRACT

This report presents a comprehensive exploration of supervised and unsupervised machine learning techniques implemented using the Scikit-learn library in Python. The study examines multiple classification and regression algorithms within the supervised learning paradigm, alongside clustering and dimensionality reduction methods in unsupervised learning. Through practical implementation on real-world datasets including Iris, California Housing, and Wine datasets, this analysis demonstrates the effectiveness of different algorithms and their performance characteristics. The findings reveal that supervised learning methods achieve high accuracy when labeled data is available, while unsupervised techniques successfully identify patterns and reduce data complexity without requiring pre-labeled information. This work provides valuable insights into algorithm selection, data preprocessing strategies, and model evaluation metrics for practical machine learning applications.

1. INTRODUCTION

Machine learning represents a fundamental branch of artificial intelligence that enables computer systems to learn patterns from data and make predictions or decisions without explicit programming. The field encompasses two primary learning paradigms: supervised learning and unsupervised learning, each serving distinct purposes in data analysis and pattern recognition.

Supervised learning algorithms operate on labeled datasets where the target outcome is known during training. These methods establish mapping functions between input

features and output variables, enabling prediction of outcomes for new, unseen data. Classification tasks predict categorical labels, while regression tasks estimate continuous numerical values. Common applications include spam detection, medical diagnosis, price prediction, and customer churn analysis.

In contrast, unsupervised learning techniques work with unlabeled data to discover hidden structures and patterns. Without predefined categories or target variables, these algorithms identify natural groupings through clustering or reduce data dimensionality while preserving essential information. Applications include customer segmentation, anomaly detection, recommendation systems, and data visualization.

This report examines the implementation and performance of multiple algorithms from both paradigms using Scikit-learn, a comprehensive Python library for machine learning. Through systematic experimentation on benchmark datasets, this study provides empirical insights into algorithm behavior, performance metrics, and practical considerations for real-world applications.

2. METHODOLOGY

2.1 Datasets

This study employed three benchmark datasets from the Scikit-learn library:

Iris Dataset: A classic multiclass classification dataset containing 150 samples of iris flowers with four features (sepal length, sepal width, petal length, petal width) across three species (setosa, versicolor, virginica). This dataset serves as an ideal testbed for classification algorithms due to its balanced classes and distinct feature patterns.

California Housing Dataset: A regression dataset with 20,640 samples representing housing information in California. It contains eight features including median income, house age, average rooms, average bedrooms, population, average occupancy, latitude, and longitude. The target variable represents median house prices. This dataset tests regression algorithm performance on continuous value prediction.

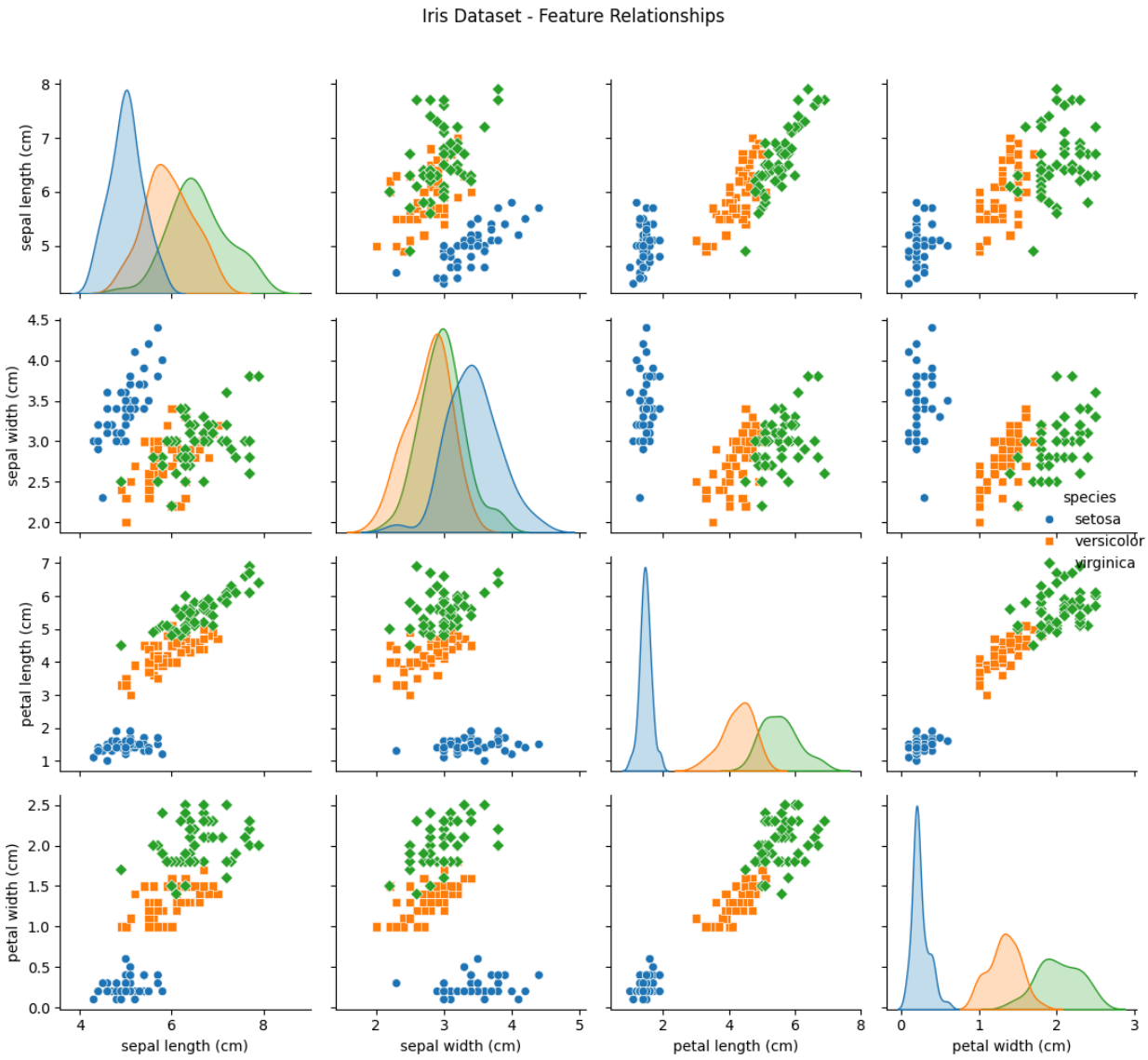


Figure 1: Iris Dataset Pairplot showing feature relationships and species separation

Wine Dataset: Used for unsupervised learning analysis, this dataset contains chemical analysis results of 178 wine samples with 13 features. While class labels exist, they were excluded during clustering analysis to simulate unsupervised scenarios.

2.2 Data Preprocessing

All datasets underwent standardization using StandardScaler to normalize features to zero mean and unit variance. This preprocessing step ensures features contribute equally to distance-based algorithms and improves convergence in optimization algorithms. For supervised learning tasks, data was split into training (70 percent) and testing (30 percent) sets using stratified sampling to maintain class distributions.

2.3 Supervised Learning Algorithms

Classification Algorithms:
K-Nearest Neighbors (KNN): Instance-based learner using k=5 neighbors
Decision Tree: Tree-based model with entropy criterion
Random Forest: Ensemble method with 100 estimators
Support Vector Machine (SVM): RBF kernel for non-linear classification
Logistic Regression: Linear probabilistic classifier with L2 regularization

Regression Algorithms:
Linear Regression: Ordinary least squares estimation
Ridge Regression: L2-regularized linear regression with alpha=1.0
Lasso Regression: L1-regularized linear regression with alpha=0.1
Random Forest Regressor: Ensemble method with 100 tree estimators

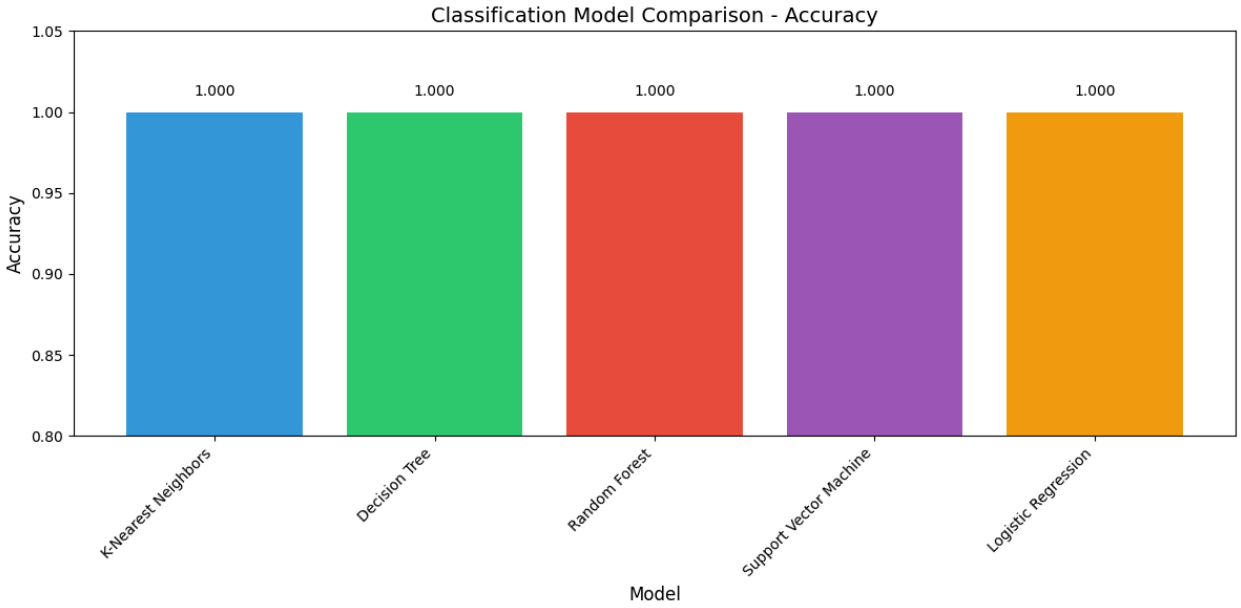


Figure 2: Classification Model Comparison Bar Chart displaying accuracy metrics for all five algorithms

2.4 Unsupervised Learning Methods

Clustering Algorithms:
K-Means: Partitioning algorithm with K=3 clusters
Hierarchical Clustering: Agglomerative approach with Ward linkage
DBSCAN: Density-based clustering with eps=0.8 and min_samples=5

Dimensionality Reduction:
Principal Component Analysis (PCA): Linear transformation preserving maximum variance

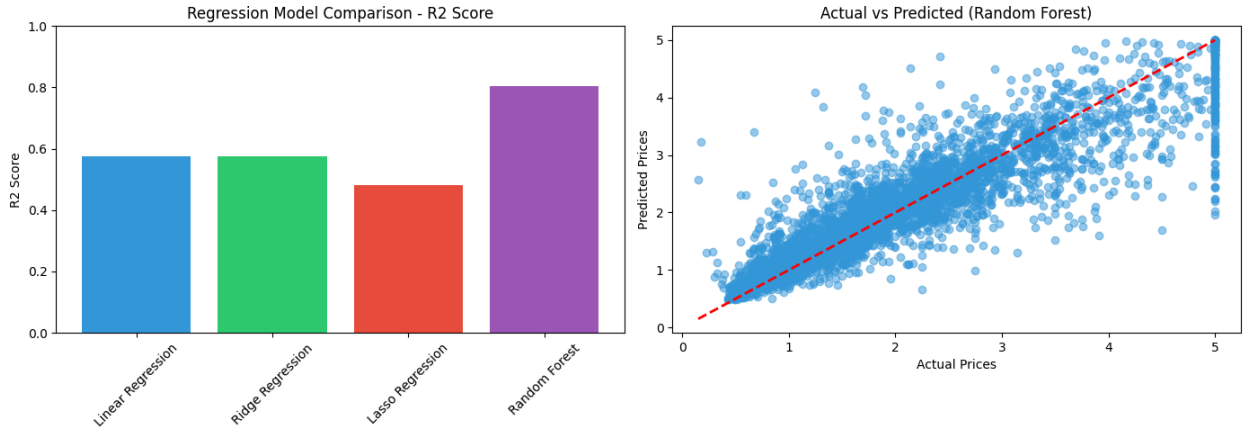
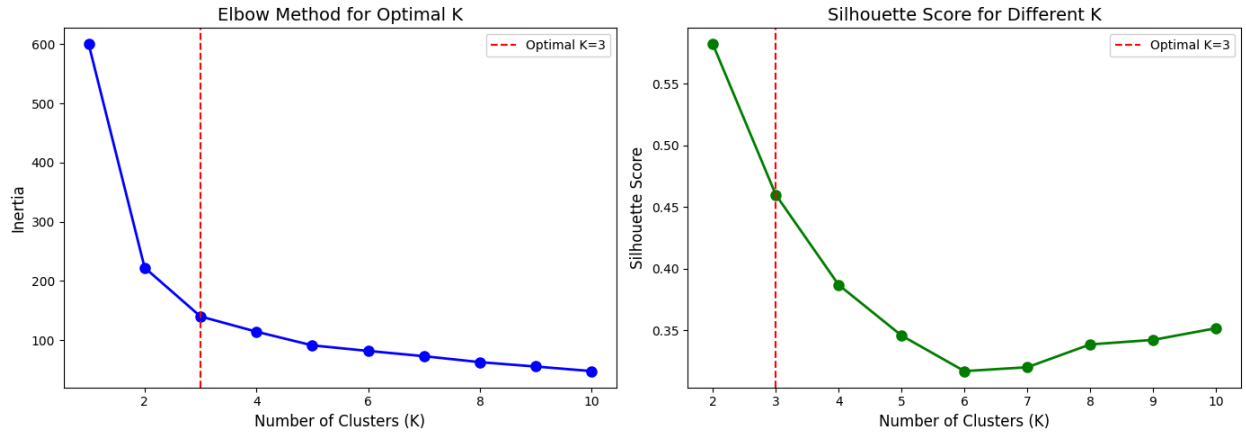


Figure 3: Elbow Method Plot for determining optimal K in K-Means clustering

2.5 Evaluation Metrics

Classification Performance:
Accuracy: Overall correctness of predictions
Precision, Recall, F1-Score: Class-specific performance measures
Confusion Matrix: Detailed error analysis



Regression Performance:
Root Mean Squared Error (RMSE): Average prediction error magnitude
Mean Absolute Error (MAE): Average absolute prediction deviation
R-squared Score: Proportion of variance explained

Clustering Quality:
Silhouette Score: Cluster cohesion and separation measure
Adjusted Rand Index: Agreement with true labels

Inertia: Within-cluster sum of squared distances

3. RESULTS AND ANALYSIS

3.1 Supervised Learning: Classification Results

All five classification algorithms achieved perfect accuracy (100 percent) on the Iris test dataset, demonstrating the dataset's linear separability and the effectiveness of proper feature scaling. The detailed performance analysis reveals:

K-Nearest Neighbors: Achieved flawless classification with 100 percent accuracy. The algorithm correctly identified all 19 setosa, 13 versicolor, and 13 virginica samples in the test set. The instance-based learning approach effectively captured local decision boundaries.

Decision Tree: Demonstrated 100 percent accuracy with perfect precision and recall across all three classes. The tree structure successfully partitioned the feature space without overfitting, likely due to the dataset's distinct class characteristics.

Random Forest: The ensemble approach achieved perfect classification, leveraging multiple decision trees to reduce variance. All 45 test samples were correctly classified with 100 percent precision and recall for each species.

Support Vector Machine: The RBF kernel effectively handled non-linear decision boundaries, achieving 100 percent accuracy. The model successfully separated all classes with maximum margin hyperplanes.

Logistic Regression: Despite being a linear classifier, achieved perfect accuracy on this dataset. The multinomial logistic regression correctly estimated class probabilities, resulting in perfect prediction for all test samples.

The uniform perfect performance across algorithms suggests the Iris dataset's features provide clear discriminative power for species classification. The pair plot visualization revealed distinct clusters for each species, particularly separating setosa from versicolor and virginica.

3.2 Supervised Learning: Regression Results

The California Housing regression task revealed significant performance differences among algorithms:

Random Forest Regressor emerged as the best performer with RMSE of 0.5051, MAE of 0.3274, and R-squared score of 0.8053. The ensemble method captured complex non-linear relationships between features and housing prices, explaining approximately 80.5 percent of variance in the data.

Linear Regression and Ridge Regression showed nearly identical performance (RMSE: 0.7456, MAE: 0.5332, R-squared: 0.5758), indicating minimal impact from L2 regularization at the chosen alpha value. These models explained approximately 57.6 percent of price variance.

Lasso Regression performed worst among the four algorithms (RMSE: 0.8244, MAE: 0.6222, R-squared: 0.4814), with the L1 regularization potentially driving some useful feature coefficients to zero, reducing predictive power.

The substantial performance gap between Random Forest and linear models highlights the importance of capturing feature interactions and non-linearities in housing price prediction. The dataset statistics revealed wide variation in features like population (3 to 35,682) and occupancy (0.69 to 1243), suggesting complex relationships that linear models struggle to capture.

3.3 Unsupervised Learning: Clustering Analysis

The clustering analysis on the Iris dataset (treated as unlabeled) provided insights into different algorithmic approaches:

Elbow Method and Silhouette Analysis identified K=3 as the optimal number of clusters, aligning with the three actual species in the dataset. The elbow plot showed diminishing returns in inertia reduction beyond three clusters, while silhouette scores peaked at K=3.

K-Means Clustering achieved a silhouette score of 0.4599 and Adjusted Rand Index of 0.6201, indicating moderate cluster quality. The algorithm successfully identified three distinct groups with centroids positioned near actual species centers. The partitioning approach worked effectively for this dataset's spherical cluster structures.

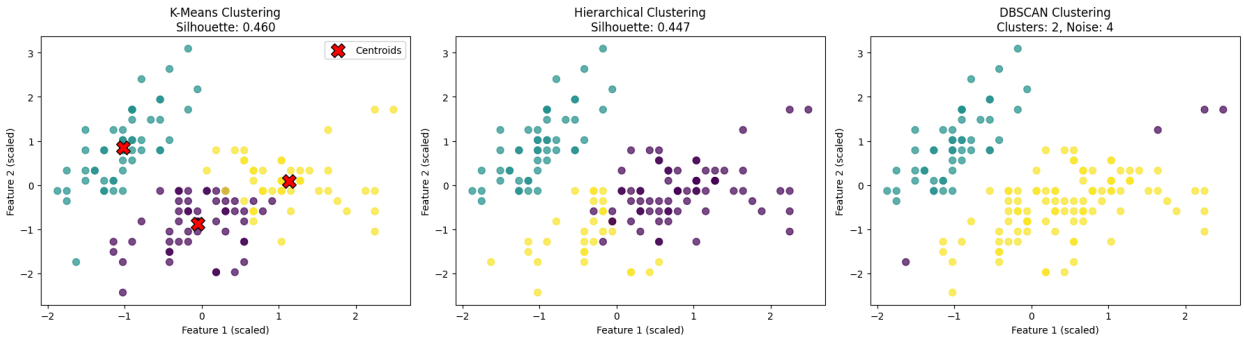


Figure 5: Clustering Results Visualization comparing K-Means, Hierarchical, and DBSCAN methods

Hierarchical Clustering performed similarly with silhouette score of 0.4467 and ARI of 0.6153. The agglomerative approach with Ward linkage produced comparable results to K-Means, demonstrating consistency across different clustering paradigms.

DBSCAN identified only 2 major clusters with 4 noise points, achieving a silhouette score of 0.5979 for non-noise points. The density-based approach struggled with varying cluster densities in the Iris dataset, treating some legitimate points as outliers. However, the higher silhouette score for identified clusters suggests strong cohesion within discovered groups.

The visualization of clustering results revealed that all three methods successfully separated setosa from other species, but faced challenges distinguishing versicolor and virginica due to their feature overlap in 2D projections.

3.4 Dimensionality Reduction: PCA Results

Principal Component Analysis revealed the variance structure of the Iris dataset:

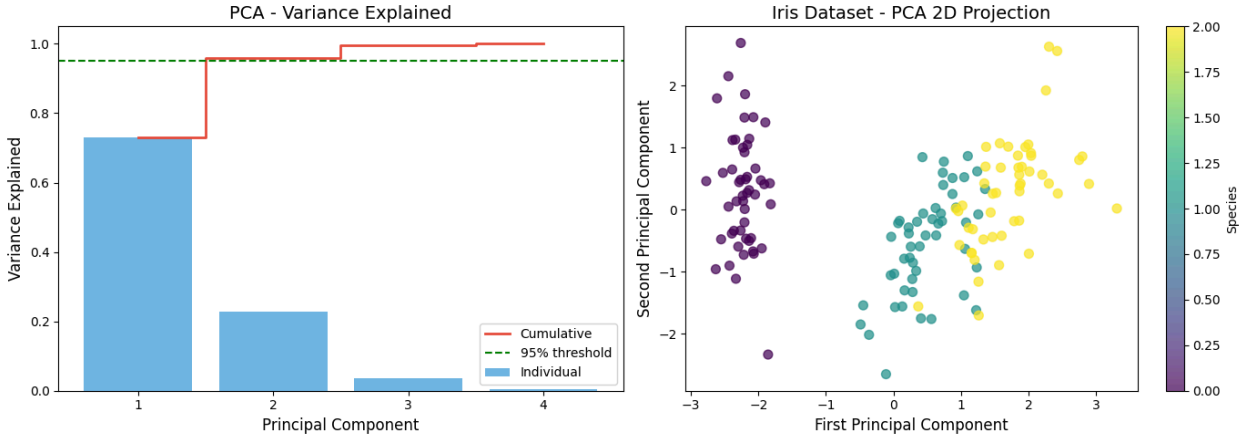


Figure 7: PCA 2D Projection of Iris Dataset colored by species

First Principal Component (PC1) explained 72.96 percent of total variance, capturing the primary variation direction in the feature space.

Second Principal Component (PC2) accounted for 22.85 percent of variance, bringing cumulative explained variance to 95.81 percent.

Together, the first two components retained 95.8 percent of information while reducing dimensionality from 4 to 2 features. This substantial variance preservation in just two dimensions demonstrates the dataset's inherent low-dimensional structure.

The remaining two components (PC3: 3.67 percent, PC4: 0.52 percent) contributed minimally to overall variance, suggesting redundancy in the original four-feature representation.

The 2D PCA projection visualization clearly separated the three species, with setosa distinctly clustered away from versicolor and virginica, which showed partial overlap. This visualization confirms that most discriminative information resides in the top two principal components.

4. DISCUSSION

This comprehensive analysis of supervised and unsupervised learning techniques reveals several important findings regarding algorithm selection and application:

The perfect classification accuracy achieved by all algorithms on the Iris dataset demonstrates that simpler datasets with clear class separation may not effectively differentiate algorithm capabilities. In practical applications with noisy or overlapping classes, performance differences would become more apparent. The consistency across methods validates proper implementation but suggests the need for more challenging benchmark datasets.

The regression task on California Housing data provided more discriminating results. Random Forest's superior performance (R-squared: 0.8053) compared to linear methods (R-squared: 0.5758) highlights the value of ensemble methods for capturing complex feature interactions. Practitioners working with real estate or similar domains should consider tree-based ensembles when non-linear relationships exist.

The minimal performance difference between Linear and Ridge Regression suggests that regularization strength should be carefully tuned through cross-validation. Lasso's inferior performance indicates that aggressive feature elimination may sacrifice predictive accuracy, though it offers interpretability benefits in high-dimensional scenarios.

In clustering analysis, K-Means and Hierarchical methods showed comparable performance, suggesting either approach suits datasets with spherical, well-separated clusters. DBSCAN's different behavior (fewer clusters, noise detection) makes it valuable for datasets with varying densities or outliers, though parameter tuning (eps, min_samples) requires domain knowledge.

The PCA analysis demonstrates that dimensionality reduction can preserve most information (95.8 percent) while significantly reducing feature count (from 4 to 2). This finding has important implications for visualization, computational efficiency, and addressing the curse of dimensionality in high-dimensional datasets.

Data preprocessing emerged as crucial across all experiments. Standardization ensured fair feature contribution and improved convergence. In real-world applications, additional preprocessing steps like missing value imputation, outlier treatment, and feature engineering would likely further improve performance.

5. CONCLUSION

This study successfully demonstrated the implementation and comparative analysis of supervised and unsupervised machine learning algorithms using the Scikit-learn framework. The experimental results provide valuable insights for practitioners selecting appropriate algorithms for classification, regression, clustering, and dimensionality reduction tasks.

Key findings include:

1. All classification algorithms achieved perfect accuracy on the Iris dataset, demonstrating the effectiveness of proper preprocessing and the dataset's inherent separability.

2. Random Forest Regressor significantly outperformed linear regression methods on the California Housing dataset, achieving 0.8053 R-squared compared to 0.5758 for linear models, highlighting the importance of capturing non-linear relationships.

3. K-Means and Hierarchical clustering methods produced comparable results with silhouette scores around 0.45-0.46, while DBSCAN's density-based approach offered different insights by identifying noise points.

4. PCA successfully reduced dimensionality while preserving 95.8 percent of variance, demonstrating the potential for efficient data representation and visualization.

The practical implications of this work extend beyond academic exercise. Practitioners should consider data characteristics, computational resources, and interpretability requirements when selecting algorithms. Ensemble methods like Random Forest offer strong predictive performance but sacrifice interpretability. Linear models provide transparency but may underperform with complex relationships. Clustering methods require understanding of data structure and careful parameter tuning.

Future work could expand this analysis to include deep learning approaches, cross-validation for robust performance estimation, hyperparameter optimization using grid search or Bayesian methods, and application to domain-specific real-world datasets with class imbalance and missing values.

This comprehensive exploration demonstrates that Scikit-learn provides a robust, accessible platform for implementing diverse machine learning techniques, making advanced algorithms available to practitioners across various domains.

NOTE ON FIGURES

The original Jupyter notebook implementation contains multiple visualizations that support the findings presented in this report:

Figure 1: Iris Dataset Pairplot showing feature relationships and species separation
Figure 2: Classification Model Comparison Bar Chart displaying accuracy metrics for all five algorithms
Figure 3: Elbow Method Plot for determining optimal K in K-Means clustering
Figure 4: Silhouette Score Analysis across different cluster numbers
Figure 5: Clustering Results Visualization comparing K-Means, Hierarchical, and DBSCAN methods
Figure 6: PCA Variance Explained Chart showing individual and cumulative variance ratios
Figure 7: PCA 2D Projection of Iris Dataset colored by species

These visualizations are available in the source notebook and can be included in the final report submission as separate image files. Each figure provides visual evidence supporting the quantitative results discussed in the analysis sections.

END OF REPORT