Big Market outlet sales prediction (Regression Task).

Alamein International University Faculty of Computer Science & Neural Networks Engineering Course: AI231 – Neural Networks Final Project - Part 1

Group Members:

Basmala Elkady 23101372

Hany Ziad 23100109

Hoda Mahmoud 23101498

Jana Mamdouh 23101377

Mennatullah Mohamed 23102384

Juwariya Mohamed 23101379

Table of Contents

Problem Definition

This project involves Regression task to predict the outlet sales using a tabular dataset, with MSE 0.3090 , MAE 0.4287, RMSE 0.5559,R^2 0.7125.

Project Scope

We used Big Mart Sales Prediction Datasets from Kaggle, implementing the full workflow: data collection, preprocessing, model design, training, evaluation, and improvements with visualizations.

---

2. Data Collection:

The data is from [Big Mart Sales Prediction Datasets] it includes :

11 features describe the for 1559 products across 10 stores in different cities.

Float number represents the outlet sales of the item.

- **Training Set**: 60 % of the data.
- **Validation set**: 15 % of the data.
- **Testing Set**: 15% of the data.
- **Classes**: 11 ['Item_Weight', 'Item_Visibility', 'Item_MRP','Outlet_Establishment_Year','Item_Identifier_Encoded', 'Item_Fat_Content_Encoded', 'Item_Type_Encoded','Outlet_Identifier_Encoded', 'Outlet_Size_Encoded', 'Outlet_Location_Type_Encoded', 'Outlet_Type_Encoded']
- **Format**: CSV file.
- **Context**: state the challenge of outlet sales prediction and highlight the important of enhance data.

---

3. Data Preprocessing

Steps Applied

Data Cleaning and Imputation

A. Standardizing Inconsistent Categories

Typographical errors and aliases in categorical columns were unified to ensure proper feature mapping:

- Item_Fat_Content: Aliases like LF and low fat were mapped to Low Fat, and reg was mapped to Regular.
- Outlet_Size: Variations (e.g., MEdium, SMALL, HIGH) were mapped to the standard categories (Medium, Small, High).

B. Missing Values Imputation

- Item_Weight: Missing weights were imputed using the mean weight specific to each product (Item_Identifier). Any remaining NaNs were filled with the overall column mean.
- Outlet_Size: Missing store sizes were imputed using the mode (most frequent size) within each corresponding store type (Outlet_Type).

C. Handling Inconsistent Zero Values

- Item_Visibility: Zero values (0) were considered inconsistent, as a product is assumed to have some visibility if sold. These zero entries were replaced with the mean Item_Visibility value.

D. Handling Negative Values

- Item_Weight: Negative weights, which are physically impossible, were corrected using the abs() function, converting them to their absolute positive values.

3. Outliers Handling

Outliers in key numerical columns were identified using the Interquartile Range (IQR) method.

- Bounds: The lower ($Q1 - 1.5 \times IQR$) and upper ($Q3 + 1.5 \times IQR$) bounds were calculated for each relevant column.
- Correction (Clipping): The clipping technique was applied to Item_Weight and Item_Visibility. This method caps the extreme values at the calculated IQR bounds, mitigating their effect on model training while retaining the data points.

4. Feature Engineering and Transformation

A. Encoding

- Label Encoding was applied to all remaining categorical features (Item_Identifier, Outlet_Identifier, Item_Fat_Content, etc.). This process converts non-numeric labels into machine-readable numerical format.
- The original categorical columns were dropped after creating the new encoded versions (col + '_Encoded').

B. Scaling (Standardization)

- Standard Scaler was applied to all input features (X). This process transforms the features to have a mean of zero (0) and a standard deviation of one (1).
- Importance: This scaling is crucial for Neural Networks and gradient-based optimization, as it prevents features with larger ranges (e.g., Item_MRP) from disproportionately influencing the loss function during Gradient Descent.

5.Initial Exploratory Data Analysis (EDA)

Missing Values Check, Duplicates Check, Descriptive Statistics, Categorical Analysis.

---

5. Model Building:

1. Network Architecture:

- **L1 (Hidden)** 128 ReLU Extract basic features and patterns.
- **L2 (Hidden)** 64 ReLU Deepen feature representation.
- **L3 (Hidden)** 32 ReLU Reduce dimensionality before output.
- **L4 (Output)** 1 Linear Output a continuous value (Sales prediction).

- **Weight Initialization: HeInitialization** was used for the ReLU layers to ensure proper gradient flow and stable learning.

2. Loss Function and Regularization

A. Loss Function

The **Mean Squared Error (MSE)** function was used as the loss function, which is the standard measure for regression problems.

B. Regularization

L2 Regularization was applied to mitigate overfitting by adding a penalty term proportional to the square of the weight values to the loss function.

3. Hyperparameters Used

Learning Rate, Number of Epochs, Momentum Factor, L2 Lambda, Mini-Batch Size.

6. Training and Evaluation:

Training Steps

- **Optimizer:** momentum
- **Loss:** MSE
- **Metrics:** R^2
- **Epochs: 10000.**
- **Results:**
- **Test MSE (Optimized): 0.3022**
- **Test MAE (Optimized): 0.4216**
- **Test RMSE (Optimized): 0.5497**
- **Test R^2 Score (Optimized): 0.7189**

7. Model Improvements:

Optimization Algorithms

The training employed a sophisticated optimization scheme: Mini-Batch Gradient Descent combined with Momentum.

A. Mini-Batch Gradient Descent

The training set was split into small batches (Mini-Batches) of size 42. Weights were updated after processing each mini-batch.

- Benefit: This balances the speed of Stochastic Gradient Descent (SGD) with the stability of Batch Gradient Descent, leading to faster and smoother convergence.

B. Momentum

The Momentum algorithm was used to accelerate learning in consistent directions and dampen oscillations. The momentum vector v acts as an exponentially weighted moving average of past gradients.

8. Observations and Conclusions:

Model performance increase when we use momentum, L2 and mini batch.

Model Evaluation on Test Set (Optimized)

Test MSE (Optimized): 0.3022

Test MAE (Optimized): 0.4216

Test RMSE (Optimized): 0.5497

Test R^2 Score (Optimized): 0.7189