Enhancing Retail Sales Forecasting Using Exponential Factorization Machines: A Data-Driven Approach to Demand Prediction

Hira Ajmal
ajmalhira.ajmal@gmail.com

***Abstract*—Accurate sales forecasting is essential for retailers to streamline inventory management, minimize losses, and maximize profitability. This paper introduces a novel Exponential Factorization Machine (EFM) model designed specifically for predicting retail sales, particularly for new stock-keeping units (SKUs) with extended lead times and short life cycles. Unlike conventional forecasting techniques, EFM leverages product attributes, marketing dynamics, and store-level interactions to enhance predictive performance. The model integrates percentage error minimization (PES) and adaptive batch gradient descent (ABGD) to improve accuracy while reducing risks associated with overstocking and stockouts. Using real-world data from a footwear retailer in Singapore, the results demonstrate that EFM surpasses existing models in terms of Mean Absolute Percentage Error (MAPE) and Mean Absolute Error (MAE). This study presents a data-driven forecasting approach, equipping retailers with actionable insights for better decision-making in rapidly evolving market environments.**

I. Introduction

Accurate sales forecasting plays a crucial role in inventory management, procurement planning, and revenue optimization. Traditional models, including assortment planning [1], marketing analytics [2], and revenue management [3], rely on historical sales data, making them ineffective for newly introduced products [4].

New Stock Keeping Units (SKUs) introduce forecasting challenges due to uncertain consumer demand, long production cycles, and potential stock imbalances [5]. Proxy-based approaches, such as POS data comparisons and expert judgment, often fail to capture intricate attribute-based variations in sales patterns. Existing techniques, including multinomial logit (MNL) [2] and regression trees [3], struggle to model nonlinear attribute interactions effectively.

This study presents an Exponential Factorization Machine (EFM) framework for predicting sales by integrating structured product attributes with transactional data. The research is conducted in collaboration with a global retailer specializing in ladies' footwear, where frequent style turnovers create high forecasting uncertainty.

Footwear attributes vary across multiple dimensions, encompassing physical characteristics (e.g., size, heel height), latent factors (e.g., brand influence), and market-driven aspects (e.g., discounts, pricing). These variables, consisting of categorical and numerical data, require specialized modeling techniques. Figure 1 illustrates the classification of heel height variations among SKUs.

Fig. 1: Heel Height Categorization Across SKUs

The forecasting model estimates sales over an SKU's eleven-week launch period, six months before release, using a structured prediction pipeline shown in Figure 2.

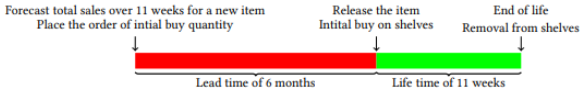


Fig. 2: Sales Forecasting Workflow for New SKUs

Sales estimation is formulated as a supervised regression problem, where target sales volumes depend on a combination of categorical and continuous product attributes. Instead of traditional linear models, a Factorization Machine (FM) with an exponential transformation is employed to improve generalization [6]. The model is optimized using an Adaptive Batch Gradient Descent (ABGD) algorithm, minimizing the following modified loss functions:

$$L_{ES} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{1}$$

$$L_{PES} = \sum_{i=1}^{n} \left(\frac{\hat{y}_i}{y_i} - 1 \right)^2 \tag{2}$$

where y_i represents actual sales and $\hat{y}_i$ denotes predicted values. The PES loss prioritizes relative accuracy, making it suitable when overstocking penalties exceed understocking risks, as in fast fashion and perishable goods.

A. Contributions

This work enhances sales forecasting by:

- Developing an EFM-based model that captures complex attribute interactions, improving SKU-level demand prediction.
- Reducing computational overhead by selectively modeling critical attribute interactions, improving efficiency.

- Evaluating ES vs. PES loss functions, demonstrating conditions where PES achieves superior forecasting accuracy.
- Introducing a data-driven feature selection method using a greedy optimization approach, eliminating manual attribute curation.

By addressing limitations in conventional demand forecasting, this approach provides a robust, scalable method for predicting sales under high uncertainty, aiding strategic inventory and procurement decisions.

II. Literature Review

The Exponential Factorization Machine (EFM) extends the Factorization Machine (FM) framework, originally formulated to capture high-order variable interactions efficiently [7]. Unlike traditional methods such as Support Vector Machines (SVMs) and logistic regression, FMs reduce the complexity of polynomial interactions through latent factorization, making them particularly effective in sparse datasets [8]. This characteristic has positioned FMs as a core technique in domains including personalized recommendations, financial forecasting, and predictive analytics [9]. However, most FM implementations primarily model second-order interactions due to computational constraints [10], which can limit their expressiveness in sales prediction tasks.

Factorization-based models have been successfully extended into deep learning paradigms, notably Deep Factorization Machines (DeepFM) for click-through rate (CTR) estimation [11]. Given the structural parallels between recommendation engines and retail sales forecasting, FM-based approaches provide a strong foundation for SKU-level demand prediction [12]. Since retail sales data predominantly consist of count values, an exponential transformation is applied within the EFM framework to better capture demand patterns.

A. Optimization Techniques

Conventional FM training utilizes Stochastic Gradient Descent (SGD) [7], Alternating Least Squares (ALS) [13], and Bayesian Inference methods [14]. While SGD is computationally efficient, it requires careful tuning of step sizes. ALS, though effective for closed-form solutions, struggles with large-scale categorical data. Markov Chain Monte Carlo (MCMC) methods offer probabilistic optimization but introduce convergence challenges in non-Gaussian settings.

For EFM training, this study employs an Adaptive Batch Gradient Descent (ABGD) method, which dynamically scales learning rates across training epochs. This approach ensures robust convergence over large-scale datasets, processing approximately 5,000 SKU records per cycle while mitigating gradient explosion and stagnation issues. The proposed loss function is formulated as:

$$L_{Exp} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 e^{-\alpha|y_i - \hat{y}_i|} \tag{3}$$

where α controls sensitivity to extreme errors, ensuring stable updates.

B. Feature Selection Strategies

Predictive modeling in sales forecasting necessitates an optimal balance between feature complexity and generalization [15]. Product attributes span multiple dimensions, including physical features (e.g., material, color), market-driven factors (e.g., pricing, discounting), and latent characteristics (e.g., brand reputation, manufacturer effects). While prior studies primarily emphasize dominant attributes like size and price, a broader scope incorporating interdependent features is essential.

Given that exhaustive feature selection is computationally intractable [16], a Greedy Forward Attribute Selection (GFAS) algorithm is introduced, integrating k-fold cross-validation to iteratively refine feature subsets. This technique ensures optimal attribute selection while reducing overfitting risks in high-dimensional SKU data.

C. Evaluation Metrics

Model performance is assessed using widely recognized error metrics, including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE) [17]. While RMSE penalizes large deviations more heavily than MAE, MAPE normalizes errors across varying SKU scales, enhancing comparability [18]. Industry benchmarks indicate an acceptable MAPE threshold of 77%, with consumer goods sectors averaging 76% [18].

Recognizing the asymmetric impact of over- and under-predictions, an alternative Penalty-Weighted Exponential Loss (PWEL) function is introduced:

$$L_{PWEL} = \sum_{i=1}^{n} w_i (y_i - \hat{y}_i)^2 e^{-\beta|y_i - \hat{y}_i|} \tag{4}$$

where w_i assigns greater penalties to overstocking errors, aligning cost-sensitive forecasting with inventory management priorities.

By addressing existing forecasting constraints, this study enhances predictive accuracy, contributing a scalable and adaptive approach to retail demand estimation.

III. Model Formulation

To estimate sales volume across Stock-Keeping Units (SKUs) in a retail setting, a structured interaction model is developed. The formulation introduces the following notations:

- $N = \{1, 2, \ldots, n\}$: Index set for SKUs.
- $M = \{1, 2, \ldots, m\}$: Set representing store locations.
- $A = \{1, \ldots, a\}$: Collection of categorical features (e.g., size, category, brand).
- $L_i = \{1, \ldots, l_i\}, i \in A$: Possible values for attribute i (e.g., "black" under color).
- $I = \{(c, c') | c \neq c', c, c' \in A\}$: Pairs of interacting attributes.
- y_{is}: Observed demand for SKU i at store s over the historical training window.
- $z_{is,cj}$: Binary encoding indicating whether SKU i at store s possesses level j for attribute c.

- $\tilde{y}_{is}$: Forecasted sales for SKU i at store s.

Each SKU associates with a single level per categorical feature, and absent values are imputed using synthetic categories.

IV. Parameter Estimation

The model requires estimation of the following parameters:

- γ_0: Global offset term.
- γ_{cj}: Weights assigned to categorical feature levels.
- $\langle \phi_{cj}, \phi_{c'j'} \rangle$: Latent interaction factors:

$$\langle \phi_{cj}, \phi_{c'j'} \rangle = \sum_{p=1}^{f} \phi_{cj,p} \phi_{c'j',p}. \tag{5}$$

The subset of attributes involved in pairwise relationships is formalized as:

$$A_I = \{c \mid \exists c' \in A \text{ where } (c, c') \in I \text{ or } (c', c) \in I\}. \tag{6}$$

To prevent excessive model complexity, a feature selection mechanism is integrated to enhance generalization.

V. Data Preparation and Evaluation

The dataset consists of instances $D = \{(z_{is}, y_{is}) \mid (i, s) \in S\}$, where z_{is} encodes SKU characteristics and y_{is} represents observed sales. Data points with stock shortages are excluded to maintain prediction reliability. The dataset is partitioned as follows:

- Training subset: $D_T = \{(z_{is}, y_{is}) \mid (i, s) \in T \subset S\}$.
- Test subset: $D_E = \{(z_{is}, y_{is}) \mid (i, s) \in E \subset S\}$.

The evaluation framework prioritizes model robustness and predictive accuracy across diverse retail scenarios.

VI. Model Training: Optimization Strategy

Model training is formulated as an unconstrained optimization problem, employing batch gradient descent with L2 regularization to mitigate overfitting risks.

A. Loss Functions

The model is optimized using two loss functions: Squared Error Loss (SEL) and Relative Squared Error (RSE). For $D_T = \{(z_{is}, y_{is}) \mid (i, s) \in T\}$ with predictions $\tilde{y}_{is}$, the loss functions are defined as:

- Squared Error Loss (SEL):

$$L_{SEL}(\Theta) = \frac{1}{2} \sum_{(i,s)\in T} (\tilde{y}_{is} - y_{is})^2. \tag{7}$$

- Relative Squared Error (RSE):

$$L_{RSE}(\Theta) = \frac{1}{2} \sum_{(i,s)\in T} \left(\frac{\tilde{y}_{is} - y_{is}}{y_{is}} \right)^2. \tag{8}$$

RSE imposes a stronger penalty for under-predicted values, aligning with inventory control considerations such as optimal stock replenishment strategies:

$$\text{Loss} = \text{Holding cost} \times (\tilde{y}_{is} - y_{is})^+ + \text{Shortage cost} \times (y_{is} - \tilde{y}_{is})^+. \tag{9}$$

Unlike conventional error functions, RSE remains differentiable, facilitating stable gradient updates during training.

Let Θ^*_{SEL} and Θ^*_{RSE} be the minimizers for SEL and RSE, respectively. Then,

$$L_{SEL}(\Theta^*_{SEL}) \leq L_{SEL}(\Theta^*_{RSE}) \leq \frac{y^2_{\max}}{y^2_{\min}} L_{SEL}(\Theta^*_{SEL}), \tag{10}$$

$$L_{RSE}(\Theta^*_{RSE}) \leq L_{RSE}(\Theta^*_{SEL}) \leq \frac{y^2_{\max}}{y^2_{\min}} L_{RSE}(\Theta^*_{RSE}). \tag{11}$$

Here, $y_{\max}$ and $y_{\min}$ denote the highest and lowest sales figures in D_T. The inequalities suggest that optimizing for one loss function does not necessarily yield the best performance under the other metric. If sales values exhibit uniform distribution, both loss functions produce identical optimal solutions; otherwise, significant variations may occur.

VII. Optimization Framework

The process of parameter estimation is approached as an optimization problem incorporating L2 regularization:

$$\Theta^* = \arg\min_{\Theta \in \mathbb{R}^K} J(\Theta) = L(\Theta) + \frac{1}{2} \sum_{\theta \in \Theta} \lambda_\theta \theta^2, \tag{12}$$

where $L(\Theta)$ represents the loss function, which can take different forms based on the error metric.

Case 1: ES-Based Optimization

$$\Theta^*_{ES} = \arg\min_{\Theta \in \mathbb{R}^K} \frac{1}{2} \sum_{(i,s)\in T} (\tilde{d}_{is} - d_{is})^2 + \frac{1}{2} \sum_{\theta \in \Theta} \lambda_\theta \theta^2. \tag{13}$$

Case 2: PES-Based Optimization

$$\Theta^*_{PES} = \arg\min_{\Theta \in \mathbb{R}^K} \frac{1}{2} \sum_{(i,s)\in T} \left(\frac{\tilde{d}_{is} - d_{is}}{d_{is}} \right)^2 + \frac{1}{2} \sum_{\theta \in \Theta} \lambda_\theta \theta^2. \tag{14}$$

VIII. Adaptive Batch Gradient Descent

The Adaptive Batch Gradient Descent (ABGD) algorithm iteratively updates parameters as follows:

$$\theta^{(t+1)} \leftarrow \theta^{(t)} - \eta \left[\frac{\partial L(\Theta)}{\partial \theta} \Big|_{\theta = \theta^{(t)}} + \lambda_\theta \theta^{(t)} \right], \tag{15}$$

where η denotes the learning rate.

Gradient Expressions:

ES Loss Gradient:

$$\frac{\partial L_{ES}(\Theta)}{\partial \theta} = \sum_{(i,s)\in T} (\tilde{d}_{is} - d_{is}) \tilde{d}_{is} g(\theta)(x_{is}). \tag{16}$$

PES Loss Gradient:

$$\frac{\partial L_{PES}(\Theta)}{\partial \theta} = \sum_{(i,s)\in T} (\tilde{d}_{is} - d_{is}) \frac{\tilde{d}_{is}}{d^2_{is}} g(\theta)(x_{is}). \tag{17}$$

Defining:

$$w_{is} = \begin{cases} \eta(\tilde{d}_{is} - d_{is})\tilde{d}_{is}, & L(\cdot) = L_{ES} \\ \eta(\tilde{d}_{is} - d_{is})\frac{\tilde{d}_{is}}{d^2_{is}}, & L(\cdot) = L_{PES} \end{cases}, \tag{18}$$

Which results in the parameter update rule:

$$\theta^{(t+1)} \leftarrow \theta^{(t)} - \sum_{(i,s)\in T} w_{is} g(\theta)(x_{is}) - \eta\lambda_\theta \theta^{(t)}. \tag{19}$$

The iterative training error is computed using:

$$TE_t = \begin{cases} \frac{1}{|T|}\sum_{(i,s)\in T} |\tilde{d}_{is}(\theta_t) - d_{is}|, & L(\cdot) = L_{ES} \\ \frac{1}{|T|}\sum_{(i,s)\in T} \left|\frac{\tilde{d}_{is}(\theta_t) - d_{is}}{d_{is}}\right|, & L(\cdot) = L_{PES} \end{cases}. \tag{20}$$

The learning rate is dynamically adjusted: if $TE_{t+1} < \epsilon$ and $TE_{t+1} > TE_t$, the step size is halved, with threshold values set as $\epsilon = 0.1$ for PES and 1.0 for ES.

IX. Model Training and Prediction

A. Feature Selection Strategy

Selecting the most informative features is essential for high-dimensional datasets to mitigate overfitting. The EFM framework considers individual attributes (A) and pairwise interactions (I) within the feature space. Given a total of 45 attributes, the number of possible interactions reaches $\binom{45}{2} = 990$, making a brute-force approach computationally prohibitive. To address this, a progressive selection methodology is implemented to determine the most effective feature set.

The Incremental Greedy Selection (IGS) algorithm iteratively expands the feature set by evaluating its influence on predictive accuracy. Since direct optimization on the test set is impractical, a k-fold cross-validation (CV) approach is employed to approximate validation performance and guide selection.

Algorithm 1 Incremental Greedy Selection (IGS)

Input: Training dataset D_T, candidate feature sets (A, I), loss function $l(\cdot)$ **Output:** Selected feature subsets (sA^*, sI^*) Split D_T into k CV folds: $\Lambda = \{T_i | i \in \{1, \ldots, k\}\}$ Initialize $(sA^*, sI^*) \leftarrow \emptyset$, direction flag $dir_1 \leftarrow 1$ Set feasibility flags $valid_{+1}, valid_{-1} \leftarrow$ True Initialize counter $itr \leftarrow 1$ Any search direction is valid Identify candidate additions $(\Delta A, \Delta I)$ using stepwise selection Update feature subsets: $currA_{itr} \leftarrow currA_{itr-1} \cup \Delta A$, $currI_{itr} \leftarrow currI_{itr-1} \cup \Delta I$ New features are included Compute validation error cvJ_{itr} Significant reduction in error ($p < 0.05$) Update $(sA^*, sI^*) \leftarrow (currA_{itr}, currI_{itr})$ Reset feasibility flags Mark direction as invalid Mark direction as invalid Adjust search direction: $dir_{itr+1} \leftarrow -dir_{itr}$ if valid Increment iteration: $itr \leftarrow itr + 1$ (sA^*, sI^*)

Initially, only a bias term is present, serving as the baseline for validation error. At each iteration, newly incorporated attributes and interactions are tested based on their effect on prediction performance. The search direction alternates unless deemed infeasible, ensuring a thorough evaluation of potential features.

To control complexity, a depth-first search method is applied while maintaining consistent cross-validation folds for validation. The significance of additional features is determined using a paired t-test ($p < 0.05$). The procedure halts when no further improvements are achievable.

B. Cross-Validation Procedure

The k-fold cross-validation process, described in Algorithm 3, divides the dataset into k partitions, where each fold alternates between validation (D_c^E) and training (D_c^T) sets. The ABGD method is then used to optimize the parameters based on the selected features, and validation performance cvJ_i is calculated according to the given loss function $l(\cdot)$.

For the ES loss function, validation performance is assessed using the mean squared error (MSE), whereas for PES loss, it is evaluated using the mean squared percentage error (MSPE):

$$cvJ_i = \begin{cases} \frac{1}{|T_i|}\sum\limits_{(j,s)\in T_i} (\hat{d}_{js} - d_{js})^2, & \text{if } l(\cdot) = l_{ES}(\cdot) \\ \frac{1}{|T_i|}\sum\limits_{(j,s)\in T_i} \left(\frac{\hat{d}_{js} - d_{js}}{d_{js}}\right)^2, & \text{if } l(\cdot) = l_{PES}(\cdot) \end{cases} \tag{21}$$

Typically, k is set to five for balancing bias and variance.

Algorithm 2 k-fold Cross-Validation

Feature sets csL, csI, partitions $\Lambda = \{T_i | i \in \{1, \ldots, k\}\}$, loss function $l(\cdot)$ Validation results $\{cvJ_i | i \in \{1, 2, \ldots, k\}\}$ **for** $i = 1$ to k **do** Define validation subset: $D_c^E \leftarrow \{(x_{js}, d_{js}) | (j,s) \in T_i\}$ Construct training subset: $D_c^T \leftarrow \bigcup_{g \neq i} \{(x_{js}, d_{js}) | (j,s) \in T_g\}$ Estimate parameters: $\Theta_c \leftarrow$ ABGD$(D_c^E, csL, csI, l(\cdot))$ Compute validation error using Eq. (1) Return $\{cvJ_i | i \in \{1, 2, \ldots, k\}\}$

C. Reference Model

The baseline reference model ignores attributes and interactions, relying solely on a global bias term β_0 for predictions:

$$\hat{d}_{is} = g_{\text{ref}}(x_{is}, \Theta) = \exp(\beta_0). \tag{22}$$

To determine the optimal β_0, the following function is minimized:

$$J_{\text{ref}}^{ES}(\beta_0) = \frac{1}{2}\sum_{(i,s)\in T} \left[\exp(\beta_0) - d_{is}\right]^2. \tag{23}$$

The optimal solution is derived analytically as:

$$\beta_{0,ES}^* = \log\left[\frac{\sum\limits_{(i,s)\in T} d_{is}}{|T|}\right]. \tag{24}$$

D. Progressive Attribute Selection

Progressive attribute selection (PAS), outlined in Algorithm 4, systematically incorporates features (ΓA) and their interactions (ΓI) to enhance predictive performance.

Algorithm 3 Progressive Attribute Selection (PAS)

Candidate feature sets fsA, fsI, training data D_T, direction parameter dp Optimized features ΓA, ΓI Initialize: $\Gamma A \leftarrow \emptyset$, $\Gamma I \leftarrow \emptyset$ Compute coefficients: $\Omega_{cf} \leftarrow \text{BGDA}(D_T, fsA, fsI)$ Estimate predicted values $\tilde{d}_{is}^{cf}$
if $dp = 1$ **then** Find attribute reducing error function Update ΓA $dp = -1$ Identify interaction minimizing prediction deviation Update ΓI Return $\Gamma A, \Gamma I$

E. Optimization Function $K_{PAS,c}$

$$K_{PAS,c}^{*} = \min_{\alpha_{cj} \in \mathbb{R}, j \in Q_c} K_{PAS,c}(\alpha_{cj}, j \in Q_c) \tag{25}$$

Which expands to:

$$\begin{cases} \sum\limits_{j \in Q_c} \sum\limits_{(i,s) \in U_{cj}} (\tilde{d}_{is}^{cf} v_{cj}^{FS} - d_{is})^2 + \kappa_{A,FS}|Q_c|, & \psi(\cdot) = \psi_{FS}(\cdot) \\ \sum\limits_{j \in Q_c} \sum\limits_{(i,s) \in U_{cj}} (z_{is} v_{cj}^{PFS} - 1)^2 + \kappa_{A,PFS}|Q_c|, & \psi(\cdot) = \psi_{PFS}(\cdot) \end{cases} \tag{26}$$

where $U_{cj} = \{(i,s) | x_{is}^{cj} = 1, (i,s) \in U\}$, and weights:

$$v_{cj}^{FS} = \frac{\sum\limits_{(i,s) \in U_{cj}} d_{is} \tilde{d}_{is}^{cf}}{\sum\limits_{(i,s) \in U_{cj}} (\tilde{d}_{is}^{cf})^2}, \quad v_{cj}^{PFS} = \frac{\sum\limits_{(i,s) \in U_{cj}} z_{is}}{\sum\limits_{(i,s) \in U_{cj}} z_{is}^2}, \quad z_{is} = \frac{\tilde{d}_{is}^{cf}}{d_{is}}. \tag{27}$$

F. Interaction Optimization Criterion

For effective interaction selection, defined:

$$R_{cc'} = \sum_{\substack{j \in Q_c, \\ j' \in Q_{c'}}} x_{is}^{cj} x_{is}^{c'j'} \sum_{p=1}^{h} \nu_{cj,p} \nu_{c'j',p} \tag{28}$$

where $\nu_{cj,p}$ and $\nu_{c'j',p}$ are estimated parameters, with $\kappa_{I,FS}$ and $\kappa_{I,PFS}$ governing interaction complexity.

The minimization process is given by:

$$K_{PIS,(c,c')}^{*} = \min_{\nu_{cj}, \nu_{c'j'} \in \mathbb{R}^h, j \in Q_c, j' \in Q_{c'}} K_{PIS,(c,c')}(\nu_{cj}, \nu_{c'j'}) \tag{29}$$

Which simplifies to:

$$K_{PIS,(c,c')} = \begin{cases} \sum\limits_{j \in Q_c} \sum\limits_{j' \in Q_{c'}} \sum\limits_{(i,s) \in U_{cj,c'j'}} \left(\tilde{d}_{is}^{cf} v_{(c,c')}^{FS} - d_{is} \right)^2 \\ \quad + \kappa_{I,FS}|Q_c||Q_{c'}|, & \text{if } \psi(\cdot) = \psi_{FS}(\cdot) \\ \sum\limits_{j \in Q_c} \sum\limits_{j' \in Q_{c'}} \sum\limits_{(i,s) \in U_{cj,c'j'}} \left(z_{is} v_{(c,c')}^{PFS} - 1 \right)^2 \\ \quad + \kappa_{I,PFS}|Q_c||Q_{c'}|, & \text{if } \psi(\cdot) = \psi_{PFS}(\cdot) \end{cases} \tag{30}$$

where $U_{cj,c'j'} = \{(i,s) | x_{is}^{cj} = 1, x_{is}^{c'j'} = 1, (i,s) \in U\}$.

G. Comprehensive Workflow

The complete framework, detailed in Algorithm 4, consists of training and validation phases.

During training, the Advanced Feature Selection and Optimization (AFSO) algorithm determines the best feature (pA^*) and interaction (pI^*) sets (Line 1). Bayesian Gradient Descent Approximation (BGDA) then estimates model parameters (Line 2).

Testing evaluates predictions $\{\tilde{d}_{is} | (i,s) \in V\}$ with error assessment using mean squared error (MSE) and root mean squared error (RMSE) at SKU-store and SKU-chain levels.

Algorithm 4 Comprehensive Workflow

Require: Training data D_T, Validation data D_V, Attributes A, Interactions I, Loss function $\psi(\cdot)$
Ensure: Predictions $\{\tilde{d}_{is} | (i,s) \in V\}$, MSE, RMSE at SKU-store and SKU-chain levels
1: Determine feature subsets: $(pA^*, pI^*) \leftarrow \text{AFSO}(D_T, \psi(\cdot), Q, I)$
2: Compute parameters: $\Omega^* \leftarrow \text{BGDA}(D_T, pA^*, pI^*, \psi(\cdot))$
3: **for** each $(i,s) \in V$ **do**
4: Compute $\tilde{d}_{is} = g(x_{is}; \Omega^*)$
5: **end for**
6: Evaluate MSE,
7: **return** Predictions and error metrics =0

H. Computational Experiments

This section details the computational assessment conducted on the sales forecasting dataset, supplemented with two publicly available datasets for benchmarking. Several adaptations of the EFM framework are examined to derive comparative insights. All models are implemented in Java and executed on distinct computing environments. Hyper-parameter optimization is performed using a Supermicro Linux server (4-core Intel Xeon E5-2623V3, 256GB RAM, Ubuntu 16.04), while additional simulations are conducted on a Dell desktop (Intel Core i7-6700 CPU, 32GB RAM, Windows 10 64-bit). The evaluated model variants include:

- **EFM-PES**: Incorporates PES-based optimization for loss minimization and feature refinement. Hyper-parameter tuning leverages grid search, leading to selected feature sets denoted as $\mathcal{A}_P^* ES$ and $\mathcal{I}_P^* ES$.
- **EFM-ES**: Employs ES-driven loss function alongside ES-centric feature selection, adhering to a similar tuning mechanism. The determined feature sets are denoted as $\mathcal{A}_E^* S$ and $\mathcal{I}_E^* S$.
- **logFM-PES**: Applies a logarithmic transformation to input data before training with PES-selected features, ensuring output predictions are exponentiated for scale recovery.
- **logFM-ES**: A variation of logFM-PES, utilizing the ES optimization framework for model learning.

I. Retail Sales Forecasting

1) Dataset Description and Preprocessing: The dataset originates from a Singapore-based women's footwear retailer,

integrating product descriptors with transactional records. A unified dataset is structured with 45 predictive features and a target variable. The preprocessing pipeline includes anomaly detection, handling of missing values, anonymization, aggregation, and discretization. Data transformation utilizes standard mining techniques, with sales consolidated over an 11-week launch window. Categorical attributes undergo discretization using the 'discretization' function from the 'infotheo' R package.

The dataset encompasses transactions spanning January 1, 2012, to July 20, 2014. Training data (D_T) covers January 1, 2012, to April 14, 2013, while test data (D_E) extends from December 31, 2013, to May 3, 2014. The segmentation strategy is illustrated in Figure 3, considering an 11-week product lifecycle and a six-month lead time. Table I outlines dataset sizes across anonymized product categories (69-y6, p-1, x-9w).

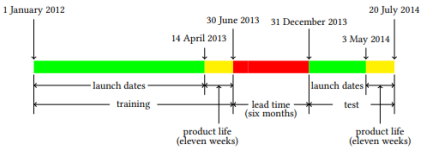


Fig. 3: Data partitioning into training and test sets.

Class	Training Set (D_T)	Test Set (D_E)
69-y6	4,984	1,179
p-1	6,222	1,824
x-9w	5,526	1,481

TABLE I: Size distribution of product class datasets.

2) Methodology and Model Enhancements: In addition to EFM-PES, EFM-ES, logFM-PES, and logFM-ES, extended model configurations are explored:

- **SP-EFM-PES**: Augments EFM with store-SKU interactions by leveraging both continuous and categorical relationships:

$$\hat{y}_{is} = g_E(x_{is};\Theta) = \exp\Bigg(\alpha_0 + \sum_{c\in\mathcal{A}}\sum_{j\in\mathcal{L}_c}\alpha_{cj}x_{is}^{cj} + \sum_{b\in\mathcal{B}}\alpha_b z_{is}^b + \sum_{(c,c')\in\mathcal{I}}\sum_{j\in\mathcal{L}_c}\sum_{j'\in\mathcal{L}_{c'}} x_{is}^{cj}x_{is}^{c'j'}\langle\nu_{cj},\nu_{c'j'}\rangle + \sum_{c\in\mathcal{A}}\sum_{j\in\mathcal{L}_c}\sum_{b\in\mathcal{B}} x_{is}^{cj}z_{is}^b\langle\omega_{cj},\omega_b\rangle + \sum_{b\in\mathcal{B}}\sum_{b'\in\mathcal{B},b'<b} z_{is}^b z_{is}^{b'}\langle\xi_b,\xi_{b'}\rangle\Bigg) \tag{31}$$

where SKU similarity in-store is estimated as:

$$\sum_{c\in\{\mathcal{A}_P^*ES\cup\mathcal{A}_{\mathcal{I}_P^*ES}\}}\sum_{j\in\mathcal{L}_c}\frac{\mathbb{1}(x_{is}^{cj}=x_{i's}^{cj})}{|\mathcal{A}_P^*ES\cup\mathcal{A}_{\mathcal{I}_P^*ES}|} \tag{32}$$

Top-matching SKUs over the 11-week period define the continuous feature set $\mathcal{B}$. The PES loss function remains consistent, with categorical interactions guiding feature refinement.

3) Predictive Modeling Approaches: To gain insights into sales trends, various forecasting models were explored:

- **SP-EFM-ES**: This model employs an ES-based loss function and modifies input variables along with their interactions to derive sA'_{ES} and sI'_{ES} in alignment with the EFM-ES framework.
- **Lasso Regression**: Implemented using the `glmnet()` function from the `glmnet` R package, this method efficiently performs internal feature selection across available predictors.
- **Random Forest (RF)**: Developed using `h2o.randForest()` from the `h2o` package, which autonomously determines significant attributes.
- **Regression Tree (RT)**: Constructed using `rpart()` from the `rpart` package, enabling decision tree-based learning with feature selection capabilities.
- **Support Vector Regression (SVR)**: Modeled using `svm()` from `e1071`, incorporating all variables initially before refining selections based on model optimization.

4) Evaluation Metrics: To assess model effectiveness, Mean Absolute Percentage Error (MAPE) and Mean Absolute Error (MAE) were employed, with results detailed in Table II.

TABLE II: Comparison of Sales Forecasting Models Using MAPE and MAE

	SKU-Store Level		SKU-Chain Level	
Model	MAPE	MAE	MAPE	MAE
EFM-PES	4.55%	1.59	4.57%	3.81
EFM-ES	5.15%	2.49	5.32%	3.61
logFM-PES	6.90%	3.25	6.50%	4.25
logFM-ES	7.20%	2.48	7.02%	3.91
SP-EFM-PES	6.50%	3.62	8.20%	5.06
SP-EFM-ES	7.23%	2.3	9.50%	4.05
Lasso	85.99%	28.01	86.40%	67.95
RF	116.43%	31.6	109.67%	76.65
RT	142.82%	39.03	135.05%	94.61
SVR	89.77%	24.56	96.40%	59.58

5) Analysis of Results: A review of sales forecasting for women's footwear (SKU: 69-y6, p-1, x-9w) indicates:

1) EFM-PES and EFM-ES demonstrate superior accuracy, achieving the lowest error rates across models tested.
2) Models relying on automated feature selection, such as Lasso, RF, RT, and SVR, exhibit relatively lower predictive performance.
3) logFM-PES and logFM-ES fail to outperform EFM-based methodologies, reinforcing the efficacy of the latter.

Comparisons with prior research confirm the effectiveness of the models, with SKU-chain level predictions attaining MAPE values near 5% and MAE below 7 units, improving on previous benchmarks like the 16.2% MAPE for new SKU forecasting.

J. Application to Student Performance and Fire Prediction

1) Dataset Characteristics and Preprocessing: Two publicly available datasets were analyzed to assess the applicability of EFM, logFM, PES, and ES methodologies:

- **Secondary School Student Performance (SCTP)**: This dataset, sourced from the UCI Machine Learning Repository[1] and Kaggle[2], contains academic, social, and demographic variables influencing student grades (G1, G2, G3) in Mathematics (395 instances) and Portuguese (649 instances). To ensure meaningful MAPE calculations, zero-valued G3 entries were substituted with 0.1.
- **Forest Fires (FF)** [19]: Retrieved from the UCI Machine Learning Repository[3] and Kaggle[4], this dataset includes 517 recorded fire incidents with meteorological parameters used to predict burned area. To facilitate MAPE computation, instances with zero burned area were replaced with 0.1.

TABLE III: Hyperparameter Settings for FM-Based Methods on SCTP and FF Datasets

Loss Function FF	Hyper-parameter	SCTP-P	SCTP-M
1.95×10^{-6}	Learning rate η	4.95×10^{-6}	3.5×10^{-6}
PES	maxInteractions	4000	4000
5.0×10^{-4}	λ_A, PES	0.005	1.0×10^{-3}
5.0×10^{-4}	λ_I, PES	0.10	1.0×10^{-3}
0.1	λ_v	1.0×10^{-3}	0.1
0	λ_w	10	0
0.1	Standard deviation σ	0.1	0.1
2.05×10^{-10}	Learning rate η	4.80×10^{-10}	3.15×10^{-10}
ES	maxInteractions	5000	10000
100	λ_A, ES	1000	100
500	λ_I, ES	1000	100
0	λ_v	100	0
0	λ_w	0	0
0.1	Standard deviation σ	0.1	0.1

K. Experimental Framework and Configuration

A comprehensive comparative analysis is conducted utilizing EFM-PES, EFM-ES, logFM-PES, logFM-ES, SVR, and RF models. The EFM-based strategies incorporate numerical attributes, with effectiveness gauged through MAPE and MAE. To ensure robust evaluation, a five-fold cross-validation strategy is employed, systematically dividing data into equal partitions and alternating the validation set per iteration. The reported outcomes denote the mean MAPE and MAE values. Hyperparameter tuning for SVR and RF is guided by findings from previous studies [19]. Feature selection in FM-based methodologies adheres to the principles outlined in the Section, emphasizing continuous predictors. A grid search approach is utilized for optimizing model parameters.

1) Predictive Performance Assessment: Table IV presents the forecast evaluation results. Key observations include: (1) FM-based models outperform SVR and RF in accuracy metrics; (2) EFM-PES and EFM-ES demonstrate superior predictive performance for the SCTP dataset, minimizing MAPE and MAE, respectively; (3) logFM-PES and logFM-ES show optimal results for the FF dataset in terms of MAPE and MAE.

TABLE IV: Prediction results: Mean MAPE and MAE over five evaluations for SCTP and FF datasets.

	Mean MAPE			Mean MAE		
	SCTP-P	SCTP-M	FF	SCTP-P	SCTP-M	FF
EFM-PES (proposed)	13.2%	11.05%	34.8%	2.00	2.95	16.85
EFM-ES (proposed)	14.10%	13.25%	36.9%	1.00	1.02	16.10
logFM-PES	15.05%	15.00%	33.9%	2.45	3.55	15.10
logFM-ES	16.30%	17.25%	35.1%	2.00	3.25	14.10
SVR	20.5%	11.2%	39.2%	2.65	3.40	16.60
RF	17.2%	11.2%	41.0%	1.85	2.00	17.00

X. Analysis and Interpretation

A. Effect of Logarithmic Transformation and Exponential Modelling

Both logarithmic transformation and exponential formulation contribute differently to response variable modeling. Empirical findings highlight that exponential formulation (EFM-PES, EFM-ES) achieves superior precision in sales prediction and SCTP datasets, whereas log-transformation (logFM-PES, logFM-ES) is preferable for the FF dataset. These disparities arise due to variations in response variable distributions, as shown in Table V. The logarithmic transformation is effective for normalizing highly skewed distributions, whereas exponential modeling benefits datasets with more uniformly distributed response values.

TABLE V: Empirical distribution of response variables across training and test datasets.

Response Variable	[Min, 0.25Max]	[0.25Max, 0.5Max]	[0.5Max, 0.75Max]	[0.75Max, Max]
Retail Sales in 69-y6	21.00%	28.80%	28.10%	22.10%
Retail Sales in p-1	27.95%	22.40%	23.00%	26.65%
Retail Sales in x-9w	23.30%	27.10%	26.55%	23.05%
Portuguese G3 Grade in SCTP	2.60%	13.10%	64.00%	20.30%
Math G3 Grade in SCTP	11.80%	35.10%	42.50%	10.60%
Burned Area in FF	99.40%	0.20%	0.20%	0.20%

B. Contrasting PES and ES Loss Optimization

A detailed evaluation of PES and ES loss minimization within EFM models reveals that EFM-PES consistently enhances MAPE performance, whereas EFM-ES exhibits lower MAE. The underlying distinction lies in how each loss function handles scaling relative to response values. Table VI

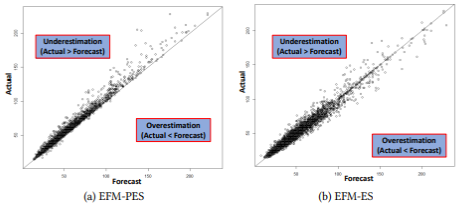
[1]https://archive.ics.uci.edu/ml/datasets/student+performance
[2]https://www.kaggle.com/dipam7/student-grade-prediction
[3]https://archive.ics.uci.edu/ml/datasets/Forest+Fires
[4]https://www.kaggle.com/elikplim/forest-fires-data-set

presents training metrics, confirming that EFM-PES predominantly favors smaller response values, leading to a slight underestimation trend, whereas EFM-ES yields a more neutral prediction bias.

TABLE VI: Training metrics associated with PES and ES loss functions.

Dataset	Method	MPES	Underestimation Ratio
69-y6	EFM-PES	5.5×10^{-5}	0.54
	EFM-ES	6.8×10^{-5}	0.41
p-1	EFM-PES	8.0×10^{-5}	0.62
	EFM-ES	9.5×10^{-5}	0.47
x-9w	EFM-PES	1.1×10^{-4}	0.60
	EFM-ES	3.7×10^{-5}	0.45

Figure 4 illustrates SKU-store sales against model forecasts, highlighting that EFM-PES tends to slightly underestimate while EFM-ES provides balanced estimations. These insights confirm that PES loss is advantageous when prioritizing smaller response values, whereas ES loss offers better generalization across a broader response range.



(a) EFM-PES (b) EFM-ES

Fig. 4: Comparison of SKU-store actual sales versus forecasts from EFM-PES and EFM-ES.

C. Adaptive Normalization for PES Optimization in Linear Regression

This section introduces a novel instance-based normalization technique designed specifically for Percentage Error Squared (PES) minimization within linear regression models. Drawing from pricing and demand forecasting methodologies, a linear relationship is defined as $d = \alpha_0 + \alpha_1 x$, constrained by $d \geq 0$ and a strictly decreasing function $d(x)$, enforcing $\alpha_1 < 0$. For illustration, α_0 is set to 1250 and α_1 to -12.

To evaluate performance, a synthetic dataset of 100 samples (x_i, d_i) is generated, where x_i is randomly drawn from a uniform distribution over $[1, 50]$. The dependent variable is formulated as $d_i = 1250 - 12x_i + \xi_i$, incorporating noise $\xi_i \sim \mathcal{N}(0, \tau)$. The variability in observations is controlled by the standard deviation τ.

It compare the performance of Exponential Squared (ES) and PES loss minimization for estimating parameters. The ES-based solution is computed via traditional least squares (LS), while PES minimization follows a new Adaptive Percentage Regression (APR) technique, defined as:

1) **Instance-Based Normalization:** Convert (x_i, d_i) into $(x_i/d_i, 1)$ prior to regression.
2) **Optimized Regression:** Apply LS to the transformed dataset to obtain stable estimates.

Unlike conventional column-wise normalization, this row-wise approach ensures uniform scaling across observations while preserving the predictor structure.

The effectiveness of APR is tested under different noise conditions: $\tau = 15$ and $\tau = 250$. Figure 5 illustrates that for low variance ($\tau = 15$) with $\frac{d^2_{\max}}{d^2_{\min}} = 3.2$, APR and LS yield comparable predictions. However, under high variance ($\tau = 250$) with $\frac{d^2_{\max}}{d^2_{\min}} = 58.5$, APR significantly diverges from LS, exhibiting a stronger underestimation bias.

To systematically evaluate this effect, 200 datasets are generated with increasing τ values from 1 to 250. The resulting Mean Exponential Squared Error (MESE), Mean Percentage Error Squared (MPES), underestimation ratio, and $\frac{d^2_{\max}}{d^2_{\min}}$ are analyzed. Figure 6 illustrates that APR's underestimation effect intensifies as $\frac{d^2_{\max}}{d^2_{\min}}$ grows, highlighting the need for a threshold-based correction strategy.

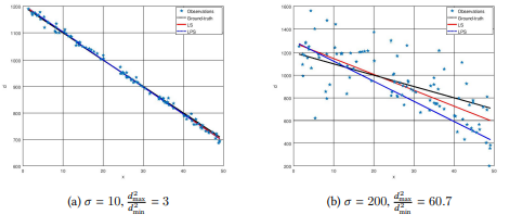

(a) $\sigma = 10$, $\frac{d^2_{\max}}{d^2_{\min}} = 3$ (b) $\sigma = 200$, $\frac{d^2_{\max}}{d^2_{\min}} = 60.7$

Fig. 5: Comparison of LS and APR performance: (a) $\tau = 15$, $\frac{d^2_{\max}}{d^2_{\min}} = 3.2$, (b) $\tau = 250$, $\frac{d^2_{\max}}{d^2_{\min}} = 58.5$.

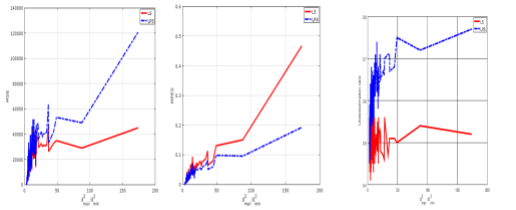

(a) MSE versus $\frac{d^2_{\max}}{d^2_{\min}}$ (b) MPSE versus $\frac{d^2_{\max}}{d^2_{\min}}$ (c) Underestimation ratio versus $\frac{d^2_{\max}}{d^2_{\min}}$

Fig. 6: Trends in MESE, MPES, and underestimation ratio for LS and APR with varying τ.

XI. Conclusions and Future Directions

This work investigates sales forecasting techniques for newly launched products characterized by long procurement cycles and short market lifespans. A primary challenge is the lack of historical data, necessitating reliance on predictive modeling. An Exponential Factorization Machine (EFM) framework is introduced, integrating product attributes and interaction effects. Key insights from this study include:

- Attribute-based factorization significantly improves sales forecasts for new products, addressing a limitation in traditional marketing models.

- PES minimization systematically underestimates predictions, making it suitable for applications where overstocking risks must be minimized, such as perishable goods management.
- The proposed row-wise normalization strategy for PES minimization provides an alternative to standard feature-wise scaling, improving estimation stability.
- Exponential transformation demonstrates superior performance for non-skewed positive response variables, contrasting with log-based approaches that perform better on right-skewed distributions.

Future research will focus on extending the EFM framework to incorporate substitution effects and developing probabilistic models to analyze the trade-offs between exponential and log-based transformations across diverse data distributions.

REFERENCES

[1] M. Fisher and A. Raman, *The New Science of Retailing: How Analytics are Transforming the Supply Chain and Improving Performance*. Harvard Business Press, 2013.

[2] P. S. Fader and B. G. S. Hardie, "Forecasting repeat sales at cdnow: A case study," *Interfaces*, vol. 28, no. 3, pp. 43–55, 1998.

[3] P. Ferreira and D. Simchi-Levi, "Assortment optimization in revenue management," *Manufacturing Service Operations Management*, vol. 14, no. 2, pp. 246–260, 2012.

[4] J. H. H. Frederick H. Abernathy, John T. Dunlop and D. Weil, *A Stitch in Time: Lean Retailing and the Transformation of Manufacturing—Lessons from the Apparel and Textile Industries*. Oxford University Press, 1999.

[5] F. Caro and J. M. Gallien, "Clearance pricing optimization for a fast-fashion retailer," *Operations Research*, vol. 60, no. 6, pp. 1404–1422, 2012.

[6] S. Rendle, "Factorization machines," in *Proceedings of the 10th IEEE International Conference on Data Mining (ICDM)*. IEEE, 2010, pp. 995–1000.

[7] ——, "Factorization machines," pp. 995–1000, 2010.

[8] A. F. Mathieu Blondel and N. Ueda, "Higher-order factorization machines," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 29, 2016.

[9] F.-l. C. Ting Chen, Zhaowen Wang and S.-C. Cheung, "Factorization machines for user preference prediction in e-commerce," in *Proceedings of the 22nd International Conference on Pattern Recognition (ICPR)*, 2012, pp. 1573–1578.

[10] W.-S. C. Yuchin Juan, Yong Zhuang and C.-J. Lin, "Field-aware factorization machines for ctr prediction," in *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys)*, 2016, pp. 43–50.

[11] Y. Y. Z. L. Huifeng Guo, Ruiming Tang and X. He, "Deepfm: A factorization-machine based neural network for ctr prediction," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, 2017, pp. 1725–1731.

[12] A. R. Marshall Fisher and A. Sheen, *Machine Learning for Retail Demand Forecasting*. Harvard Business Review Press, 2019.

[13] R. S. Yunhong Zhou, Dennis Wilkinson and R. Pan, "Large-scale parallel collaborative filtering for the netflix prize," in *Proceedings of the 4th International Conference on Algorithmic Aspects in Information and Management (AAIM)*, 2008, pp. 337–348.

[14] J. M. H. Prem Gopalan and D. M. Blei, "Scalable recommendation with hierarchical poisson factorization," *Proceedings of the 31st Conference on Uncertainty in Artificial Intelligence (UAI)*, 2015.

[15] J. R. H. Olivier Toubia and R. Garcia, "Probabilistic polyhedral methods for adaptive choice-based conjoint analysis," *Marketing Science*, vol. 26, no. 5, pp. 596–610, 2007.

[16] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[17] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International Journal of Forecasting*, vol. 22, no. 4, pp. 679–688, 2006.

[18] J. S. Armstrong, "Long-range forecasting: From crystal ball to computer," *John Wiley & Sons*, 1985.

[19] P. Cortez and A. de Jesus Raimundo Morais, "A data mining approach to predict forest fires using meteorological data," 2007.