International Journal of Operations Research and Artificial Intelligence

www.ijorai.reapress.com

Int. J. Oper. Res. Artif. Intell. Vol. 1, No. 2 (2025) 90–100.

Paper Type: Original Article

A Review of the Structure and Application of Scikit-Learn Datasets in Machine Learning Model Development

Mohammad Mahdi Salehi[1,*], Hossein Zarei[1]

[1] Department of Mathematics and Computer Science, Shiraz Branch, Islamic Azad University, Shiraz, Iran; mahdy.salehy1343@gmail.com; hssinofficial@gmail.com.

Citation:

Received: 25 July 2024 *Revised: 17 September 2024* *Accepted: 18 November 2024*	Salehi, M. M., & Zarei, H. (2025). A review of the structure and application of scikit-learn datasets in machine learning model development. *International journal of operations research and artificial intelligence, 1*(2), 90-100.

Abstract

Scikit-learn is a widely used Python library that provides a diverse range of datasets alongside robust machine learning algorithms. This paper presents a comprehensive review of the structure and practical applications of key datasets available in Scikit-learn, emphasizing their role in developing and evaluating machine learning models. The datasets are categorized into built-in, fetchable, and synthetic types, each suited for different research and educational purposes. Through Exploratory Data Analysis (EDA), visualization, and baseline modeling on representative datasets such as Iris, Breast Cancer Wisconsin, and California Housing, this study highlights how these datasets facilitate various machine learning tasks, including classification and regression. Insights into dataset characteristics like class distribution, feature separability, and target variability guide the selection and optimization of algorithms. Overall, this review underscores the value of Scikit-learn datasets as foundational resources for prototyping and education, while also advocating for integration with larger, real-world datasets to address complex industrial challenges.

Keywords: Scikit-learn, Machine learning datasets, Data preprocessing, Model evaluation.

1 | Introduction

Machine learning has rapidly advanced as a cornerstone technology across various fields, including healthcare, finance, and Natural Language Processing (NLP). Central to this advancement is the availability of diverse and well-structured datasets, which are essential for training, validating, and benchmarking machine learning models. The Scikit-learn library, known for its simplicity and comprehensive functionalities, not only offers a wide range of algorithms but also provides numerous datasets that facilitate educational, research, and

Corresponding Author: mahdy.salehy1343@gmail.com

https://doi.org/10.48314/ijorai.v1i2.64

experimental purposes. Familiarity with these datasets, their structure, and appropriate applications is crucial for effective model development and reproducibility.

Datasets in Scikit-learn are generally categorized into three main types: Built-in datasets, fetchable datasets, and synthetic datasets. Built-in datasets are typically small and curated, serving as excellent resources for learning and prototyping. Fetchable datasets, which can be downloaded from external repositories, are larger and more representative of real-world scenarios, thereby suitable for applied machine learning research. Synthetic datasets allow users to generate controlled data tailored to specific experimental needs, enabling systematic testing of algorithm performance under varying conditions. This classification ensures that Scikit-learn supports a wide array of learning objectives and facilitates the design of versatile machine learning workflows.

Prior to model training, Exploratory Data Analysis (EDA) and visualization play an indispensable role in revealing the intrinsic properties of datasets. Visualization techniques such as scatter plots, histograms, and bar charts help illustrate feature distributions, class separability, and data imbalances. These insights guide data preprocessing and model selection, improving both the interpretability and effectiveness of learning algorithms. Scikit-learn's datasets often include detailed metadata and can be easily converted to common data structures for seamless analysis and visualization.

A structured approach to preprocessing and modeling enhances reproducibility and efficiency in machine learning projects. Scikit-learn offers pipeline mechanisms that integrate preprocessing steps such as scaling and encoding with model training and evaluation, enabling consistent and streamlined workflows. Additionally, compatibility with libraries like Pandas and NumPy facilitates flexible data manipulation and exploratory analysis, accelerating both educational and research activities in the machine learning domain.

In summary, understanding the types and characteristics of datasets provided by Scikit-learn empowers users to select appropriate data sources for diverse machine learning tasks. While these datasets form an excellent foundation for prototyping and teaching, augmenting them with larger and more complex external datasets is often necessary for addressing real-world challenges. This paper aims to provide a detailed review of Scikit-learn datasets, highlighting their structure, practical applications, and integration into machine learning workflows to aid researchers and practitioners in their model development endeavors.

This article reviews the datasets provided by the Scikit-learn library and their applications in machine learning model development. In the Background section, the importance of diverse data types and their role in machine learning workflows is discussed, emphasizing numerical, categorical, textual, and image data. The Methodology section outlines the classification of datasets into built-in, fetchable, and synthetic categories. It describes the process of loading, preprocessing, EDA, and baseline model training on selected examples such as the Iris, California Housing, and synthetic datasets. The article then presents a detailed examination of key dataset categories, including classical datasets like Iris and Breast Cancer Wisconsin, larger fetchable datasets like California Housing and 20 Newsgroups, and synthetic datasets generated for controlled experiments. Visualization techniques for understanding data distributions and feature relations are also presented. Finally, the Conclusion highlights how these datasets support educational and research activities, their strengths and limitations, and recommends complementing them with real-world large-scale data for robust machine learning applications.

2 | Background

Scikit-learn has become one of the most popular libraries for implementing machine learning algorithms in Python. In addition to offering a broad range of modeling techniques, it provides easy access to various datasets for education, benchmarking, and experimentation. Understanding the structure, types, and usage of these datasets is essential for effectively utilizing the library and designing reproducible experiments. This section provides a comprehensive overview of dataset types, data structures, and their role in the machine learning pipeline, with a focus on Scikit-learn.

Data plays a fundamental and irreplaceable role in the development, evaluation, and implementation of intelligent models. Without access to high-quality and well-structured data, no algorithm can effectively learn patterns or generate meaningful predictions. Data serves as the "fuel" of learning systems, enabling them to discover behaviors, relationships, and hidden trends within diverse problems. Therefore, access to suitable and varied datasets is one of the most critical prerequisites for both research-oriented and industrial machine learning projects.

The data used in machine learning can be categorized into several main types based on their structure and application. The first type is numerical or quantitative data, in which features are represented by numeric values (Such as height, weight, or price). The second is categorical data, where the variables consist of discrete classes or labels. In addition, other types include textual data, image data, time series data, and audio data. Each data type requires specific preprocessing techniques, feature extraction methods, and modeling strategies—posing new challenges in algorithm development and implementation.

In the context of machine learning, a dataset refers to an organized and structured collection of data made available for model development and evaluation. These datasets typically contain labeled data (For supervised learning) or unlabeled data (For unsupervised learning) and often include metadata such as feature names, variable types, and additional descriptions. Standardized datasets allow researchers to compare model performance across different methods and assess algorithms under consistent and reproducible conditions.

Scikit-learn is one of the most widely used libraries for machine learning in Python, offering extensive capabilities for classification, regression, clustering, dimensionality reduction, feature selection, and model evaluation. In addition to providing access to core machine learning algorithms, Scikit-learn includes a variety of built-in and fetchable datasets that are particularly useful for training, testing, and demonstrating models. One of the key advantages of this library lies in its simplicity and integration of tools that make it ideal for both academic research and practical education.

Datasets in Scikit-learn can generally be divided into three main categories: 1) built-in datasets that can be accessed directly through functions such as load_iris() or load_wine(), 2) fetchable or downloadable datasets accessible via internet through functions like fetch_california_housing() or fetch_20newsgroups(), and 3) synthetic or artificial datasets that can be generated using functions like make_classification() and make_regression(). This classification system addresses diverse educational and experimental needs, providing a versatile environment for model testing and evaluation.

Most Scikit-learn datasets are returned in the form of a Bunch object, which behaves similarly to a dictionary and includes components such as data (Features), target (Labels), feature_names, and DESCR (Description). Users can easily convert these datasets into Pandas DataFrames for convenient analysis and visualization. Furthermore, the as_frame=True parameter in recent versions of the library enables direct output of datasets as DataFrames, streamlining the preprocessing and exploration stages of a machine learning workflow.

Among the key advantages of Scikit-learn datasets are their ease of access, variety of types, and full compatibility with the library's algorithms. These datasets are especially suitable for learning, prototyping, and comparing models. However, some of the included datasets are relatively small and simplistic, which may limit their applicability in industrial or large-scale research contexts. In such cases, it is recommended to incorporate larger and more complex datasets from external sources such as Kaggle or OpenML.

Overall, the datasets provided by Scikit-learn constitute a valuable resource for learning and experimentation in machine learning. Their simplicity, diversity, and integration into a widely adopted library make them an excellent starting point for students, researchers, and practitioners. It is recommended that, in addition to utilizing these datasets, learners explore more realistic and large-scale data to gain deeper practical insights and prepare for real-world applications.

3 | Methodology

This study adopts a structured methodology to examine the datasets available in the Scikit-learn library and to demonstrate how these datasets can be effectively used in machine learning workflows. The methodology is based on both descriptive analysis and practical experimentation using Python programming. It encompasses the identification of dataset types, their structure, loading mechanisms, and integration into learning models. The goal is to establish a reproducible and generalizable framework for exploring data within Scikit-learn.

The first step involves classifying the datasets into three main groups: Built-in datasets, fetchable datasets, and synthetic datasets. This classification provides a clear understanding of their origin, size, format, and intended use. Built-in datasets are small and come packaged with the library, fetchable datasets require an internet connection to download from external sources, and synthetic datasets are generated programmatically for simulation and testing purposes.

To evaluate the datasets, we selected a representative sample from each category. Specifically, we used the Iris dataset from the built-in group, the California Housing dataset from the fetchable group, and a synthetic classification dataset created using make_classification(). These datasets were chosen based on their popularity, structure, and relevance to standard machine learning tasks such as classification and regression.

Each dataset was loaded using the corresponding dataset loader function provided by Scikit-learn. For example, the Iris dataset was loaded using load_iris(), which returns a Bunch object containing data, targets, feature names, and descriptions. Similarly, fetch_california_housing() was used to download and access the California housing data. For synthetic data, make_classification() was employed with predefined parameters for the number of features, classes, and samples.

After loading the datasets, the next step was to convert them into Pandas DataFrames for easier manipulation and visualization. This step involved extracting the data and target components from the Bunch object and combining them into a structured tabular format. Where applicable, the as_frame=True option was used to automatically generate the data as a DataFrame, preserving column names and data types.

Following data conversion, basic EDA was conducted on each dataset. This included examining the shape and structure of the data, identifying missing values or anomalies, summarizing statistics for numerical features, and visualizing class distributions or target variables. EDA provided critical insights into the nature of the datasets and informed the next steps in the methodology.

Data preprocessing was applied where necessary. For example, feature scaling was performed using StandardScaler for datasets with varying feature magnitudes. Categorical variables (If present) were encoded using one-hot encoding or label encoding. In some cases, train-test splitting was implemented using train_test_split from sklearn.model_selection to divide the datasets into training and testing subsets, facilitating fair evaluation of model performance.

To demonstrate the usability of the datasets, baseline machine learning models were trained and tested on each one. For classification tasks, algorithms such as K-Nearest Neighbors (KNN), Decision Tree Classifier, and Support Vector Machines (SVM) were employed. For regression tasks like the California Housing dataset, models including Linear Regression and Random Forest Regressor were applied. Model performance was evaluated using appropriate metrics such as accuracy, precision, and recall for classification, and MAE/R^2 for regression.

The implementation of models was standardized using Scikit-learn's Pipeline functionality to streamline the process of preprocessing, training, and evaluation. This approach ensured reproducibility and consistency across different datasets. Each pipeline included preprocessing steps such as scaling or encoding, followed by the chosen estimator. The use of pipelines also facilitated hyperparameter tuning and cross-validation if needed.

Cross-validation was incorporated to assess the stability and generalization of the models. Specifically, 5-fold cross-validation was applied to each classification and regression model using cross_val_score, providing an average performance metric across different data splits. This helped mitigate overfitting and allowed for a more reliable comparison of model capabilities across datasets.

Additionally, visualization tools such as matplotlib and seaborn were used to enhance the interpretability of results. For example, heatmaps of correlation matrices, scatter plots of feature relationships, and bar charts of class frequencies were generated. These visualizations provided more profound insights into dataset characteristics and supported model selection decisions.

To ensure reproducibility and transparency, all Python code used in the methodology was implemented in Jupyter Notebooks and thoroughly commented. The notebook environment enabled step-by-step execution and easy sharing of the analysis workflow. Key outputs, including model accuracy, confusion matrices, and regression plots, were recorded to support the findings and conclusions of this study.

In summary, the methodology presented in this study combines dataset exploration, preprocessing, modeling, and evaluation into a unified workflow. By using Scikit-learn's built-in functionalities, the process becomes efficient, replicable, and adaptable for a variety of datasets and machine learning tasks. The findings from this structured approach are discussed in the next section.

The Scikit-learn library offers a comprehensive set of datasets organized into three primary categories: built-in datasets, fetchable datasets, and synthetic datasets. These datasets serve as invaluable resources for researchers and practitioners to develop, benchmark, and validate machine learning algorithms across a wide range of applications.

Built-in datasets

Built-in datasets are pre-packaged datasets that come bundled with Scikit-learn. They are usually small to medium-sized and well-curated, making them ideal for educational purposes, algorithm prototyping, and reproducible research. Due to their convenience and simplicity, these datasets have become standard benchmarks in the machine learning community.

Iris dataset: Introduced by Fisher [1], the Iris dataset is one of the most classical datasets in pattern recognition. It consists of 150 samples, each with four numerical features—sepal length, sepal width, petal length, and petal width—belonging to three species of the Iris flower. The task is to classify the samples into these three species. This dataset is widely used for multiclass classification algorithm evaluation and visualization [2].

Wine dataset: This dataset contains 178 samples derived from chemical analysis of wines from three different cultivars in Italy. Each sample includes 13 continuous features describing various chemical properties such as alcohol content, malic acid, and phenols. The Wine dataset is utilized extensively for multiclass classification, feature selection, and dimensionality reduction studies [3].

Breast Cancer Wisconsin Dataset: A binary classification dataset comprising 569 samples with 30 numeric features extracted from digitized images of fine needle aspirates of breast masses. The dataset is used to classify tumors as malignant or benign. Its relatively large size and rich feature set make it a standard benchmark in medical diagnosis machine learning research [4].

Fetchable datasets

Fetchable datasets are datasets not included with the library by default, but can be downloaded from external repositories or data portals. These datasets tend to be larger and more complex, providing more realistic scenarios for algorithm testing and benchmarking.

California housing dataset

This regression dataset contains information collected during the 1990 U.S. Census, including demographic and housing information for various districts in California. It consists of 20,640 samples with eight numerical

features such as median income, house age, and average rooms. The target variable is the median house value, making this dataset a popular choice for evaluating regression models [5].

20 Newsgroups dataset

This dataset consists of around 20,000 newsgroup posts distributed across 20 different topics, providing a rich corpus for text classification tasks. Due to its size and diversity, it is widely used in NLP research to benchmark document classification algorithms [6].

Labeled Faces in the Wild (LFW) dataset

Comprising over 13,000 face images of more than 5,700 individuals, this dataset is used for face recognition and verification research. Images vary in pose, lighting, and expression, making it a challenging real-world dataset for computer vision applications [7].

Synthetic datasets

Synthetic datasets are programmatically generated data tailored to specific learning problems or scenarios. They allow researchers to systematically control data properties such as class imbalance, feature correlations, noise, and cluster separability. Synthetic datasets are essential for algorithm testing, sensitivity analysis, and educational demonstrations.

I. make_classification(): This function generates multiclass or binary classification datasets with user-defined parameters controlling the number of informative features, redundant features, clusters per class, class separation, and noise level. It is beneficial for testing classifier behavior under various controlled scenarios [8].

II. make_regression(): This function creates regression datasets by generating random linear relationships between features and targets, with options to add noise and control the number of informative features. It enables testing regression algorithms' robustness to noise and feature relevance [9].

III. make_blobs(): Used to generate isotropic Gaussian blobs for clustering problems. It allows specifying the number of centers, cluster standard deviation, and sample size. This dataset is widely employed to evaluate clustering algorithms such as K-Means and DBSCAN [10].

The categorization of datasets in Scikit-learn supports a broad spectrum of machine learning tasks and educational needs. Built-in datasets provide a quick and easy way to test algorithms and demonstrate concepts. Fetchable datasets allow the application of machine learning to realistic and larger-scale problems, enabling the development of robust and generalizable models. Synthetic datasets offer flexibility for systematic experiments and validation under controlled data conditions.

Selecting the appropriate dataset category depends on the research goal, task complexity, and desired level of realism. For newcomers and educational purposes, built-in datasets are recommended. For applied research and benchmarking, fetchable datasets provide more challenging data. For algorithm development and stress-testing, synthetic datasets serve as powerful tools.

4 | Overview of Key Scikit-Learn Datasets and Their Applications

Scikit-learn offers a rich collection of datasets that serve as foundational tools for the development, evaluation, and benchmarking of machine learning algorithms. These datasets span a diverse range of domains, complexities, and problem types, providing users with the ability to test and refine models under varying conditions. The appropriate selection of datasets is critical, as it directly influences the generalizability and performance of machine learning models. This section presents an in-depth examination of several key datasets within Scikit-learn that are widely recognized for their utility in classification, regression, and NLP tasks. Moreover, the discussion highlights which algorithms have proven effective on these datasets, offering practical guidance for researchers and practitioners alike.

The datasets included in the Scikit-learn library play a fundamental role in the development and evaluation of machine learning algorithms. These datasets exhibit considerable diversity, enabling the assessment of model performance across various tasks. Selecting an appropriate dataset for each machine learning problem directly impacts model quality and accuracy. Several representative Scikit-learn datasets are introduced here, and their practical applications are elaborated. The primary focus lies on datasets useful for classification, regression, and text analysis tasks. Additionally, the algorithms that demonstrate optimal performance on each dataset are examined. This study provides a clearer perspective on dataset and model selection for real-world projects.

One of the most well-known datasets is the Iris dataset, which contains information about 150 iris flowers belonging to three different species. The numerical features enable evaluation of multiclass classification algorithms. Algorithms such as KNN, SVM, and Decision Trees perform exceptionally well on this dataset. Due to its small size and simple structure, it is highly suitable for initial model training and testing. Furthermore, the Iris dataset serves as a standard benchmark for classification methods. Its widespread application has led to extensive educational resources centered around it. These characteristics make Iris a starting point for many research studies and educational activities.

The Breast Cancer Wisconsin dataset is another significant dataset extensively used in medical diagnosis applications. It contains 569 samples with 30 numerical features, aiming to classify samples into benign or malignant categories. Machine learning algorithms such as Logistic Regression, SVM, and Random Forest perform well on this dataset. Additionally, boosted algorithms like XGBoost and neural networks yield satisfactory results. The dataset's complexity and high dimensionality make it a suitable candidate for evaluating advanced methods. This dataset plays a crucial role in cancer research and disease modeling. Consequently, proficiency with this dataset is essential for researchers in the medical and data mining fields.

The California Housing dataset is a prominent example in regression tasks. It includes economic and demographic features from over 20,000 samples, with the target being the prediction of median housing prices in various regions. Linear regression, Random Forest, and Gradient Boosting algorithms perform well on this dataset. Additionally, deep neural networks are employed to model the nonlinear complexities inherent in the data. This large dataset allows testing algorithms at an industrial scale. Due to the data's complexity, selecting appropriate algorithms and tuning parameters is crucial. California Housing serves as a benchmark for evaluating regression models in numerous studies.

The Digits dataset is a notable example for image classification tasks, consisting of handwritten digits from 0 to 9. With 1,777 samples and 8×8 pixel dimensions, it presents pattern recognition and computer vision challenges on a small scale. Algorithms such as KNN, SVM, and Random Forest perform adequately on this data. For more advanced projects, Convolutional Neural Networks (CNNs) are employed to improve accuracy. This dataset serves as a standard example for training and evaluating image classification algorithms. Its use facilitates the development of novel methods in computer vision. Additionally, it has extensive educational applications.

The 20 Newsgroups dataset is one of the most essential datasets in NLP and text classification. It contains approximately 20,000 news documents related to 20 different topics. The dataset presents challenges such as linguistic, topical, and structural diversity of documents. Algorithms like Naive Bayes, SVM, and neural network-based methods perform notably well on this data. Moreover, recent advances using Transformer models have significantly improved classification accuracy. This dataset underpins numerous NLP research efforts and competitions. Mastery of this dataset enhances skills necessary for text-based machine learning applications.

Synthetic datasets generated using functions such as make_classification in Scikit-learn are potent tools for algorithm testing and analysis. These datasets allow fine control over parameters, including the number of features, classes, noise level, and class imbalance. Their primary application is simulating diverse scenarios and evaluating algorithm sensitivity to data variations. All major classification algorithms can be applied to these datasets, yielding good results. They are also highly beneficial for teaching foundational machine learning

concepts. Using synthetic data provides complete experimental control, which is often impossible with real-world data.

Selecting appropriate data and related algorithms requires a deep understanding of data characteristics and problem structure. Small and simple datasets like Iris are suitable for quick training and testing of algorithms; however, larger and more realistic datasets are necessary for industrial and complex problems. Large datasets such as California Housing increase feature complexity and pose greater modeling challenges. Moreover, textual and image data require specialized algorithms like language models and CNNs. Consequently, real-world projects typically employ a combination of datasets and algorithms to achieve optimal results.

The practical applications of Scikit-learn datasets in machine learning model development are extensive. These datasets enable developers and researchers not only to evaluate their algorithms but also to play a key role in student education and interactive training development. Using these datasets as standards allows for comparison of results across different studies, enhancing research quality. Additionally, their easy accessibility and comprehensive documentation have made these datasets highly popular in the machine learning ecosystem. Ultimately, leveraging these datasets aids in improving model performance and accelerating development processes.

Ultimately, Scikit-learn datasets provide a valuable resource for all levels of machine learning users. From beginners to advanced researchers, these datasets enable experimentation, education, and algorithm development within a standardized and reliable environment. Their diversity and ease of use make them ideal choices for testing novel methods. The future of machine learning necessitates access to high-quality, comprehensible, and diverse data, a need well met by Scikit-learn. Therefore, thorough study and utilization of these datasets is imperative for success in the machine learning domain.

Visualization is a crucial step in EDA, as it helps to uncover patterns, distributions, and relationships within datasets. In this section, we present graphical analyses for three prominent datasets from Scikit-learn: Iris, Breast Cancer Wisconsin, and California Housing. These visualizations provide insight into the structure and characteristics of the data, which is essential for selecting appropriate machine learning models and understanding their potential limitations.

The first visualization is a scatter plot created for the Iris dataset, illustrating two key features: Sepal length and sepal width. Data points are colored based on the species of the flower, allowing for precise observation of class distribution in a two-dimensional feature space. This plot effectively demonstrates the separability of the three species and highlights how distinct clusters emerge based on these morphological measurements.

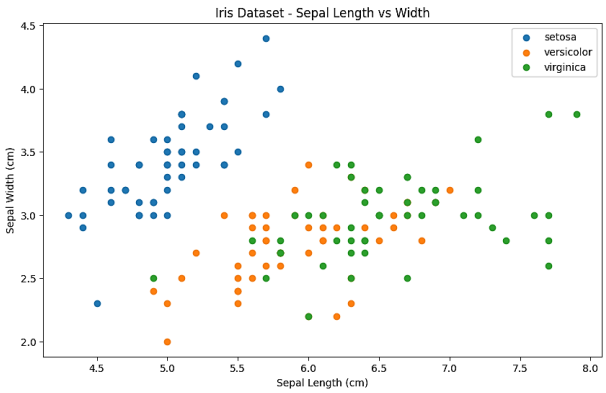

Fig. 1. Iris dataset-Sepal length vs. Width.

The second visualization is a bar chart representing the class distribution in the Breast Cancer dataset. It displays the number of samples classified as benign and malignant. This chart provides a straightforward picture of the dataset's balance, revealing whether the classes are evenly represented or if there is an imbalance that might affect the performance of classification algorithms.

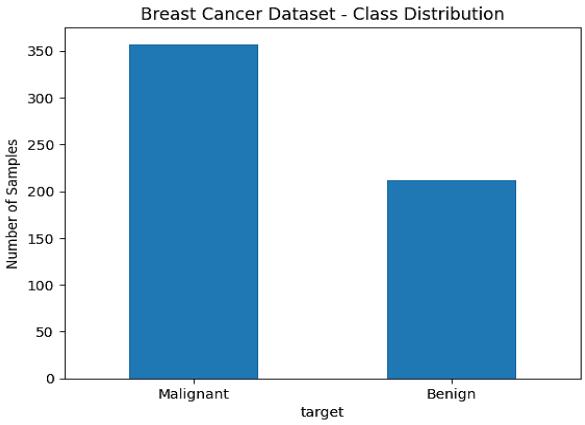

Fig. 2. Breast cancer dataset-class distribution.

The third visualization is a histogram showing the distribution of the target variable in the California Housing dataset, which represents median house prices across different regions. The histogram illustrates how house prices are spread across various intervals, with most prices concentrated in the middle range. This visualization is beneficial for understanding the nature of the regression target and guiding the choice of modeling approaches.

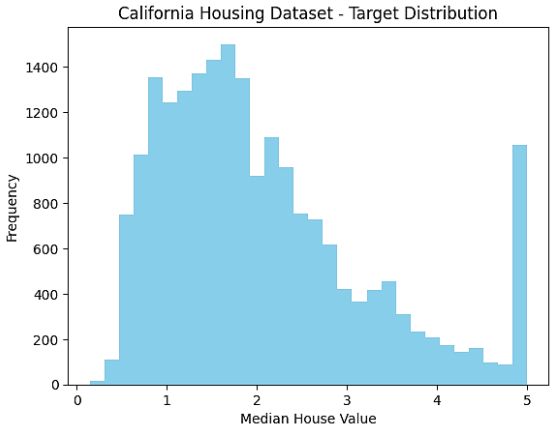

Fig. 3. California housing dataset-target distribution.

Regarding the detailed descriptions of these visualizations, the Iris scatter plot displays each data point as a flower sample, color-coded by species: Setosa, Versicolor, and Virginica. The horizontal axis corresponds to sepal length, and the vertical axis to sepal width. The points for each class form relatively distinct groups, indicating that these two features provide good discriminatory power among species.

The Breast Cancer class distribution bar chart contains two bars representing malignant and benign classes. The bar for the benign class is taller, indicating a larger number of benign samples. This suggests that the dataset is somewhat imbalanced toward the benign class, a factor that should be considered during model training to avoid biased predictions.

Finally, the California Housing histogram depicts the frequency of median house prices grouped into price intervals. Most house prices fall within a central range, while fewer instances exist at very low or very high price values. Understanding this distribution is critical for selecting appropriate regression models and interpreting their performance in predicting housing prices.

This study extensively utilizes official resources from the Scikit-learn library, encompassing a wide range of datasets, including built-in toy datasets, fetchable real-world datasets, and synthetic dataset generators. These datasets are instrumental for educational purposes, experimental evaluations, and benchmarking machine

learning algorithms. The following references include direct links to the official documentation and detailed descriptions of the datasets and functions discussed throughout the paper.

Table 1. Summary of key Scikit-learn datasets with descriptions and references.

Dataset Link Number	Dataset Name	Brief Description
1	Iris	A classic dataset for flower classification with three different species.
2	Wine	A chemical dataset for the classification of three different grape cultivars in Italy.
3	Breast Cancer Wisconsin	A cancer diagnosis dataset with 30 numerical features for benign and malignant classification.
4	California housing	Demographic and economic data for predicting median house prices in California.
5	20 Newsgroups	A collection of around 20,000 news posts for text classification into 20 different topics.
6	LFW	Face image data for face recognition and verification tasks.
7	make_classification	Synthetic data generated for multiclass or binary classification problems.
8	make_regression	Synthetic data for regression problems with control over features and noise parameters.
9	make_blobs	Synthetic data for clustering tasks with Gaussian clusters.

Table 1 presents a summarized overview of prominent datasets available in the Scikit-learn library, highlighting their names and brief descriptions. This compilation facilitates easy identification and access to essential datasets commonly used in machine learning research and education. The listed datasets span a diverse range of domains, including classical classification problems such as the Iris and Wine datasets, medical diagnosis exemplified by the Breast Cancer Wisconsin dataset, real-world regression data like the California Housing dataset, as well as text and image data for NLP and computer vision tasks. Additionally, synthetic datasets generated via functions such as make_classification, make_regression, and make_blobs provide controlled environments for algorithm testing and experimentation. Together, these datasets form a comprehensive resource for developing, benchmarking, and validating machine learning models across various application areas.

5 | Conclusion

This review has systematically explored the primary datasets included in the Scikit-learn library, demonstrating their significance in machine learning model development and assessment. By categorizing datasets into built-in, fetchable, and synthetic groups and applying standardized preprocessing and modeling approaches, we have highlighted their practical utility in classification, regression, and other common tasks. Visualization and exploratory analysis revealed critical dataset properties such as class balance and feature discriminability, which are essential for informed algorithm selection. While Scikit-learn datasets offer excellent educational and prototyping value, addressing real-world problems often requires integration with larger and more complex datasets. Additionally, combining ensemble learning techniques with multi-criteria decision-making frameworks presents promising avenues for optimizing model performance. Ultimately, the comprehensive and accessible datasets provided by Scikit-learn serve as vital resources that support machine learning education and accelerate research and development efforts.

Author Contribution

The author was solely responsible for the conception and design of the study, development of the methodology, implementation of the computational framework, validation of the results, sensitivity analyses, and preparation of the manuscript.

Funding

This work was conducted without any financial support from funding agencies in the public, commercial, or non-profit sectors.

Data Availability

All data generated or analyzed during this study are included in this published article.

Conflicts of Interest

The author declares that there are no conflicts of interest relevant to the content of this article.

References

[1] Fisher, R. (1936). *UCI machine learning repository: Iris data set*. http://archive.ics.uci.edu/ml/datasets/Iris
[2] *Iris dataset*. https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_iris.html
[3] *Wine dataset*. https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_wine.html
[4] *Breast Cancer Wisconsin Dataset*. https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_breast_cancer.html
[5] *California Housing Dataset*. https://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_california_housing.html
[6] *20 Newsgroups Dataset*. https://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_20newsgroups.html
[7] *Labeled Faces in the Wild (LFW) Dataset*. https://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_lfw_people.html
[8] *Make_classification() Synthetic Dataset*. https://scikit-learn.org/stable/modules/generated/sklearn.datasets.make_classification.html
[9] *Make_regression() synthetic dataset*. https://scikit-learn.org/stable/modules/generated/sklearn.datasets.make_regression.html
[10] *Make_blobs() synthetic dataset*. https://scikit-learn.org /stable/modules/generated /sklearn.datasets.make_blobs.html