Transactions on Computer Science and Intelligent Systems Research
ISSN: 2960-1800, eISSN: 2960-2238 | Volume 9
AIDML 2025

# The Investigation on the Influence of Dimension Manipulation on Regret Performance Using Upper Confidence Bound Algorithm

Wentao Qian *

Department of Statistics, Capital University of Economics and Business, Beijing, China

* Corresponding Author Email: xiao930@126.com

**Abstract.** This paper explores the impact of dimensionality manipulation on recommendation algorithm performance amid the $5 trillion global e-commerce landscape, where traditional methods suffer from the curse of dimensionality-feature redundancy eroding efficiency and distorting recommendations. Focusing on the "exploration-exploitation" balance, we use Principal Component Analysis (PCA) for dimensionality reduction, combined with the Upper Confidence Bound (UCB) algorithm to quantify regret performance differences. Using an Amazon product dataset, PCA reduces 15 features to 7 principal components, retaining core variance while mitigating redundancy. Feature crossing generates interaction features (e.g., price-star rating products) to enrich the feature space. Experimental results show dimensionality reduction boosts computational efficiency but risks losing feature semantics, slightly degrading UCB's exploitation accuracy. Dimensionality elevation, though increasing short-term exploration costs, enhances long-term performance by preserving critical feature correlations, outperforming reduced dimensions. This study highlights the trade-offs in dimensionality manipulation: reduction alleviates complexity but may distort information, while elevation enhances expressiveness at the cost of exploration. The findings offer a new approach to address dimensionality challenges in e-commerce recommendation systems, emphasizing the need to balance feature complexity and semantic integrity.

**Keywords:** Upper Confidence Bound, Principal Component Analysis, Dimension Manipulation.

## 1. Introduction

Imagine a consumer navigating a massive e-commerce platform, confronted with hundreds or thousands of products while considering factors such as price, appearance, and sales volume. The challenge of making a choice amid such complexity highlights the critical need for recommendation systems. Against the backdrop of a global e-commerce market exceeding $5 trillion (Statista data, 2024), recommendation algorithms have emerged as a core competitive advantage for e-commerce platforms to enhance user experience and commercial value. Leading platforms like Amazon, eBay, and Alibaba employ sophisticated algorithms; however, traditional approaches often fall prey to the curse of dimensionality [1]. Feature redundancy reduces computational efficiency, while diluted key information introduces recommendation bias.

Previous research on e-commerce recommendation algorithms primarily focuses on proposing new models or modifying existing ones to achieve better personalized recommendations. Yet most studies remain confined to algorithmic models themselves, neglecting the fundamental impact of feature space dimensionality manipulation on the exploration-exploitation mechanism. For example, Shuai Zhang pointed out that traditional methods "fail to analyze the influence of transmembrane feature combinations," thereby not addressing how dimensionality changes in feature space affect the essence of exploration and exploitation [2]. He et al. noted that conventional approaches struggle to capture complex data relationships [3]. Other works emphasize integrating external contextual data with recommendation systems to generate user-based recommendations, but few investigate dimensionality manipulation based on the native features of datasets. This paper introduces a novel path to address the curse of dimensionality by applying Principal Component Analysis (PCA) [4] for dimensionality reduction and augmentation, combined with the Upper Confidence Bound (UCB) algorithm to quantify regret performance differences across different dimensions.

This study focuses on feature dimensionality manipulation: reducing feature quantity via PCA-based dimensionality reduction and generating new features through feature crossing for dimensionality augmentation. By comparing the performance of the UCB algorithm on original, reduced-dimensional, and augmented-dimensional datasets, the research aims to reveal how feature dimensionality changes impact the exploration-exploitation balance in recommendation algorithms [5].

## 2. Method

### 2.1. Data preprocessing

This study utilizes the publicly available "*Amazon Products Dataset 2024*" from the Kaggle platform, where the original dataset has a dimension of 44696×17, containing 17 features and 44,696 product records. In the data preprocessing stage, feature selection is first performed to remove two features irrelevant to the core logic of recommendation algorithms-ASIN and product titles—leaving 15 features for in-depth processing.

For boolean features *isBestSeller* and *boughtInLastMonth*, the text values "*True*" and "*False*" are mapped to numerical values 1 and 0 respectively, achieving binary conversion to adapt to quantitative analysis models [6]. Price-related features price and *listPrice* first undergo removal of the currency symbol "*$*" and then conversion to floating-point data (e.g., "*$39.99*" is processed as 39.99), eliminating dimensional discrepancies that could interfere with subsequent calculations.

During data cleaning, missing values in numerical features such as stars and reviews are imputed using median values to mitigate the impact of abnormal data. For the categorical feature *category_id*, if the missing value proportion is below 5%, corresponding records are directly deleted to maintain the integrity of categorical data. Meanwhile, duplicate samples are removed by detecting the unique product identifier ASIN, ensuring the uniqueness and accuracy of the dataset.

To facilitate subsequent dimensionality manipulation, the textual categorical feature *category_id* undergoes one-hot encoding. Assuming it contains 20 unique categories, encoding generates 20 binary features that are merged with the remaining 6 numerical features. To focus on high-interaction products, product records with user review counts reviews>50 are filtered, resulting in a final dataset with N samples. At this point, all features have completed numerical conversion, and missing/duplicate values have been cleaned, providing standardized input for the subsequent application of Principal Component Analysis (PCA) and the Upper Confidence Bound (UCB) algorithm [7].

### 2.2. Core Formulas and Algorithmic

The UCB algorithm is used to address the Multi - armed Bandit (MAB) problem. By balancing "Exploration" and "Exploitation", it selects the optimal product (arm). The core formula is UCB:

$$UCB_i(t) = \bar{R}_i(t-1) + c\sqrt{\frac{ln\,(t)}{n_i(t-1)}} \tag{1}$$

Where $\bar{R}_i(t-1)$ the historical average is reward of the $i$-th product, is the number of selections up to the $(t-1)$-th round, $c$ is the exploration coefficient, and $t$ is the current total number of rounds.

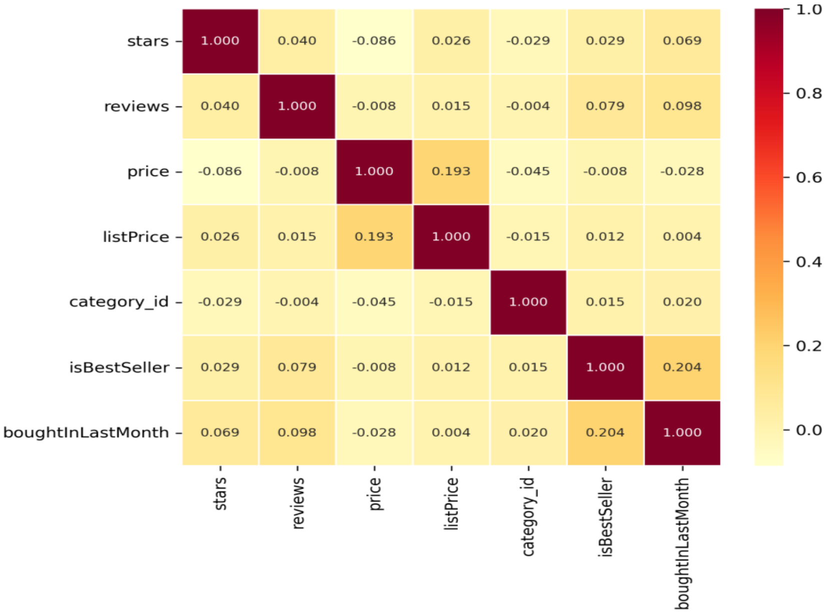
In the data pre - processing stage, features (including the one - hot - encoded *category_id* feature) are first standardized using $Z$ - score to ensure consistent dimensions. Subsequently, a covariance matrix $A$ is constructed. By solving the eigenvalue equation $|A - \lambda I| = 0$ and the eigenvector equation $AX = \lambda X$, eigenvalues and eigenvectors $(X)$ sorted by importance are obtained. When constructing principal components, the standardized feature vectors are projected into the eigenvector space. The expression for the $i$-th principal component is:

$$PC_i = \sum_{j=1}^{p} w_{ij} x_j \tag{2}$$

Where $w_{ij}$ is the eigenvector weight and $x_j$ is the standardized original feature value. Finally, retain the first 7 principal components, forming a 7-dimensional feature matrix after dimensionality reduction. This not only retains the core variation information of the original data but also significantly reduces the dimensionality of the feature space, providing support for the efficient operation of the subsequent UCB algorithm.

### 2.3. Dimensionality Enhancement Algorithm Design

Dimensionality enhancement generates new features through feature crossing to increase dimensionality. Based on correlation analysis (as shown in Fig. 1), the price feature, which has the highest correlation with the core reward feature (stars), is selected for crossing to generate a "price - star" interaction feature. This approach retains the key trends of the original features while providing richer decision - making clues for the algorithm.



| | stars | reviews | price | listPrice | category_id | isBestSeller | boughtInLastMonth |
|---|---|---|---|---|---|---|---|
| stars | 1.000 | 0.040 | -0.086 | 0.026 | -0.029 | 0.029 | 0.069 |
| reviews | 0.040 | 1.000 | -0.008 | 0.015 | -0.004 | 0.079 | 0.098 |
| price | -0.086 | -0.008 | 1.000 | 0.193 | -0.045 | -0.008 | -0.028 |
| listPrice | 0.026 | 0.015 | 0.193 | 1.000 | -0.015 | 0.012 | 0.004 |
| category_id | -0.029 | -0.004 | -0.045 | -0.015 | 1.000 | 0.015 | 0.020 |
| isBestSeller | 0.029 | 0.079 | -0.008 | 0.012 | 0.015 | 1.000 | 0.204 |
| boughtInLastMonth | 0.069 | 0.098 | -0.028 | 0.004 | 0.020 | 0.204 | 1.000 |

**Fig. 1** Heat map of correlation between each variable.

## 3. Results and Discussion

To quantify the performance differences of the Upper Confidence Bound (UCB) algorithm on the original dataset, dimensionality-reduced dataset, and dimensionality-augmented dataset, this study uses Cumulative Regret as the evaluation metric [8]. Its mathematical definition is:

$$R = E\left[\sum_{t=1}^{T} \left(r_{a^*,t} - r_{a,t}\right)\right] \tag{1}$$

Where $r_{a^*,t}$ represents the reward value of the optimal product at the $t$-th round, and $r_{a,t}$ is the reward value of the product actually selected by the algorithm. Cumulative regret reflects the cumulative deviation between the algorithm's selection strategy and the theoretically optimal strategy-smaller values indicate a more significant optimization effect of the algorithm on the exploration-exploitation balance.

### 3.1. Model Performance

Fig. 2 illustrates the trend of cumulative reward changing with time steps for the UCB algorithm on the Amazon dataset. It can be clearly observed that cumulative reward shows a continuous upward trend as time steps increase. This result indicates that the UCB algorithm can effectively explore and exploit product features with high reward potential in the data, gradually accumulating rewards.

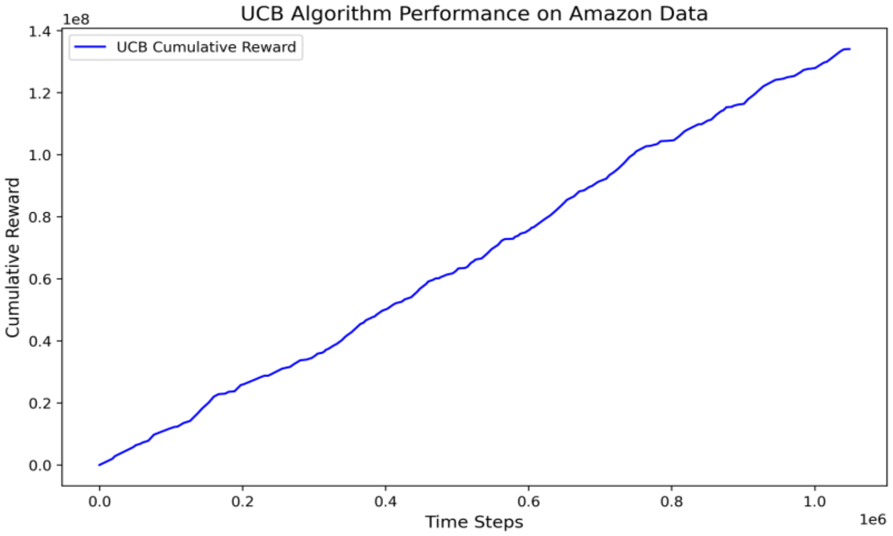


**Fig. 2** UCB Algorithm Performance on Amazon Data.

A further analysis of the curve's growth characteristics reveals that although cumulative reward consistently increases, its growth rate undergoes dynamic changes. In the initial stage, as the algorithm is in the exploration phase and has limited knowledge of each product's reward characteristics, cumulative reward grows relatively slowly. As time steps progress, the algorithm gradually identifies products or feature combinations with higher rewards through continuous trials and learning, and the growth rate of cumulative reward accelerates. This trend of slow initial growth followed by faster acceleration fully demonstrates the effectiveness of the UCB algorithm in balancing "exploration" and "exploitation"-exploring to acquire information in the early stage and efficiently utilizing historical information to maximize cumulative rewards in the later stage.

This performance holds significant implications for Amazon's product recommendation scenarios. In practical applications, as the number of user interactions increases (i.e., as time steps extend), the UCB algorithm continuously optimizes its recommendation strategy, preferentially selecting products that yield high rewards (such as those with high star ratings and high purchase conversion rates). This not only enhances the overall revenue of the recommendation system but also provides users with more valuable recommendations and creates higher commercial value for the platform.

### 3.2. Mechanism Analysis

Dimensionality reduction, while enhancing computational efficiency, may lead to the loss of feature physical meaning. Although the first 7 principal components retained in this study capture over 85% of the variance information in the original data, some principal components generated through linear combinations no longer possess clear business meanings (e.g., the independent interpretability of original features such as user ratings and prices is weakened). This weakening results in a decline in the algorithm's estimation accuracy of product rewards, thereby affecting "exploitation" efficiency.

Dimensionality augmentation provides richer decision-making clues for the algorithm by introducing feature interaction information. This study selects the price feature-with the highest correlation to the core reward feature (star ratings)-for crossing, generating new features (such as price-star interaction terms) that retain the key trends of original features. These new features help the algorithm more accurately identify high-reward products. Although the exploration demand for new features increases short-term regret, the long-term information gain from feature interactions compensates for the exploration cost, making the performance of the dimensionality-augmented dataset superior to that of the dimensionality-reduced dataset.

### 3.3. Research Implications and Future Directions

This study reveals the bidirectional impact of feature dimensionality manipulation on the regret performance of the UCB algorithm: dimensionality reduction alleviates the curse of dimensionality but may cause information distortion, while dimensionality augmentation enhances feature expression but increases exploration costs. Empirical results show that conducting dimensionality manipulation under the premise of preserving core feature semantics can more effectively optimize the "exploration-exploitation" balance of recommendation algorithms.

Future research can be expanded in the following directions: first, explore the impact of different dimensionality reduction precisions (such as retaining more principal components) and dimensionality augmentation strategies (such as multi-feature crossing combinations) on regret performance, and construct a quantitative evaluation framework for dimensionality manipulation; second, integrate deep learning techniques to automatically extract nonlinear feature representations after dimensionality reduction, avoiding semantic loss caused by traditional PCA-based dimensionality reduction; third, design a dynamic dimensionality adjustment mechanism that enables the algorithm to optimize the feature space in real time based on user interaction data, achieving a dynamic balance between computational efficiency and recommendation accuracy.

## 4. Conclusion

This study constructs three types of feature spaces-original, dimensionality-reduced, and dimensionality-augmented-by using principal component analysis and feature crossing techniques, and quantifies the impact of dimensionality manipulation on the regret performance of e-commerce recommendation systems based on the UCB algorithm. Key findings include: on the original dataset, the cumulative regret of the UCB algorithm exhibits the theoretically expected logarithmic growth; while dimensionality reduction improves computational efficiency, the loss of feature semantics leads to a slight decline in regret performance; dimensionality augmentation enhances information expression through feature interactions, outperforming the dimensionality-reduced dataset in performance.

The research results provide a new perspective for feature engineering in e-commerce recommendation systems: during dimensionality manipulation, it is necessary to balance the complexity of the feature space and semantic integrity, avoiding information loss caused by excessive dimensionality reduction or a surge in exploration costs from blind dimensionality augmentation. In the future, combining cutting-edge machine learning methods can further explore more efficient dimensionality manipulation strategies, providing theoretical support and practical paths for solving the curse of dimensionality in recommendation systems.

## References

[1] Deshpande M, Karypis G. Item-based top-n recommendation algorithms. ACM Transactions on Information Systems (TOIS), 2004, 22(1): 143-177.

[2] Zhang S, Yao L, Sun A, et al. Deep learning-based recommender system: A survey and new perspectives. ACM computing surveys (CSUR), 2019, 52(1): 1-38.

[3] He X, Liao L, Zhang H, et al. Neural collaborative filtering. Proceedings of the 26th international conference on World Wide Web. 2017: 173-182.

[4] Babamoradi H, van den Berg F, Rinnan Å. Bootstrap based confidence limits in principal component analysis—a case study. Chemometrics and Intelligent Laboratory Systems, 2013, 120: 97-105.

[5] Saito K, Notsu A, Ubukata S, et al. Performance Investigation of UCB Policy in Q-learning. 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA). IEEE, 2015: 777-780.

[6] Hall M A. Correlation-based feature selection of discrete and numeric class machine learning. 2000.

[7] Wang Q, Zeng C, Zhou W, et al. Online interactive collaborative filtering using multi-armed bandit with dependent arms. IEEE Transactions on Knowledge and Data Engineering, 2018, 31(8): 1569-1580.

[8] Combes R, Jiang C, Srikant R. Bandits with budgets: Regret lower bounds and optimal algorithms. ACM SIGMETRICS Performance Evaluation Review, 2015, 43(1): 245-257.