






Article

CloudPricingOps: A Decision Support Framework to Explore Pricing Policies of Cloud Services

George Fragiadakis ^{*,†}, Anargyros Tsadimas [†], Evangelia Filiopoulou [†], George Kousiouris [†],
Christos Michalakelis [†] and Mara Nikolaidou 

Harokopio Department of Informatics and Telematics, University of Athens, 17778 Athens, Greece; tsadimas@hua.gr (A.T.); evangelf@hua.gr (E.F.); gkousiou@hua.gr (G.K.); michalak@hua.gr (C.M.); mara@hua.gr (M.N.)

* Correspondence: gfragi@hua.gr

† These authors contributed equally to this work.

Abstract: To maximize the business value of the cloud, the cost of cloud solutions is explored alongside technical quality and performance. To enable this form of exploration engineering, finance and business teams collaborate in the context of FinOps, the operational framework that provides the required decision-making. Prominent providers, such as Google and Microsoft, provide FinOps to their customers, integrating cost factors when designing a cloud solution. However, different providers apply different pricing policies for their products, and these policies also change through time. Therefore, there are numerous efforts to explore price evolution through time for different cloud products applying different decision-making methods using different datasets. In an effort to establish a systematic approach to support decision-making on alternative pricing policies for cloud services and compare them across services and providers, the CloudPricingOps framework is proposed in this paper. It constitutes a decision support system that provides alternative decision-making methods, such as hedonic models, time series, and clustering and machine learning techniques, to deal with problems related to cloud product pricing policy analysis, comparison and prediction. It also constitutes a systematic method of discrete steps to integrate additional decision-making methods to deal with these problems and input datasets to be used regardless of the method or the cloud pricing problem that needs to be solved. Two discrete examples based on real data are also presented in this paper to demonstrate the usefulness of the CloudPricingOps framework for cloud engineering and business teams.

Keywords: cloud computing; decision support system; cloud pricing



Citation: Fragiadakis, G.; Tsadimas, A.; Filiopoulou, E.; Kousiouris, G.; Michalakelis, C.; Nikolaidou, M. CloudPricingOps: A Decision Support Framework to Explore Pricing Policies of Cloud Services. *Appl. Sci.* **2024**, *14*, 11946. <https://doi.org/10.3390/app142411946>

Academic Editor: Michele Girolami

Received: 14 November 2024

Revised: 8 December 2024

Accepted: 10 December 2024

Published: 20 December 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Cloud financial management, often referred to as Cloud FinOps (Financial Operations), is a strategic approach to managing and optimizing the financial aspects of cloud computing [1]. It is a cultural practice that is essential for organizations to ensure the efficient use of cloud resources, minimize idle resources, and align spending with business objectives. It enables organizations to derive maximum business value by helping engineering, finance, technology, and business teams collaborate on data-driven decisions [1].

The FinOps framework utilizes a set of domains and capabilities to ensure financial accountability in cloud environments. The domains of the FinOps framework describe the fundamental business outcomes organizations should achieve from the FinOps practice. In essence, practicing FinOps will allow organizations to understand their cloud usage and costs, quantify their business value, optimize both usage and rates paid, and manage effective practices [1]. Capabilities refer to the functional activities that can be performed within each domain, such as forecasting, budgeting, planning, and estimating [2]. A key concept in FinOps is leveraging financial aspects to empower engineering teams to

make data-driven decisions. FinOps emphasizes collaboration between finance, DevOps, and engineering teams to manage cloud spending effectively. Specifically, engineering teams are given detailed information about cloud resource costs. This visibility helps them understand how their technical decisions impact the overall budget. With real-time or frequent cost insights, teams can optimize their usage of cloud resources, scaling up or down based on demand, thus reducing waste and ensuring that resources are being used efficiently. Moreover, engineers are involved in forecasting cloud costs, ensuring that they can predict spending based on upcoming projects or scale needs, which is crucial for long-term planning [1].

The global cloud FinOps market is projected to grow from USD 13.5 billion in 2024 to USD 23.3 billion by 2029, with a compound annual growth rate (CAGR) of 11.4% over the forecast period [3]. In the coming years, several trends are expected to shape this market's growth. A key trend will be the increasing use of artificial intelligence (AI) and machine learning (ML) for analytics and cost forecasting, which will enhance the ability to predict cloud expenditures, identify potential cost-saving opportunities, and allocate resources more effectively based on real-time data and trends [3]. Major cloud providers are implementing mechanisms to manage cloud finance. They offer a suite of FinOps tools that solely focus on the services of each individual provider [4,5]. However, there are third-party FinOps platforms that offer cloud financial management across multiple cloud providers [6,7].

Different providers implement varying pricing policies for their products, and these policies evolve over time. As a result, many initiatives have been launched to examine the temporal evolution of prices for different cloud products, employing diverse decision-making methods and datasets. Towards this direction, we introduce a decision support framework called CloudPricingOps. This framework is designed to enhance decision-making by optimizing cloud costs through a comprehensive analysis of cloud pricing policies. CloudPricingOps includes three capabilities—price policy analysis, cost comparison, and price prediction—all aligned with FinOps culture. It enables decision-makers to make data-driven choices by leveraging well-established methodologies and a reusable dataset. Additionally, it empowers users to manage cloud costs more effectively, ensuring better cost control and strategic financial planning.

The rest of this paper is structured as follows: Section 2 covers related works, followed by the introduction of the CloudPricingOps framework in Section 3. Section 4 offers a detailed examination of the framework's implementation, and Section 5 discusses various case studies that illustrate the framework in action. Lastly, Section 6 concludes the paper, summarizing key findings and implications.

2. Related Work

There are several works that introduce cloud frameworks aligned with FinOps principles, as well as approaches that describe cloud decision support systems. In this context, the relevant literature addresses both of these aspects. Cloud FinOps culture involves identifying opportunities for cost savings by analyzing cloud usage patterns and implementing strategies to reduce unnecessary spending.

2.1. Cloud FinOps Culture

The leading cloud service providers, such as Amazon AWS [8] and Microsoft Azure [9], are incorporating mechanisms in their platforms to optimize the cost of cloud resource usage in alignment with the FinOps concept. In [10], the authors guide enterprise architects, cloud architects, CFOs, CTOs, product managers, finance managers, and FinOps practitioners looking to optimize their use of the AWS cloud. They make readers aware that effectively managing and optimizing costs requires utilizing AWS services to monitor and govern expenses, usage, and spending. Automating processes can streamline cost optimization efforts, and designing an architecture tailored to one's workload while optimizing data transfer is vital. Furthermore, cost savings can be realized by maximizing

efficiency through elasticity strategies, and applying cost optimization levers can help lower computing and storage costs. Ultimately, identifying strategies to create and manage cost metrics adds significant value to an organization.

According to [11], implementing FinOps practices on Microsoft Azure requires a clear understanding of the various activities involved in each phase. This encompasses recognizing architectural patterns for disruptible workloads utilizing Spot VMs and optimizing savings through Reservations, Savings Plans, and Spot VMs. Furthermore, waste analysis can be enhanced with customizable pre-built workbooks. Developing a compelling financial proposal for savings is vital, and applying these details to real-world case studies will improve applied knowledge. Finally, accurately forecasting cloud expenditures, establishing budgets, and monitoring expenses will promote financial accountability.

Moreover, several scholars have explored FinOps practices. In [2], the authors explored the benefits of adopting a cloud FinOps approach, which include improved cost visibility, reduced waste, and better alignment with business goals, ultimately leading to a more efficient and effective use of cloud resources. They revised key principles and practices of cloud FinOps, such as cost optimization strategies, financial governance frameworks, and data-driven decision-making, which help organizations identify cost-saving opportunities and implement financial governance. Moreover, ref. [12] outlined a strategic roadmap for building and scaling FinOps teams, emphasizing the importance of achieving cost efficiency and fostering a culture of cost awareness while maintaining or improving the performance and functionality of cloud-based systems. It highlighted the transformative impact of Cloud FinOps' functionality on organizations, enabling them to maximize the value of their cloud investments and achieve optimal cost efficiency through equipping both technical and non-technical staff with the necessary techniques, procedures, and tools to monitor, report, and analyze cloud expenses.

A cloud FinOps framework was introduced in [13], which focused on optimizing cloud spending and enhancing financial resource management through collaboration between technology and finance teams. It presented a cost analysis for cloud-native near-real-time serverless streaming solutions, emphasizing the importance of understanding the cost implications associated with scaling these solutions to accommodate varying numbers of concurrent users. It underlined that both the streaming and idle times of the system significantly influence overall costs.

Moreover, ref. [14] indicated that cloud computing promoted sustainability through resource consolidation and efficiency improvements, resulting in reduced carbon emissions compared to traditional on-premises infrastructure. It emphasized the importance of leveraging FinOps to align cost optimization with minimizing energy consumption, ultimately positioning responsible cloud adoption as both an ethical imperative and a strategic advantage in reducing environmental footprint.

In [15], the authors discussed the integration of prediction models with the FinOps concept to optimize cloud resource usage by employing machine learning to achieve their aim of predicting long-term cloud resource usage, which is crucial for creating efficient resource reservation plans.

2.2. Cloud Decision Support Frameworks

Considerable research has focused on developing decision support systems (DSSs). In this section, we present key theoretical frameworks for DSS development in the context of cloud computing, along with examples of tools/prototypes based on DSS implementations.

2.2.1. Theoretical Frameworks

The theoretical roots of decision support systems essentially lie in solving a multi-criteria decision-making (MCDM) [16]. The core of multi-criteria decision-making (MCDM) is to assist decision-makers who have various preferences. Typically, MCDM problems do not have a single optimal solution; instead, the goal is to find the most suitable solution based on the decision-makers' preferences. In cloud computing, decision support systems

typically apply multi-criteria decision-making (MCDM) solutions using various methods, such as the analytical hierarchy process (AHP).

The method's effectiveness relies on the appropriate selection of criteria. To aid in this, popular frameworks can be utilized to guide criteria selection, or the primary factors influencing the decision-making process itself can be considered as criteria.

The adoption of cloud computing within the Technology–Organization–Environment framework [17] was explored in [18,19]. Specifically, in [18], the authors developed a decision support model for adopting a cloud computing system by analyzing critical variables in a hierarchical structure. This involved examining decision areas such as technology, organization, and environment, along with seven factors and twenty-three attributes. Additionally, the research explored a comparative analysis between demanders and providers of cloud computing adoption to suggest important factors for successful adoption, such as top management support, competitive pressure, and compatibility. The study employed the Analytic Hierarchy Process (AHP) and Delphi analysis to identify these underlying decision factors for cloud computing adoption.

Additionally, in [19], the authors identified and ranked critical factors affecting cloud computing adoption in the Indian context. A mixed-methods approach, combining qualitative and quantitative techniques, was utilized. Experts from 13 organizations, comprising 8 cloud service providers and 5 cloud service users, participated in interviews. Content analysis helped validate existing factors and reveal new ones relevant to the Indian setting.

In [20], an integrated approach was proposed that combined the Analytic Hierarchy Process (AHP) and the TOPSIS method to identify the optimal machine learning-as-a-Service (MLaaS) cloud. Key functional attributes of MLaaS were established as criteria, with the AHP used to calculate the weights of these criteria, while the TOPSIS method was employed to rank the alternatives.

2.2.2. Decision Support Systems Tools

In addition to the previously discussed literature, there are several prototypes/tools that are closer to practical implementations of decision support systems (DSSs).

In [21], the authors presented a decision-making prototype, called CloudGenius. CloudGenius automated the decision-making process based on cost–performance optimization specifically for web server migration to the cloud. It guided users through a cloud migration process and offered a model and method to determine the best combined and compatible choice of VM images and compute services for a single web server. The authors developed a Hadoop and genetic algorithm-based approach to cope with computational complexities in a growing market of cloud service offerings.

In [22], the authors presented a dynamic DSS for selecting an appropriate cloud storage provider, considering cost and performance. This system is implemented as a web-based application using PHP v.8.3.14 and MySQL v.8.0.33, designed to assist decision-makers in choosing the best cloud storage provider by comparing and ranking them using scraping and parsing modules. The system included a mechanism for ranking cloud service providers, which is a critical component of the decision-making process. This ranking is based on various criteria and parameters that are essential for selecting the most suitable provider.

Moreover, in [23], a decision support system was introduced to assist decision-makers in identifying their concerns and aligning them with suitable tools or techniques. Although the toolkit was designed to support various aspects of decision-making, it addressed only cost modeling. The cost calculation relied on the user defining the application model using UML deployment, which required additional effort. The toolkit's vision included Technology Suitability Analysis, Energy Consumption Analysis, Stakeholder Impact Analysis, Responsibility Modeling, and Cost Modeling.

The InCLOUDER product was specifically designed for cloud migration, allowing organizations to define their migration criteria, such as application architecture, properties, requirements, and cloud service offerings. Based on these criteria and a weighted evaluation

using the Analytic Hierarchy Process, the service generated potential alternatives. The InCLOUDER prototype also incorporated select criteria from the Service Measurement Index (SMI) to help end users make informed decisions [24].

In [25], the authors presented the first framework for a decision support system that integrated risk analysis tools with cloud service selection in multi-cloud environments. They identified specific requirements that served as guidelines for developing cloud-service-selection DSSs. Additionally, they discussed the importance of establishing a unified model for data gathering and curation as an essential component of DSSs. The proposed DSS was tested with real customers in an industrial setting and presented key advantages based on the experiences of independent professionals across two distinct use cases.

In [26], an innovative decision support model was introduced. The proposed decision support model focused on marginal resource allocation, helping cloud providers manage SLAs proactively, before execution. This model addressed all possible scenarios, including whether the consumer was new and whether they requested similar or different marginal resources. It relied on a user-based collaborative filtering approach, enhanced by a top-k nearest neighbor algorithm and a fuzzy logic system, to support decision-making. The proposed framework enabled cloud providers to optimize resource management effectively, reducing the risk of SLA violations or penalties.

2.3. Aim and Contribution

Cloud decision-making frameworks contribute significantly to the effective management of cloud resources, enabling organizations to make informed decisions that enhance operational efficiency, optimize costs, and drive strategic value. Examining current theoretical research and the available tools for implementing the DSS concept indicates that DSS proposals often emphasize various decision-making factors.

In the current work, we propose CloudPricingOps framework, a decision-making model aimed at quantifying business value, specifically focusing on providers' pricing policies. It includes three main capabilities—price policy analysis, cost comparison, and price prediction—and is applicable across a range of cloud services rather than being restricted to a specific one, as seen in previous studies. Additionally, it integrates various established methodologies and reusable data to address the full range of challenges. It transforms vast and intricate datasets into actionable insights, thus empowering stakeholders to make well-informed decisions in a dynamic and competitive market environment.

The overview presented in Table 1 showcases the integration of diverse decision-making methods across various studies in cloud service analysis. As indicated in the table, most of the approaches presented in the available literature target a specific capability of FinOps domain utilizing specific or compatible decision-making methods. Furthermore, they usually focus on specific cloud service types. One could say that, in a way, they limit themselves in a specific problem domain, restricting the reusability of the proposed DM methods in other capabilities (e.g., problem domains) or service types, with any adjustment that might be needed. The current work aims at resolving this issue, building a framework integrating different DM methods, to address multiple capabilities in the Cloud FinOps domain (e.g., problem domains) in diverse service types (e.g., input data). In each current state, it distinguishes itself by integrating multiple analytical techniques, including the hedonic method, clustering, and machine learning approaches, to address pricing analysis, cost comparison, and price prediction. Furthermore, it clusters for extensibility, since it provides the capability to incorporate additional DM methods and utilize them to explore existing or additional FinOps capabilities. It also enables the creation of dataset libraries, ready to be used to explore any supported capability using any integrated DM method. Thus, in essence, it provides a framework to explore different cloud FinOps capabilities, related to price problems, using different DM methods for different cloud service types. In contrast to existing approaches explored in Table 1, the proposed concept promotes reusability and provides an integrated approach when dealing with price-related issues in

the cloud service market. The framework itself is limited by the technical constraints of integrating new DM methods and datasets, usually stemming out of external restrictions.

For instance, while previous studies primarily focused on singular approaches such as AHP or specific analysis like cost or SLA prioritization, the flexibility of the current method allows for combining these methods or extending beyond them. This adaptability is particularly valuable in domains like cloud services, where evolving requirements and datasets demand robust and scalable solutions. Thus, the current work underscores a methodological advancement, serving as a foundational model for integrating new methods seamlessly while ensuring continuity and comparability.

It allows users to make more informed decisions, leading to better satisfaction and trust in the services provided. It is structured to guide stakeholders through the complexities of cloud service selection, using a systematic approach that integrates a range of analytical methods. By offering a structured and integrated analysis, the Cloud PricingOps framework simplifies the complexities of cloud service decision-making. It provides a data-driven foundation that enables stakeholders such as cloud architects, IT teams, and financial managers to navigate the cloud market with confidence, ensuring that their choices are both informed and strategically aligned with their long-term goals.

Table 1. Overview of the literature review.

Refs.	Capabilities	DM Methods	Cloud Service Type	
[20]	Factor Prioritization	AHP, TOPSIS	MLaaS	
[18]	Theoretical Framework	Factor Prioritization	IaaS	
[19]		Factor Prioritization	IaaS	
[21]		Cost Analysis	AHP	IaaS
[23]	Implementation of DSS	Cost Framework	IaaS	
[24]		VM Selection for Cloud Migration	AHP	IaaS
[25]		Risk Analysis	Critical Analysis of Existing DSS	IaaS
[26]		SLA Prioritization	Top-K Nearest Neighbor Algorithm	IaaS
Current Work	Pricing Analysis, Cost Comparison, Price Prediction	Hedonic Method, Clustering, ML Techniques	IaaS/ CaaS/ PaaS	

3. Introducing CloudPricingOps Framework

The Cloud PricingOps framework, as depicted in Figure 1, is developed as a comprehensive decision support system, addressing the intricate challenges of pricing, comparison, and future trend anticipation in the cloud services market. It aims to lead stakeholders through the complexities of choosing cloud services, employing a structured approach that incorporates various analytical methods.

3.1. Stakeholders and Use Cases

The CloudPricingOps framework is tailored to meet the needs of a diverse set of stakeholders, each with unique objectives when it comes to navigating cloud services.

The framework's stakeholders include engineering, finance, and leadership roles, such as cloud architects, financial managers, and IT teams. Engineering roles rely on the framework for insights into resource allocation and performance trade-offs. They use the framework's analytical capabilities to better understand the impact of various service configurations on costs, allowing them to optimize their designs for both performance and budgetary requirements.

Furthermore, leaders in small- and medium-sized enterprises (SMEs) frequently struggle to understand complex cloud pricing models due to a lack of in-house expertise. For these users, the framework simplifies the decision-making process, offering a clear comparison of service options without requiring deep technical knowledge. This enables SMEs to focus on selecting services that meet their operational needs, reducing uncertainty and facilitating more effective cloud adoption.

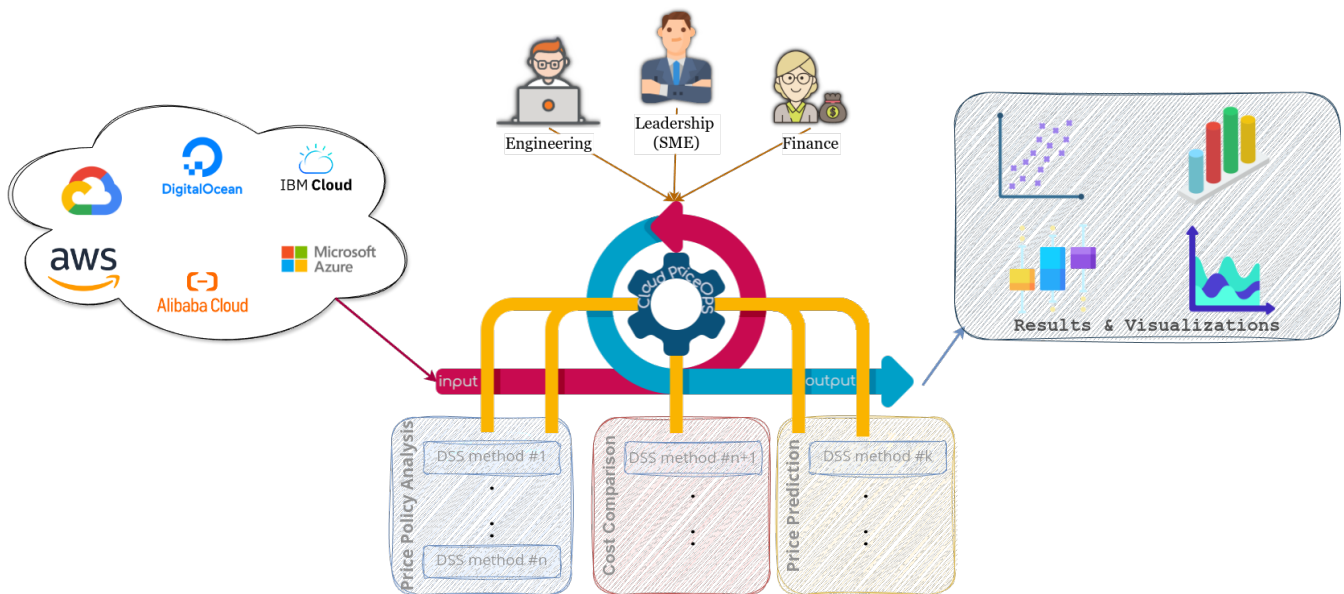


Figure 1. CloudPricingOpS framework—conceptual view.

Finally, finance roles are primarily concerned with controlling costs and ensuring that cloud expenditures are in line with the organization’s financial goals. The framework’s forecasting capabilities provide these users with insights into potential price trends, allowing them to plan budgets with greater accuracy and anticipate future financial requirements. By aligning projected costs with expected market developments, financial managers can better manage risks associated with price volatility.

3.2. Inputs and Outputs

The framework operates by processing various inputs and delivering tailored outputs that directly address the needs of different users. Inputs to the Cloud CloudPricingOps framework typically include data from a wide array of cloud service providers, such as pricing structures, feature sets, and historical pricing trends. These data can be ingested through API integrations or manual uploads, ensuring flexibility in how users interact with the system. The data reflect the complexity and variability of the cloud service landscape, capturing differences in service types, provider-specific features, and time-based pricing variations.

The outputs generated by the framework are designed to deliver precise, user-focused insights that support decision-making. These outputs range from comparative analyses of service offerings to predictions of future pricing trends, presented through a user-friendly interface that includes visualizations, reports, and summary metrics. The aim is to provide stakeholders with not just data but actionable intelligence that allows them to select the most suitable services, anticipate market changes, and optimize their resource allocation and budgets effectively. By converting intricate data into clear and meaningful insights, the framework serves as a reliable guide for navigating the evolving cloud services market.

3.3. Provided FinOps Capabilities

The framework is structured to deliver a multi-dimensional analysis of cloud service pricing focusing on three core capabilities. Each capability targets a specific aspect of cloud service analysis, working in synergy to provide a holistic view of the market.

- *Price policy analysis* is a capability that enables users to understand how different service attributes influence pricing structures, offering insights into the factors that drive cost variations. Specifically, it draws on methods that assess the influence of various service attributes on pricing, offering users a clearer understanding of how different features contribute to overall costs.

- *Cost comparison* is a capability that simplifies the evaluation process by categorizing services, making it easier to identify comparable options and select those that best match specific needs in terms of functionality and cost. Specifically, clustering approaches systematically group similar cloud services, enabling stakeholders to make meaningful comparisons and identify those that offer the best value. Services can be categorized based on various factors, such as size and price, making it easier to evaluate and select the most cost-effective options.
- *Price prediction* is a capability that aims to forecast future price trends, empowering stakeholders to make strategic decisions and align with long-term financial objectives. By adopting statistical and machine learning methods, it analyzes historical data to generate accurate price forecasts, offering valuable insights across various domains for informed decision-making.

4. CloudPricingOps Framework Implementation

CloudPricingOps framework has grown into an extensible DSS that aids in decision-making in the cloud service FinOps domain. The framework should provide a modular environment in which users can add new methods or use the default pool of methods to solve specific problems. This section outlines the framework’s structure and implementation, emphasizing its adaptability for a wide range of use cases.

The system is designed to handle data from various sources, analyze them using established and customizable methods, and provide visualized insights. Figure 2 illustrates the architecture of the CloudPricingOps framework, demonstrating its capability to support different modules, decision-making methods and tools, and also extensions for diverse user requirements.

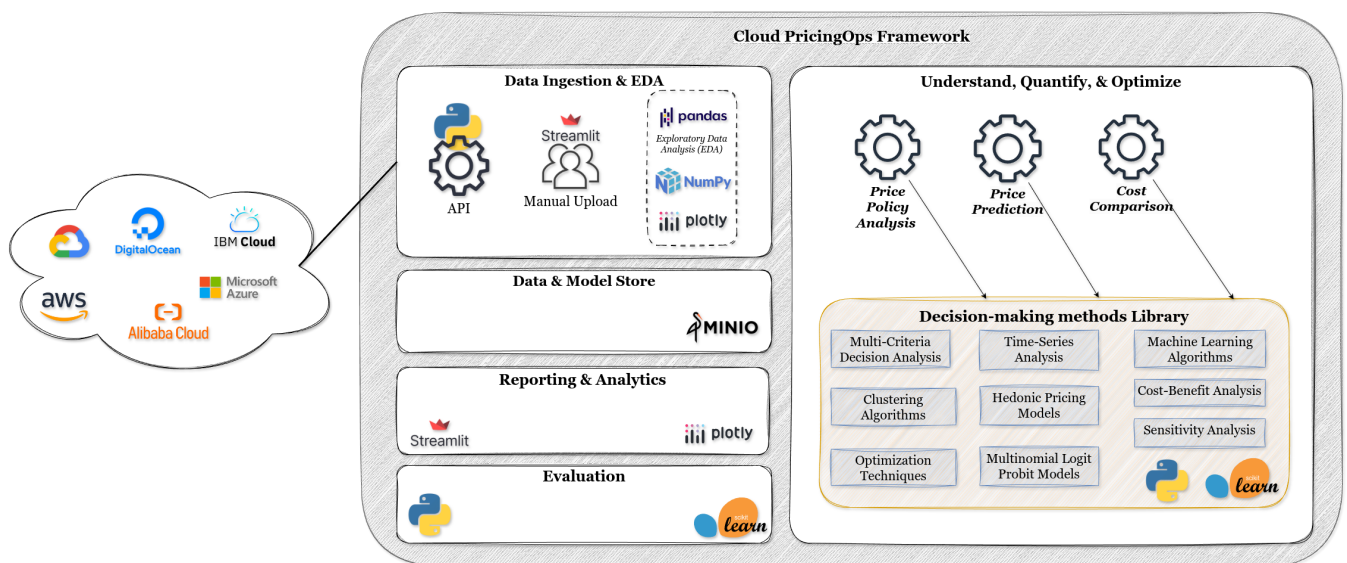


Figure 2. CloudPricingOps framework—system view.

4.1. Scope and Design Requirements

The CloudPriceOps framework is designed to streamline the management and analysis of cloud pricing data from various providers. By integrating robust functional and non-functional requirements, the framework aims to deliver a comprehensive solution that meets the diverse needs of engineers, decision-makers, and accountants seeking to optimize their cloud spending. This document outlines the key functional requirements that define the system’s capabilities, as well as the non-functional requirements that ensure its performance, usability, and maintainability.

4.1.1. Functional Requirements

Functional requirements for the CloudPricingOps framework encompass various critical capabilities that the system must deliver to effectively manage and analyze cloud pricing data. Key functionalities include data integration, which involves fetching data from multiple cloud providers such as AWS, GCP, and Azure through their APIs. The system must also facilitate data processing and analysis by parsing and normalizing these data into a common format, enabling comparisons of pricing across different providers based on user-defined criteria. User interaction is another essential aspect, allowing users to specify parameters like resource type, region, and usage patterns for cost analysis. Additionally, features for visualization are crucial; the system should present data through charts, graphs, and tables to illustrate cost breakdowns and trends, including side-by-side comparisons of providers. Finally, effective system management is necessary, providing an admin interface for API key management and integration updates, alongside integrating decision-making method libraries to evaluate various decision-making approaches.

4.1.2. Non-Functional Requirements

Non-functional requirements define the quality attributes that the CloudPricingOps framework must meet to ensure a robust user experience and system performance. Performance is a primary concern, necessitating the ability to handle large volumes of data from APIs with minimal latency. The system's usability is also critical; it should feature an intuitive interface designed for non-technical users, complete with tooltips and comprehensive documentation. Furthermore, maintainability is emphasized through a modular architecture that simplifies updates and API integration while implementing automated testing for any changes. The framework should also be extensible, allowing for easy addition of new cloud providers or pricing models with minimal effort, and providing APIs or SDKs for third-party tool integrations. Collectively, these non-functional requirements ensure that the system not only performs effectively but also remains user-friendly and adaptable to future needs.

4.2. CloudPricingOps Framework Modules

The CloudPricingOps framework's architecture was made to be modular and flexible, so it can easily accept different decision support system (DSS) methods for studying and improving the prices of cloud services. The framework supports various domains of analytical problem, including price policy analysis, price prediction, and cost comparison. This adaptability ensures that the framework can evolve with emerging needs and incorporate advanced methods as they become relevant. It is implemented using Python v.3.10.8, using the large number of libraries of DM methods already available. It consists of six modules, briefly introduced in the following.

4.2.1. Understand, Quantify, and Optimize—DM Libraries

The CloudPricingOps framework's **Understand, Quantify and Optimize** module consists of three main capability pillars: **price policy analysis**, **price prediction**, and **cost comparison**. These are flexible modules that can accommodate different methods, allowing users to choose the most suitable one from the **CloudPricingOps Library** for their analysis.

- **Price Policy Analysis:** This pillar focuses on understanding the key drivers behind cloud service pricing by evaluating the impact of various attributes. Users can leverage *hedonic pricing models* to assess the contribution of features like CPU, memory, storage, and geographical location on overall pricing. *Multinomial logit and probit models* are available to model probabilities and analyze the likelihood of choosing specific configurations based on price sensitivity. *Factor analysis and sensitivity analysis* can help users identify and prioritize the features that influence the price variations, thus supporting decisions to optimize configurations.

- **Cost Comparison:** This module provides insights into cost effectiveness by comparing services between providers and configurations. *Clustering algorithms* (e.g., K-Means, Hierarchical Clustering) categorize services with similar attributes, helping users quickly identify comparable offerings from different providers. *Multi-criteria decision analysis (MCDA)* can be used to classify services based on multiple criteria, such as cost, performance, and availability, supporting more tailored decision-making. *Cost–benefit analysis* aids in evaluating the financial trade-offs among services, empowering users to make choices aligned with their budget and performance goals.
- **Price Prediction:** This module is dedicated to forecasting future pricing trends, enabling stakeholders to anticipate and plan for pricing shifts. *Time series analysis* and *machine learning algorithms* (e.g., CatBoost, Decision Trees, and Random Forests) are available for developing robust predictive models, helping users forecast costs based on historical data. For dynamic and evolving markets, *optimization techniques* like linear programming and genetic algorithms allow users to simulate and identify optimal configurations in anticipation of future demand and pricing changes. *Sensitivity analysis* within this module can show how potential changes in key cost factors, such as usage volume or resource type, will impact future prices, aiding in proactive budget planning.

Each of these analytical modules is built on a diverse library of decision-making methods, providing users with the flexibility to adapt the framework to various analytical needs. The modular and extensible design of the Cloud PricingOps framework enables users to incorporate new methods as they become available or tailor analyses to specific use cases, ensuring that the framework remains relevant and adaptable to emerging trends and evolving requirements in cloud cost management.

4.2.2. Data Ingestion and Exploratory Data Analysis (EDA)

The **Data Ingestion and EDA** module enables the integration of data from multiple cloud providers—such as AWS, Google Cloud, IBM Cloud, and others. Data can be imported either through *API connections* or manual uploads. The module is implemented by a set of robust libraries including *Pandas* v. 2.2.0 and *NumPy* v.2.1.0 for data handling, and *Plotly* for initial visual exploration.

By providing a user-friendly interface through *Streamlit*, the system allows stakeholders to quickly conduct exploratory data analysis, generating key statistical summaries and visualizations. This process is crucial for understanding the structure of the datasets and preparing them for the next phase.

Processed data from the EDA phase is stored in the **Data and Model Store** using *MinIO*. This makes it easy to retrieve the data for later analysis or iterative refinement.

4.2.3. Data and Model Store

The **Data and Model Store** serves as a centralized repository for processed data and resulting models from each analysis phase. This component ensures that users can revisit past analyses, refine models, or apply stored data to new problems, fostering a reusable and iterative approach to cloud service analysis. *MinIO*, a high-performance object storage, provides the necessary infrastructure for efficient storage and retrieval.

4.2.4. Reporting and Analytics

The **Reporting and Analytics** phase is where the processed data are transformed into actionable insights. Using *Plotly* v. 5.21.0 and *Streamlit* (v.1.38), the system offers an interactive interface that displays the outcomes of the chosen analysis method. Visualizations include dynamic graphs, heatmaps, and comparison tables, designed to be intuitive and informative.

The modular architecture allows users to easily switch between different visualization techniques, facilitating a tailored experience that suits their decision-making requirements.

4.2.5. Evaluation

The evaluation subsystem is utilized by the CloudOps engineer to incorporate and validate new Decision Support methods within the CloudPricingOps framework. This subsystem employs a structured approach to assess methods using various evaluation metrics tailored to their type. For instance, clustering algorithms like K-Means are evaluated using metrics such as silhouette scores, Davies–Bouldin Index, or inertia, which measure cluster cohesion and separation [27]. Regression-based methods, such as Ordinary Least Squares (OLS), are assessed using R-squared values, mean absolute error (MAE), and mean squared error (MSE) to determine prediction accuracy and residual patterns [28]. For machine learning models used in price prediction, metrics such as precision, recall, F1-score, and area under the ROC curve (AUC-ROC) are utilized for classification tasks, while root mean squared error (RMSE) and mean absolute percentage error (MAPE) are applied for regression [29].

The evaluation process ensures that methods align with the framework’s objectives by validating their performance on historical datasets and by stress-testing them against outlier cases. If the method does not meet the predefined standards, such as achieving acceptable thresholds in the chosen metrics, iterative refinements are conducted. These refinements may involve parameter tuning, feature engineering, or even substituting the method with a more appropriate one from the library. Additionally, evaluation results can be stored for benchmarking purposes, providing stakeholders with insights into the relative performance and suitability of each method for specific decision-making contexts. This structured and transparent evaluation process is critical for ensuring the robustness, reliability, and adaptability of the framework across diverse analytical scenarios.

4.3. CloudPriceOps Framework Implementation

Figure 3 illustrates a use case diagram for the CloudPriceOps framework, highlighting the key actors involved and the operations they perform, utilizing the functionality provided by its modules. In accordance with Figure 2, the CloudPriceOps framework uses a case diagram that comprises six subsystems that work together to enhance cloud pricing operations.

CloudOps users are categorized into three distinct groups: solution architects, decision-makers, and accountants. Each group utilizes the platform’s user interface to perform the understand, quantify, and analyze activity on PriceOps. Their activities include analyzing pricing policies, comparing costs among different cloud providers, and predicting future prices. This structured approach ensures that all stakeholders can effectively leverage the system’s capabilities to optimize cloud spending. This version enhances clarity and flow while emphasizing the roles and functions within the CloudPriceOps framework. The CloudOps engineer acts as the system administrator, responsible for (a) the integration and evaluation of decision-making methods to be used by a PriceOps understand, quantify, and analyze activity and (b) the ingestion of new datasets.

As shown in Figure 3, the primary subsystem is Cloud PriceOps, which serves as the main interface for CloudOps users. Each PriceOps activity of the Cloud PriceOPS subsystem utilizes a DM method provided by the decision-making subsystem, using pre-trained DM models and input datasets, already added by the CloudOps engineer in the Data and Model Store, responsible for storing data from cloud providers as well as the relevant decision-making models. It also uses activities supported by the Reporting and Analysis Subsystem, responsible for generating and visualizing reports. The CloudOps engineer invokes the decision-making subsystem to integrate a new DM method into the framework. To do so, the Integrate DM method activity also utilizes data to evaluate and refine the DM method for the supported cloud pricing activities. The trained and evaluated model is stored in the Data and Model Store. This comprehensive structure facilitates efficient data management and analysis within the framework and enables the CloudOps user to seamlessly explore PriceOps capabilities without having to deal with the DM method’s training and evaluation or input data management. This is taken care of by

the engineer of the CloudPriceOps framework and provided as a service to the framework’s users.

Additionally, the vision for the framework extends to a web application, which would offer an intuitive interface for cloud practitioners to interact with the framework’s features. This application would allow users to configure methods, upload datasets, and access dynamic visualizations, enhancing usability and accessibility for stakeholders. In the following, we focus on the activities performed by a CloudOps engineer to further explore the functionality provided by the proposed framework.

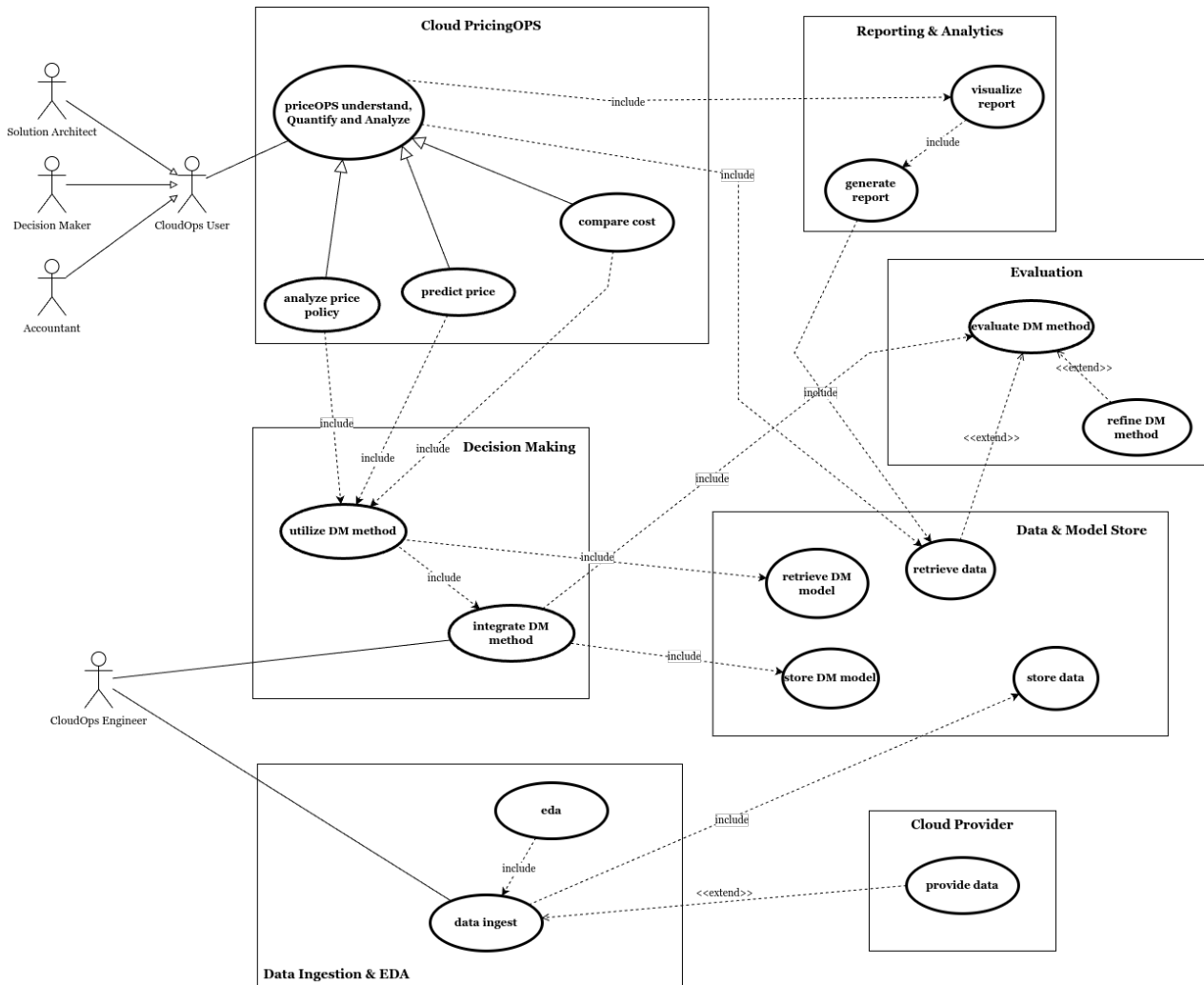


Figure 3. CloudPricingOps framework’s provided functionality—use case diagram.

4.3.1. Data Ingestion and EDA Activity

The Data Ingestion and EDA subsystem enables the integration of data from multiple cloud providers—such as AWS, Google Cloud, IBM Cloud, and others. Data can be imported either through *API connections* or manual uploads. Figure 4 illustrates an activity diagram that analyzes the “data ingest” use case depicted in Figure 3. In this process, the CloudOps engineer begins by selecting the target cloud provider from which data will be ingested. Next, the engineer initiates the “fetch data” action, followed by the exploratory analysis and data analysis actions. This sequence continues until the data are successfully stored in the Data and Model Store subsystem.

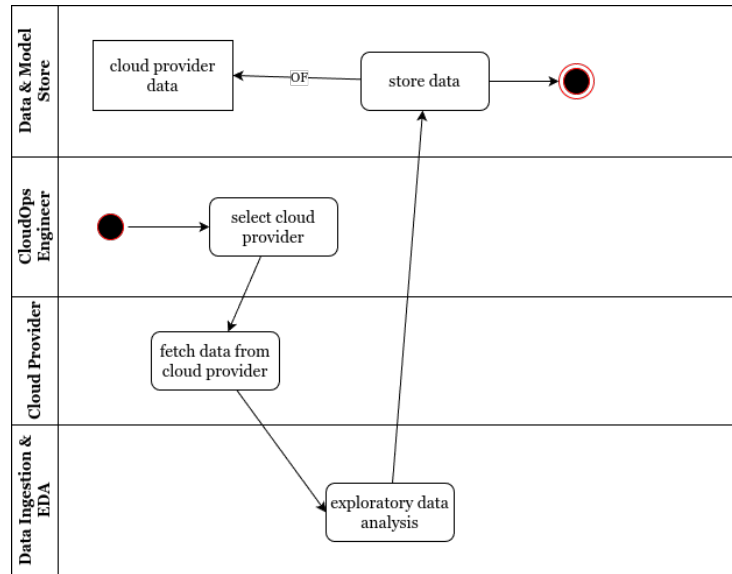


Figure 4. CloudPricingOps data ingestion activity diagram.

4.3.2. DM Method Integration Activity

The CloudCloudPricingOps framework is designed to be adaptable, allowing for the integration of various decision-making methods tailored to specific problem domains such as price policy analysis, price prediction, and cost comparison. This adaptability ensures that the framework can evolve with emerging needs and incorporate advanced methods as they become relevant. Figure 5 presents the process of selection and integration of a DM method implemented as a Python library into the CloudCloudPricingOps framework.

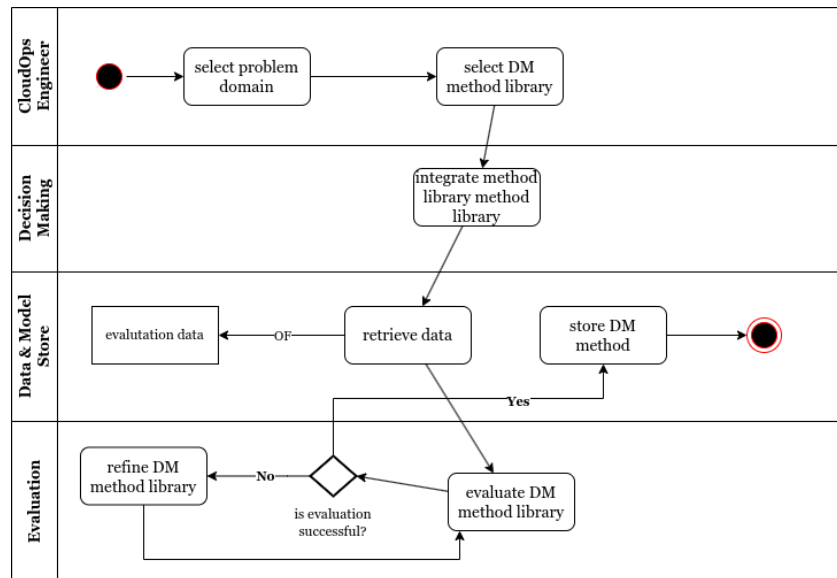


Figure 5. Integrate DM method’s activity diagram.

The integration process begins by identifying the relevant problem domain and selecting an appropriate decision-making method aligned with that domain. The selected DM method library is then integrated into the framework, enabling further evaluation of its effectiveness through a structured assessment process. This evaluation step involves testing the method on sample data, applying statistical metrics (e.g., accuracy, performance consistency), or using specific criteria aligned with the chosen method. If the method does not meet the evaluation criteria, refinements are made to optimize its performance.

Upon successful evaluation, the DM method can be applied to the dataset at hand, enabling users to derive actionable insights and present results effectively. This structured approach ensures that CloudCloudPricingOps remains versatile and robust, capable of incorporating advanced techniques such as time series analysis or multi-criteria decision analysis to support informed decision-making across various contexts.

4.4. User Interaction and Extensibility

One of the key strengths of the CloudPricingOps framework is its user-friendly interface, developed with *Streamlit*. This interface is not only intuitive but also highly customizable, giving users the flexibility to perform the following actions:

- **Choose the Analysis Method** : Users can select from the existing library of methods or seamlessly integrate new analytical techniques tailored to their specific problem. The framework's modular design allows for the addition of novel methodologies within each analysis pillar—whether its price policy analysis, cost comparison, or price prediction—providing a robust platform for experimentation and customization.
- **Adjust Service and Provider Criteria**: Whether dealing with IaaS, PaaS, or SaaS, users can specify the type of cloud services and providers they are interested in. The extensibility of the framework means that criteria can be adapted to new or emerging service models, reflecting shifts in the cloud market landscape.
- **Utilize Custom Data**: Users can import personal datasets or extend the analysis to new data sources seamlessly. The framework supports a wide variety of data formats, ensuring that both standardized and customized data inputs can be used without any compatibility issues.

This extensible design ensures that the framework can adapt to various cloud service evaluation scenarios, meeting the diverse needs of engineers, leaders, and finance departments.

5. Case Studies

The modular design of the CloudPricingOps framework enables diverse analytical applications tailored to the cloud services market. To illustrate its flexibility and adaptability, we present two case studies that showcase the framework's capacity to generate targeted insights for different decision-making scenarios. We show that the framework can be used again and again by using the same dataset in both case studies. This also shows that it can give consistent and reliable results using different analytical approaches.

5.1. Dataset Description

The datasets used in both case studies is based on real-world data collected directly from cloud providers' web calculators. These calculators are publicly accessible tools used by providers to present their pricing policies and configurations to customers. Specifically, the datasets aggregate service bundle offerings from major cloud service providers, including Amazon Web Services (AWSs), Microsoft Azure, Google Cloud, IBM Cloud, Alibaba Cloud, and DigitalOcean.

Three discrete datasets were constructed for each discrete cloud service type. The Infrastructure as a Service (IaaS) dataset includes 589 bundles, the Container as a Service (CaaS) dataset includes 640 bundles, and the Platform as a Service (PaaS) dataset includes 806 bundles. Each dataset service bundle comprises both functional and non-functional attributes, such as CPU, memory, storage capacity, geographic region, and pricing options like reserved or spot instances. These attributes provide a comprehensive basis for analyzing the pricing strategies of cloud providers. The attributes may differ for each service type; thus, the structure of each dataset is different.

These case studies not only show the framework's analytical capabilities, but they also show how easy it is to use and how flexible its input options are, allowing data to be added easily through manual uploads or API integrations. The modular approach also facilitates incremental extension, enabling users to integrate new analytical techniques tailored to their

specific needs. This aligns with FinOps principles by supporting continuous improvement, adaptability, and efficient decision-making in cloud service evaluation.

Authenticity and Preprocessing

The data were collected over the period Q3–Q4 of 2023 using the official pricing calculators of the listed cloud providers. This ensures that the datasets reflect real configurations and pricing policies at the time of collection. After collection, the datasets underwent a systematic exploratory data analysis (EDA) phase to ensure data quality, remove inconsistencies, and prepare it for analysis. This process guarantees that the data are both accurate and representative of the cloud services market.

5.2. Case Study 1: Price Policy Analysis

5.2.1. Problem Domain Selection

In this case study, a price function is explored to depict the factors affecting the price of different cloud service types. The same DM method may be used to construct discrete price functions for each service type. We showcase the CloudPricingOps framework's capability of integrating and evaluating a DM method to implement the price function capability (see Table 1) and offer it for different cloud service types, e.g., IaaS, PaaS and CaaS, that is, different datasets.

5.2.2. Method Selection and Integration

Within the price policy analysis capability, we determined that for constructing the price function, the Ordinary Least Squares (OLS) regression method from the Scikit-learn library was a good candidate due to its effectiveness in quantifying the influence of specific service attributes (e.g., CPU, memory, storage) on pricing. This selection was based on a preliminary evaluation of method suitability for handling varied cloud service data, including IaaS, PaaS, and CaaS offerings. The integration process followed the steps described in Figure 5.

5.2.3. Evaluation and Refinement

The selected OLS regression method was integrated into the framework and underwent an initial evaluation using a sample subset of the dataset. The evaluation process involved checking the statistical significance of the coefficients and ensuring that the model fit the data well across different service types. If the evaluation had not met the required performance metrics (e.g., R-squared value or significance level), the method would have been refined or replaced, following the iterative refinement loop depicted in Figure 5. However, the initial evaluation confirmed that OLS regression was well suited for this analysis, enabling us to proceed.

Specifically, the evaluation of the regression model was conducted using R-squared (coefficient of determination) and mean absolute error (MAE). The R-squared value provided insights into how well the selected features explained the variability in pricing across cloud services, while the MAE quantified the average difference between the predicted and actual prices. These measurements proved that the Ordinary Least Squares (OLS) regression method worked, and that the coefficients that were found were strong and accurate for figuring out how features like CPU, RAM, and storage space affect prices.

While the OLS regression method may be extensively evaluated and refined using the dataset of IaaS, it can be seamlessly applied to the CaaS and PaaS datasets with minimum retraining.

5.2.4. Reporting and Analytics

The final step involved presenting the results to stakeholders, including cloud architects, financial analysts, and leadership teams. The radar charts served as intuitive visual tools, showcasing key differences in cost drivers across providers and service types. This presentation aligns with the decision-making support process of the framework, providing

clear, data-driven insights that assist in optimizing cloud strategies. The insights were visualized using radar charts to enable stakeholders to easily interpret the results.

5.2.5. Application and Insights

The validated method was applied to all three datasets corresponding to IaaS, PaaS, and CaaS service types, already integrated in the Data and Model Store. This method made it easier to explore the same problem for different datasets. It should be noted that since each dataset contains different attributes, the price function contains different attributes. Detailed presentation of the price functions is explored in [30].

Using the *price policy analysis* activity, a PriceOps user may explore the coefficients for each service attribute, providing a detailed breakdown of their impact on overall pricing. More importantly, it makes it easier to compare the impact of the same attributes, for example, CPU, memory, and location, with the prices across different datasets.

This kind of analysis, expanded across all three cloud service types (IaaS, CaaS, and PaaS), is shown in Figure 6. This comprehensive comparison enables stakeholders to understand how the influence of attributes shifts across different service models, facilitating both intra-service and cross-service evaluations.

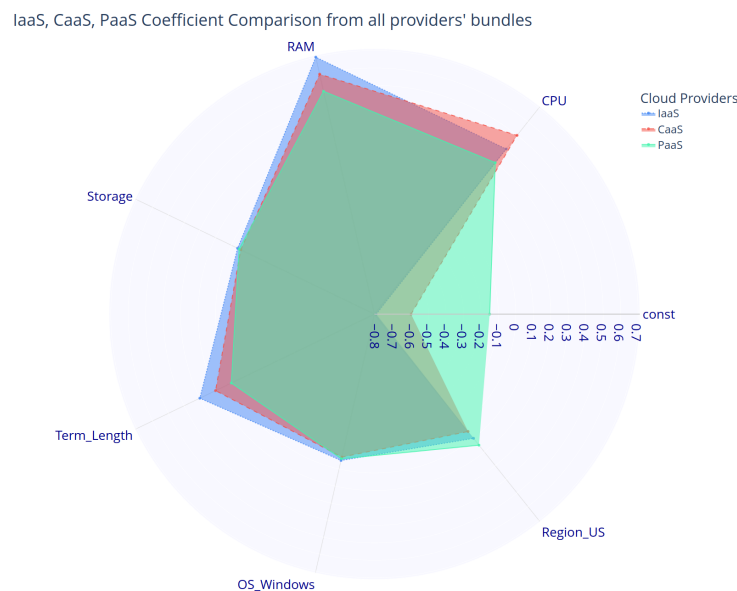


Figure 6. Comparison of coefficients across IaaS, CaaS, and PaaS datasets.

5.3. Case Study 2: Cost Comparison Using Clustering

5.3.1. Problem Domain Selection

In this case, we focused on comparing cloud service bundles based on performance metrics and costs. By clustering services with comparable performance attributes, the analysis provides stakeholders with a clearer picture of the cost–performance trade-offs, enabling more informed decisions [31]. In this case, the designer could choose between IaaS and CaaS solutions. This case study demonstrates the CloudPricingOps framework’s capability to use the same datasets to tackle different PricingOps issues—in this case, the IaaS and CaaS datasets.

5.3.2. Method Selection and Integration

Within the *cost comparison* module of the framework, multiple clustering algorithms were evaluated, including K-Means, DBSCAN, and Hierarchical Clustering. The evaluation phase involved running these algorithms on a sample subset of the dataset and comparing their performance based on metrics such as silhouette score and inertia.

K-Means clustering was selected as the optimal method due to its superior performance, particularly in creating distinct, interpretable clusters based on CPU and RAM

metrics. This choice is supported by our previous detailed study [31], where K-Means demonstrated the best balance between accuracy and computational efficiency for cloud service categorization. The framework’s extensibility allowed for the seamless integration of this algorithm, showcasing its ability to accommodate the best-suited method for the problem at hand.

Using the same datasets as in the first case study (Section 5.2), the cost comparison module was configured to focus on the most impactful attributes identified earlier: CPU, RAM, and storage. The modular design of the framework facilitated a smooth transition from price policy analysis to clustering, enabling a focused examination of performance–cost relationships [31].

The selected K-Means algorithm was applied to categorize the service bundles based on their CPU-RAM and storage–disk type combinations. The analysis generated clear clusters, providing a structured view of how different service configurations compare in terms of performance and cost.

5.3.3. Evaluation and Refinement

The clustering results were rigorously evaluated using silhouette scores to validate the cohesion and separation of the clusters. The silhouette score measures how similar each point in a cluster is to points in its own cluster compared to those in other clusters. A higher silhouette score indicates well-separated and cohesive clusters. The results confirmed that the K-Means algorithm effectively grouped services into distinct categories, such as ‘xsmall’, ‘small’, ‘medium’, and ‘large’, based on their CPU and RAM specifications.

5.3.4. Reporting and Analytics

The clusters were visualized in a table format, as shown in Figure 7, presenting the average price and cost breakdowns for each category.

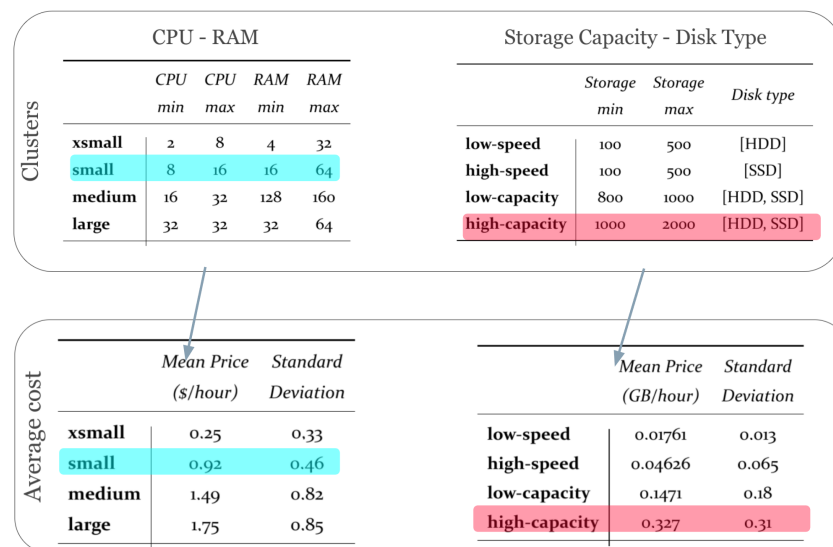


Figure 7. Clustering categories based on CPU-RAM and storage capacity with average costs.

The table shows the average price and standard deviation for CPU-RAM and storage capacity clusters. For instance, the ‘xsmall’ category, with minimal CPU and RAM resources, has an average cost of USD 0.25/h, while larger configurations like ‘large’ services cost around USD 1.75/h.

5.3.5. Application and Insights

The practical implications of the clustering results are evident in designing applications for IaaS and CaaS platforms. Figure 8 illustrates how the clustering analysis can be applied to a demo application, where total resource requirements (CPU, RAM) are compared across

services from different providers. This enables cloud architects to choose between IaaS and CaaS service types and to select the most appropriate service configurations based on the required specifications while considering the associated costs.

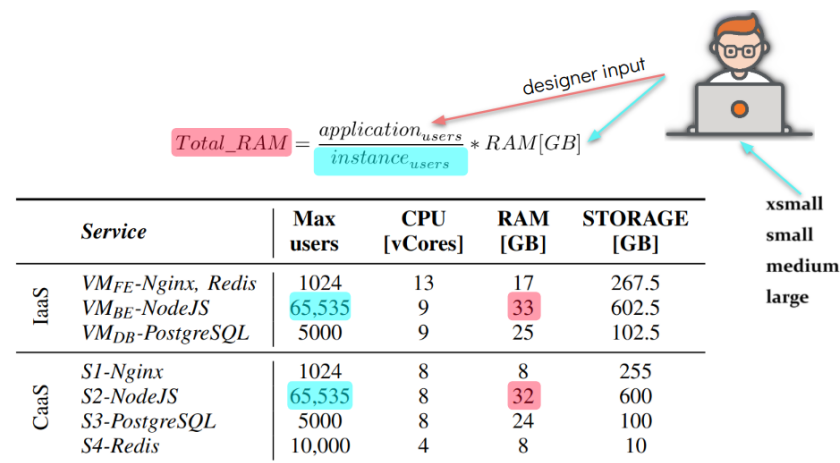


Figure 8. Application resource requirements compared across services.

By leveraging the same datasets as in the previous analysis, this case study illustrates the reusability and extensibility of the CloudPricingOps framework. Here, the focus shifted from identifying key pricing attributes to categorizing services for better cost comparison, showcasing the framework’s adaptability in providing tailored decision-making tools. The ability to select and evaluate different methods, as demonstrated with the choice of K-Means, underscores the robustness of the framework in addressing various analytical needs across cloud service evaluation contexts.

6. Conclusions

Cloud FinOps is increasingly essential for businesses looking to maximize their cloud investments and maintain a competitive edge. Moreover, cloud decision-making frameworks play a crucial role in managing cloud resources effectively, allowing organizations to make informed choices that improve operational efficiency, reduce costs, and deliver strategic value. Therefore, we introduce a decision-making framework aligned with FinOps principles, called CloudPricingOps. The CloudPricingOps framework is built to be flexible, enabling the integration of various decision support methods customized for specific problem domains, such as price policy analysis, price prediction, and cost comparison.

It addresses the challenges of cloud service pricing by providing a unified solution that integrates well-established decision-making methods. It enables stakeholders to understand pricing policies, compare services effectively, and anticipate future pricing trends.

To demonstrate the flexibility and adaptability of CloudPricingOps, we present two case studies that showcase the framework’s ability to generate targeted insights for various decision-making scenarios. By using the same dataset in both cases, we emphasize the framework’s reusability and its capacity to deliver consistent and reliable results through different analytical approaches.

In the first case study, we highlight the CloudPricingOps framework’s ability to analyze the impact of cloud service attributes on pricing through a systematic and repeatable approach. The aim is to provide stakeholders with actionable insights into the cost-driving features, allowing for comparisons between IaaS, PaaS, and CaaS service types.

Additionally, the second case study illustrates the CloudPricingOps framework’s ability to categorize similar cloud service bundles, supporting effective cost comparison and decision-making. By clustering services with comparable performance attributes, the analysis offers stakeholders a clearer understanding of the cost–performance trade-offs, facilitating more informed decisions.

While the CloudPricingOps framework offers significant advantages, it must acknowledge several limitations. The framework's performance heavily relies on the quality and accuracy of the dataset used. Any errors or omissions in the data could affect the reliability of the generated insights. Also, the results can be changed by the fact that some methods have flaws, like how regression oversimplifies relationships or how K-Means relies on predefined cluster numbers. The framework's ability to generalize across diverse use cases and evolving service models requires further testing and adaptation. Additionally, the dynamic nature of cloud pricing necessitates frequent updates to the dataset and analytical models. Lastly, future iterations must address the computational resource requirements for large-scale analyses and the potential expertise gap for users with limited experience in decision-making methods.

Future research avenues may include real-time data analysis, extending the framework to address emerging service models such as edge computing, and integrating advanced machine learning algorithms for improved predictive accuracy. Enhanced user interactions could also be a feature in future iterations of the tool, potentially offering automated recommendations based on user-defined criteria.

The adoption of The Unifying Format for Cloud Billing Data within the Cloud Cloud-PricingOps framework marks a significant step forward in cloud financial management. This initiative aims to unify disparate billing data from various cloud service providers into a single format, simplifying analysis and reporting processes. Finally, the Cloud Cloud-PricingOps framework could be an interesting tool for Cloud Service Brokerage (CSB) [32]. Major players like Accenture, IBM, Wipro, Capgemini, and DXC Technology streamline cloud adoption and enhance interoperability [33]. The CSB model includes integration, vendor management, and governance services that assist businesses in navigating complex multi-cloud environments. The Cloud CloudPricingOpstool can be highly beneficial for a cloud broker, as it enable efficient management and optimization of cloud costs.

Author Contributions: Conceptualization, M.N.; Methodology, G.F. and M.N.; Validation, A.T., G.K. and C.M.; Investigation, G.F. and E.F.; Resources, E.F.; Writing—original draft, G.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Storment, J.; Fuller, M. *Cloud FinOps*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2023.
2. Sambatur, S. Cloud Fin Ops Management. *Int. J. Soft Comput. Eng.* **2024**, *13*, 7–9. <https://doi.org/10.35940/ijscce.a3585.13060124>.
3. Marketandmarket. Cloud FinOps Market Size, Share, Growth Analysis, By Offering (Solutions (Native, Third-Party), Services), Application (Cost Management and Optimization, Cost Allocation and Chargeback), Service Model, and Deployment Model—Global Industry Forecast to 2029. Available online: <https://tinyurl.com/bdhbrzya> (accessed on 8 October 2024).
4. Microsoft. Kick Start your FinOps Efforts. Available online: <https://microsoft.github.io/finops-toolkit/> (accessed on 28 October 2024).
5. Amazon. Why Cloud Financial Management with AWS? Available online: <https://aws.amazon.com/aws-cost-management/> (accessed on 28 October 2024).
6. CloudKeeper. CloudKeeper. Available online: <https://www.cloudkeeper.com/> (accessed on 28 October 2024).
7. DoiT. DoiT. Available online: <https://www.doit.com/> (accessed on 28 October 2024).
8. Amazon. Amazon Web Service. Available online: <https://tinyurl.com/mtzvpvhn> (accessed on 28 October 2024).
9. Microsoft. Pay as You Go with Azure. Available online: <https://azure.microsoft.com/en-in/pricing/offers/ms-azr-0003p> (accessed on 28 October 2024).
10. Chung, P. *AWS FinOps Simplified: Eliminate Cloud Waste Through Practical FinOps*; Packt Publishing: Birmingham, UK, 2022.

11. Soni, M. *FinOps Handbook for Microsoft Azure: Empowering Teams to Optimize Their Azure Cloud Spend with FinOps Best Practices*; Packt Publishing: Birmingham, UK, 2023.
12. Kanumuri, V.S. 3. A Strategic Roadmap to Scaling Cloud FinOps Teams: The Vision, Mission, Charter, and Placement. *J. Econ. Manag. Res.* **2023**, *4*, 1–35. [https://doi.org/10.47363/jesmr/2023\(4\)232](https://doi.org/10.47363/jesmr/2023(4)232).
13. Mileski, D.; Gusev, M. FinOps in Cloud-Native Near Real-Time Serverless Streaming Solutions. In Proceedings of the 2023 31st Telecommunications Forum (TELFOR), Belgrade, Serbia, 21–22 November 2023. <https://doi.org/10.1109/telfor59449.2023.10372626>.
14. Sustainable Cloud Computing: Leveraging FinOps for Environmental Responsibility. In *Design of Single Chip Microcomputer Control System for Stepping Motor*. In Proceedings of the 023 31st Telecommunications Forum (TELFOR), Belgrade, Serbia, 21–22 November 2023. [https://doi.org/10.47363/jaicc/2023\(2\)293](https://doi.org/10.47363/jaicc/2023(2)293).
15. Nawrocki, P.; Smendowski, M. FinOps-driven optimization of cloud resource usage for high-performance computing using machine learning. *J. Comput. Sci.* **2024**, *79*, 102292.
16. Zeleny, M. *Multiple Criteria Decision Making Kyoto 1975*; Springer Science & Business Media: New York, NY, USA, 2012; Volume 123.
17. Baker, J. The technology–organization–environment framework *Information Systems Theory: Explaining and Predicting Our Digital Society*; Springer: Berlin/Heidelberg, 2012; Volume 1, pp. 231–245.
18. Yoo, S.K.; Kim, B.Y. A decision-making model for adopting a cloud computing system. *Sustainability* **2018**, *10*, 2952.
19. Sharma, M.; Gupta, R.; Acharya, P. Prioritizing the critical factors of cloud computing adoption using multi-criteria decision-making techniques. *Glob. Bus. Rev.* **2020**, *21*, 142–161.
20. Bhol, S.G.; Mohanty, S.; Pattnaik, P.K. Machine Learning as a Service Cloud Selection: An MCDM Approach for Optimal Decision Making. *Procedia Comput. Sci.* **2024**, *233*, 909–918.
21. Menzel, M.; Ranjan, R. CloudGenius: Decision support for web server cloud migration. In Proceedings of the 21st international conference on World Wide Web, Lyon, France, 16–20 April 2012; pp. 979–988.
22. Mateen, A.; Nam, S.Y.; Haider, M.A.; Hanan, A. A Dynamic Decision Support System for Selection of Cloud Storage Provider. *Appl. Sci.* **2021**, *11*, 11296. <https://doi.org/10.3390/app112311296>.
23. Khajeh-Hosseini, A.; Greenwood, D.; Smith, J.W.; Sommerville, I. The cloud adoption toolkit: Supporting cloud adoption decisions in the enterprise. *Softw. Pract. Exp.* **2012**, *42*, 447–465.
24. Juan-Verdejo, A.; Zschaler, S.; Surajbali, B.; Baars, H.; Kemper, H.G. InCLOUDer: A formalised decision support modelling approach to migrate applications to cloud environments. In Proceedings of the 2014 40th EUROMICRO Conference on Software Engineering and Advanced Applications, Verona, Italy, 27–29 August 2014; IEEE: New York, NY, USA, 2014; pp. 467–474.
25. Gupta, S.; Munteş-Mulero, V.; Matthews, P.; Dominiak, J.; Omerovic, A.; Aranda, J.; Seycek, S. Risk-driven framework for decision support in cloud service selection. In Proceedings of the 2015 15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, Shenzhen, China, 4–7 May 2015; IEEE: New York, NY, USA, 2015; pp. 545–554.
26. Hussain, W.; Sohaib, O.; Naderpour, M.; Gao, H. Cloud marginal resource allocation: A decision support model. *Mob. Netw. Appl.* **2020**, *25*, 1418–1433.
27. Yin, H.; Aryani, A.; Petrie, S.; Nambissan, A.; Astudillo, A.; Cao, S. A Rapid Review of Clustering Algorithms. *arXiv* **2024**, arXiv:2401.07389.
28. Botchkarev, A. Performance metrics (error measures) in machine learning regression, forecasting and prognostics: Properties and typology. *arXiv* **2018**, arXiv:1809.03006.
29. Naidu, G.; Zuva, T.; Sibanda, E.M. A review of evaluation metrics in machine learning algorithms. In *Proceedings of the Computer Science On-line Conference*; Springer: New York, NY, USA, 2023; pp. 15–25.
30. Liagkou, V.; Fragiadakis, G.; Filiopoulou, E.; Michalakelis, C.; Tsadimas, A.; Nikolaidou, M. Assessing the Complexity of Cloud Pricing Policies: A Comparative Market Analysis. *J. Grid Comput.* **2024**, *22*, 65.
31. Fragiadakis, G.; Liagkou, V.; Filiopoulou, E.; Fragkakis, D.; Michalakelis, C.; Nikolaidou, M. Cloud services cost comparison: A clustering analysis framework. *Computing* **2023**, *105*, 2061–2088.
32. Filiopoulou, E.; Chatzithanasis, G.; Michalakelis, C.; Nikolaidou, M. Cloud Broker: Customizing Services for Cloud Market Requirements. *Information* **2024**, *15*, 232.
33. MarketsAndMarkets. Cloud Service Brokerage Market Size and Forecast. 2024. Available online: <https://www.marketsandmarkets.com/Market-Reports/cloud-brokerage-market-771.html> (accessed 11 October 2024).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.