DATA SCIENCE COURSEWORK REPORT

Yasin Lester M00809695, Mohamed Salah M00913589, Arfanul Shouid Begum M00915474, Abdinasir Omar M00862122

Word Count
4615

1.Introduction

The video game industry is a dynamic and rapidly evolving entertainment sector. With increasing availability of game sales data, machine learning offers valuable tools for understanding market behaviour and forecasting future trends. This project leverages structured sales data and a Decision Tree Regressor to predict global video game sales using various features such as platform, genre, publisher, and regional sales.

The procedure followed a complete data science workflow: data preprocessing, exploratory analysis, feature engineering, model training, evaluation, and reflection on ethical issues. The focus is on deriving business-relevant insights that can aid decision-making in game development and marketing.

1.2.Dataset Description & Justification

This experiment uses a cleaned version of Kaggle's publicly accessible Video Game Sales Dataset, which contains over 16,000 video game entries. Each record contains information such as platform, genre, publisher, release year, and sales in four regions North America, Europe, Japan, and Others as well as a final field for global sales totals. The dataset's organised nature and comprehensive feature representation make it ideal for supervised regression modelling, providing important insight into industry patterns across decades. Its open-source license and lack of personal information also assured complete ethical compliance.

The Decision Tree Regressor was chosen as the final model because of its flexibility, high accuracy, and ability to manage both category and numerical data without requiring extensive preprocessing. It outperformed other models examined, such as Linear Regression and k-Nearest Neighbours, by identifying complicated trends in the data and controlling the volatility caused by blockbuster title games. With a R^2 of 0.8616 and RMSE of 0.6667, it was the most successful and understandable method for projecting worldwide video game sales.

1.3.Data Cleaning & Preprocessing

Missing values in the categorical columns were filled with "N/A" to make sure the dataset remained complete. To prepare the data for modelling, label encoding was used for the Genre column to support visualisations, while one-hot encoding was used for other categorical columns like Platform and Publisher that had many unique values. The numerical columns were scaled using Min-Max normalisation so that all values would fall within the same range and contribute equally to the model.

1.4.Exploratory Data Analysis (EDA)

During the EDA phase, a comprehensive study of the dataset's structure, patterns, and potential biases was conducted. Figure 1 clearly shows that Nintendo products such as Wii Sports and Super Mario dominate the rankings, with the top ten best-selling games worldwide. This pattern demonstrates not just the worldwide popularity of Nintendo franchises, but also the bundling approach used with consoles such as the Wii, which boosts sales numbers for certain games over their natural purchase volume.

Figure 2 displays the historical history of worldwide game sales, with a large growth between 2008 and 2009. This peak correlates with the general popularity of seventh-generation consoles, particularly the Nintendo Wii and PlayStation 3, which were critical in extending the gaming business to casual users and contributing to record sales years.

Figure 3 depicts regional sales distribution using boxplots or histograms, highlighting major variances across markets. The sales distribution implies more volatility and longer tails in North America and Europe, implying a higher prevalence of blockbuster titles in these regions. In comparison, Japan has a more steady and narrower distribution, with fewer titles that sell particularly well or poorly.

A regression line plot comparing North American and worldwide sales as shown in Figure 4 reveals a strong positive linear relationship. Data point density along the line suggests that North American performance is usually a good indicator of a game's success worldwide.

Though not entirely linear, the strength of the correlation justifies adopting NA Sales as a reliable and important input feature for predicting global sales.

Average sales per game across areas shown in Figure 5. Though its total worldwide numbers are lower, Japan surprisingly has the greatest average per title, per game. This indicates a market where less games attain moderate to great popularity, reflecting cultural tastes or a concentration on fewer, higher-quality local releases.

Highlighting Action, Sports, and Shooter games as the most commercially dominant, Figure 6 totals worldwide sales by genre. But this only reveals part of the picture.
Revealing notable variation and skewness, Figure 7 uses a boxplot to display the internal distribution of sales within each genre. Some genres, such Action and Shooter, include outliers that skew the mean, implying that a small number of very high-selling games strongly affect these genres.

Figure 8 contrasts the sales distribution across areas using Kernel Density Estimation. While NA and EU display broader curves with smoother peaks, suggesting more varied results, the KDE curves indicate that JP sales tend to group around lower numbers. This understanding supports the reason the model has better predictive power from NA and EU sales figures.

Figure 9 shows a bar chart or pie chart of regional contributions to global sales. Nearly half of all worldwide video game sales originate from North America. This disparity highlights one of the main factors influencing prediction effectiveness in the dataset and justifies weighting regional characteristics appropriately.

A pair plot of feature interactions, as seen in Figure 10, facilitates the visualisation of linear and non-linear correlations. The grouping and alignment in the NA and EU plots in relation to GlobalSales clearly support the model's emphasis on these features. In contrast, JP and OtherSales reveal more erratic correlations and less alignment.

A correlation heatmap in Figure 11 completes the investigation by mathematically measuring these links. GlobalSales (0.94) has the greatest Pearson connection with NA Sales, followed by EU Sales (0.90). With a lower correlation of 0.61, JP Sales supports the choice to give NA and EU top priority in model input. This also verifies the results of Figures 4 and 10.

These visualisations taken together not only show which areas and genres propel worldwide video game success but also expose the data structure regarding skew, variation, and market bias. Feature selection and data preparation for equitable, efficient modelling were much guided by the EDA stage.

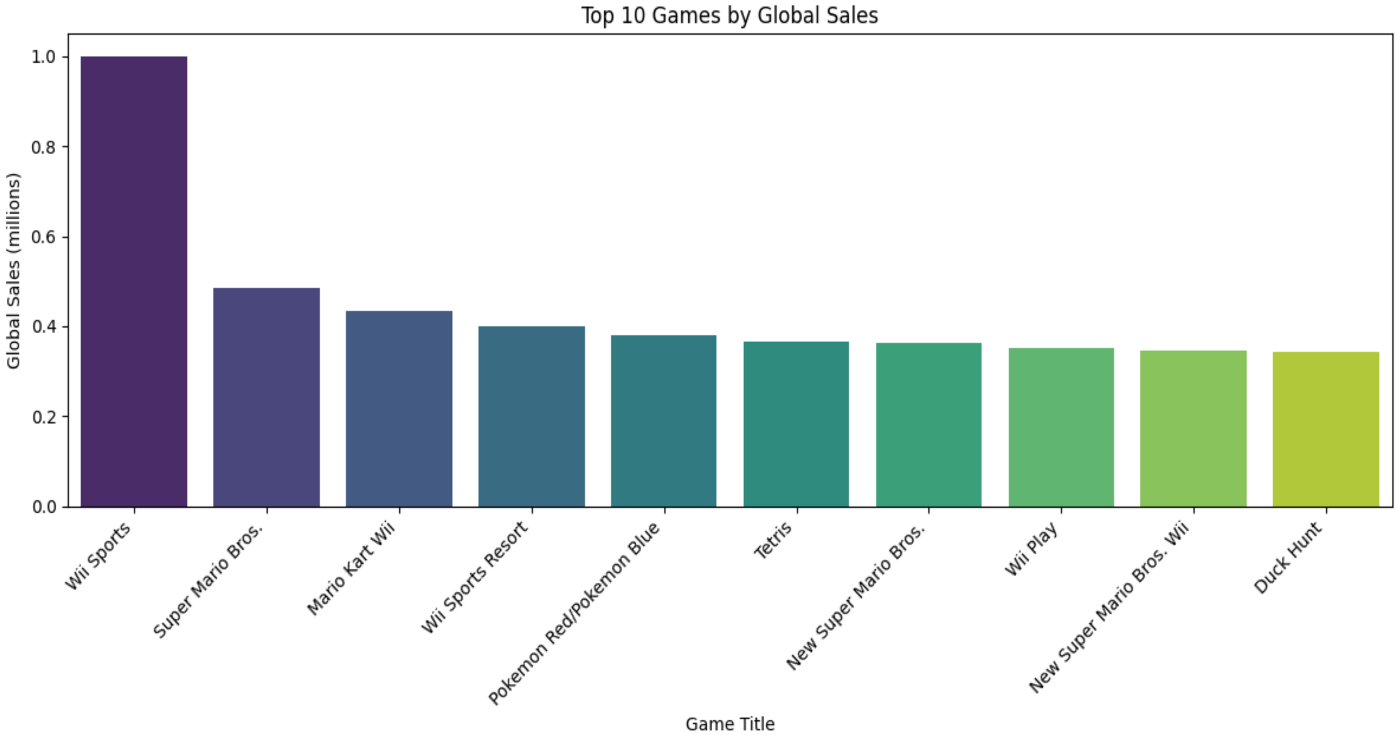


Figure 1:
Top 10 best-selling games globally, with Nintendo titles dominating overall sales.

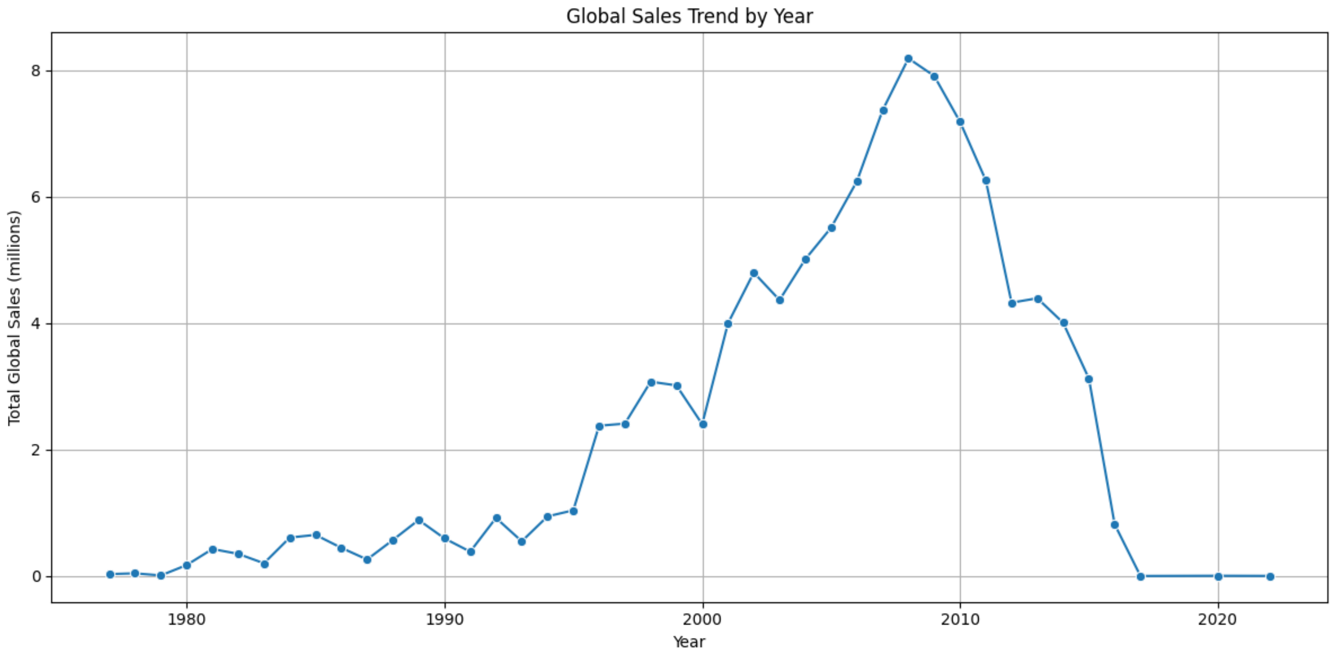


Figure 2:
Global video game sales trend over time, peaking around 2008–2009.

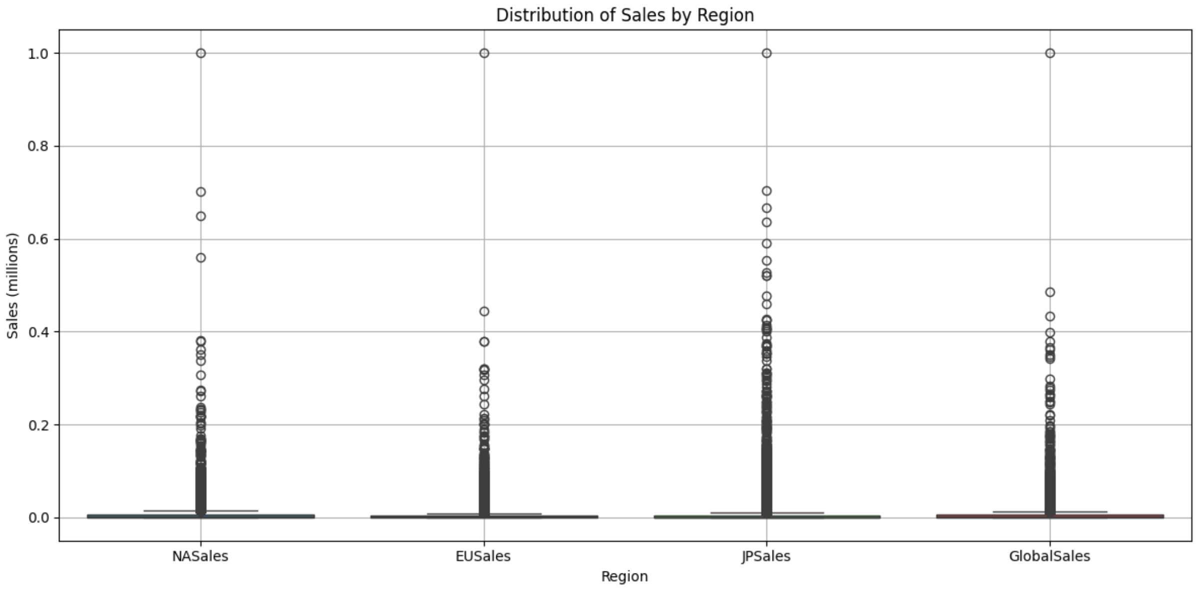


Figure 3:
Distribution of sales by region; NA and EU show greater variation than JP.

Figure 4:
Positive linear relationship between NA sales and global sales.

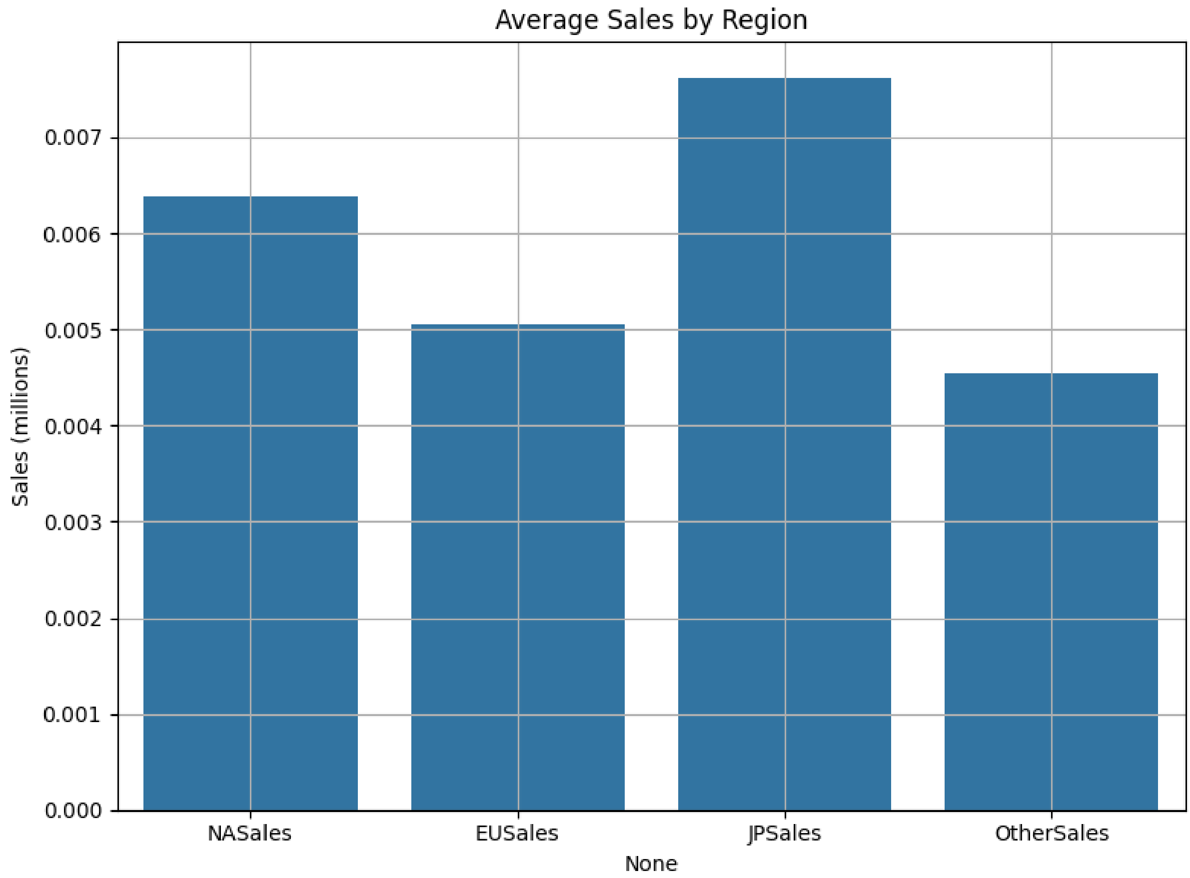


Figure 5:
Average game sales by region; JP has the highest per-game average.

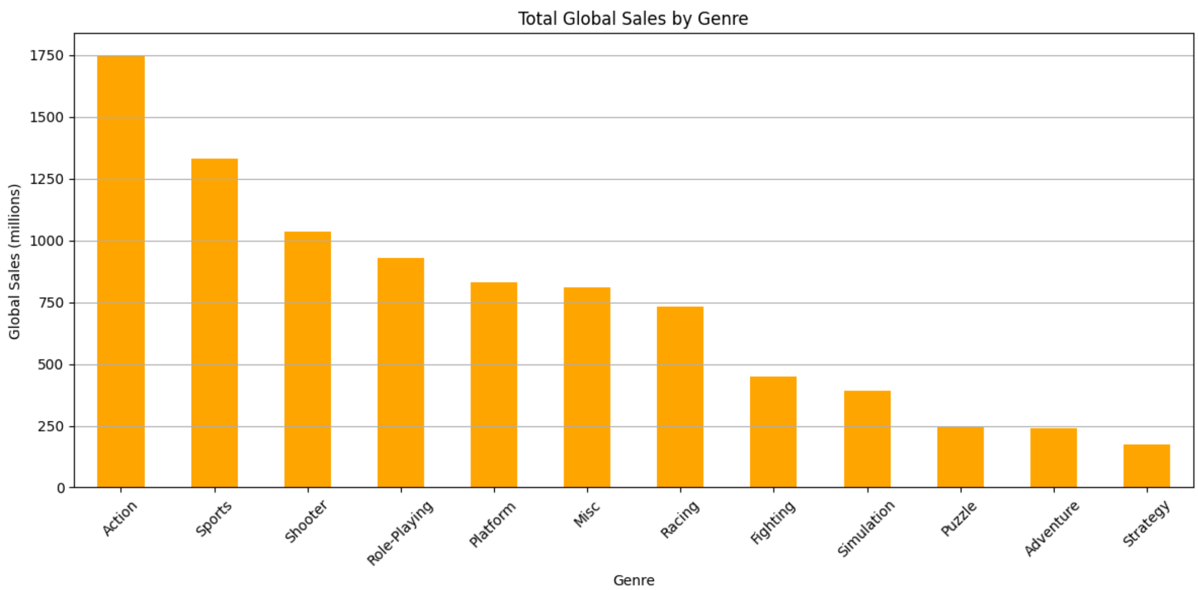


Figure 6:
Total global sales by genre, with Action, Sports, and Shooter leading.

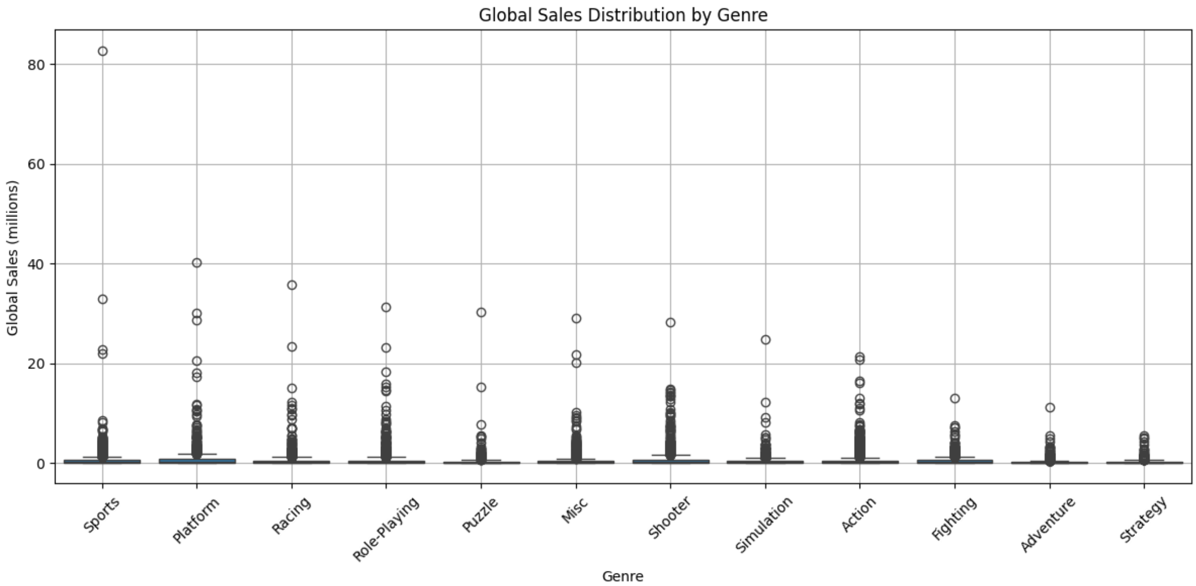


Figure 7:
Sales distribution by genre; some genres show high variability.

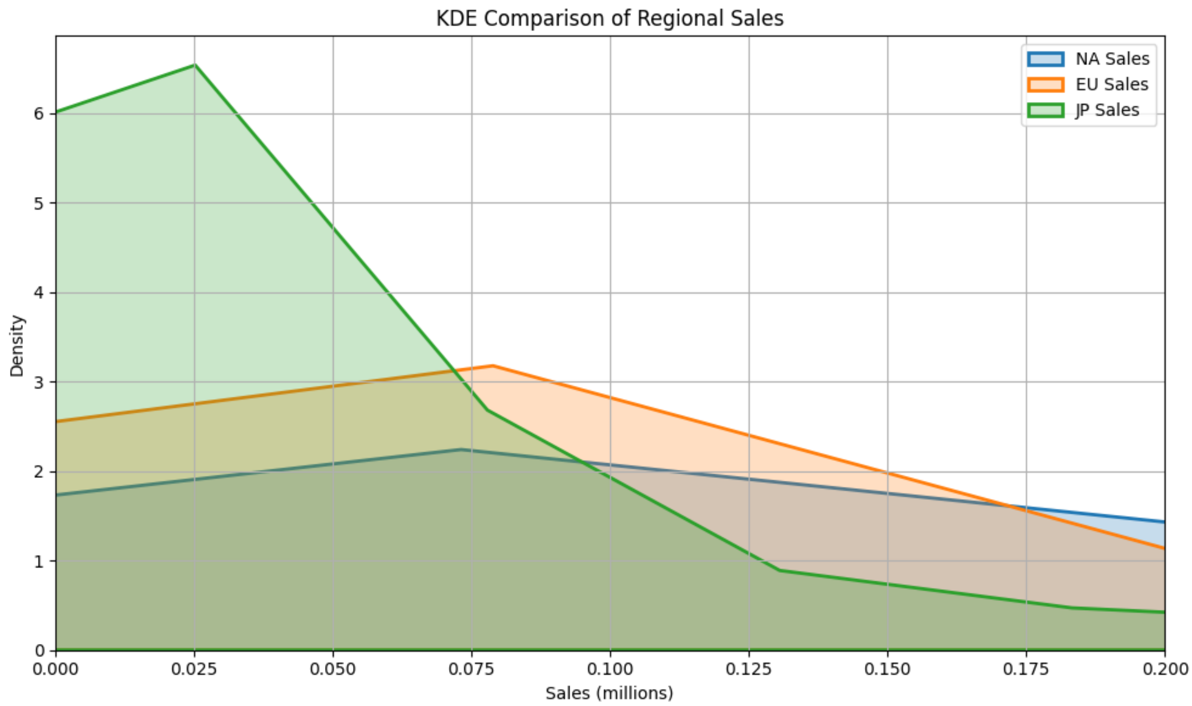


Figure 8:
Density plot comparing sales distribution across NA, EU, and JP.

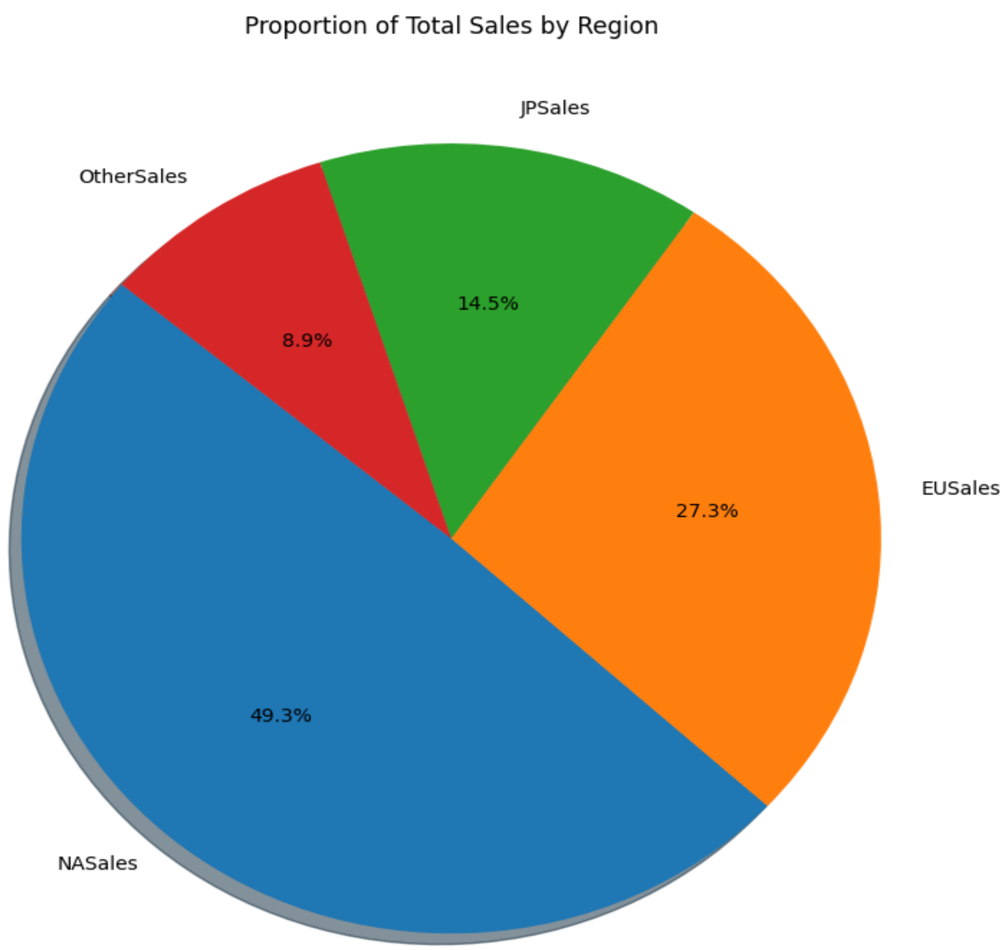


Figure 9:
Regional share of global sales; NA contributes the largest portion.

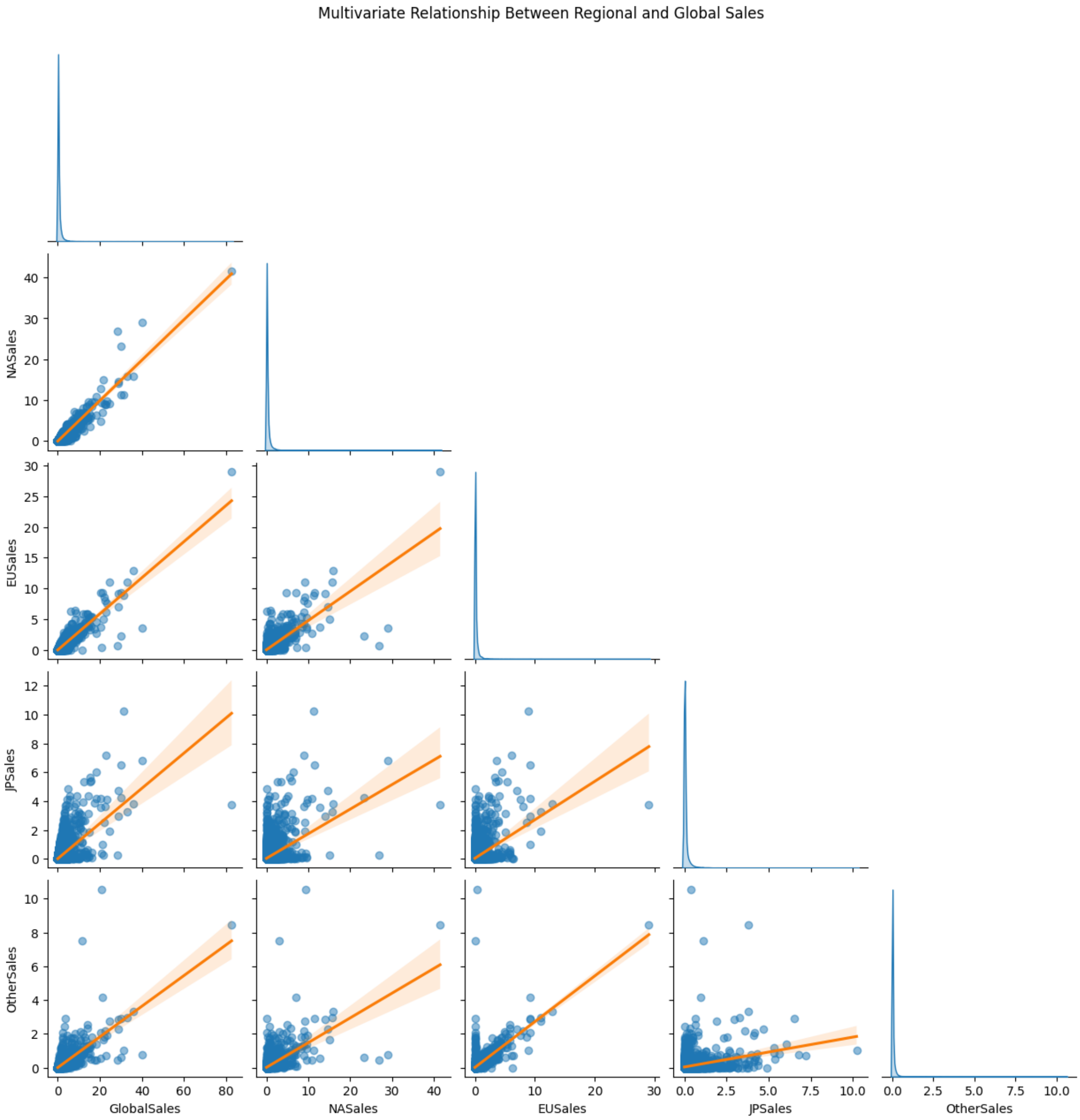


Figure 10:
Pair plot showing strong relationships between NA/EU and global sales.

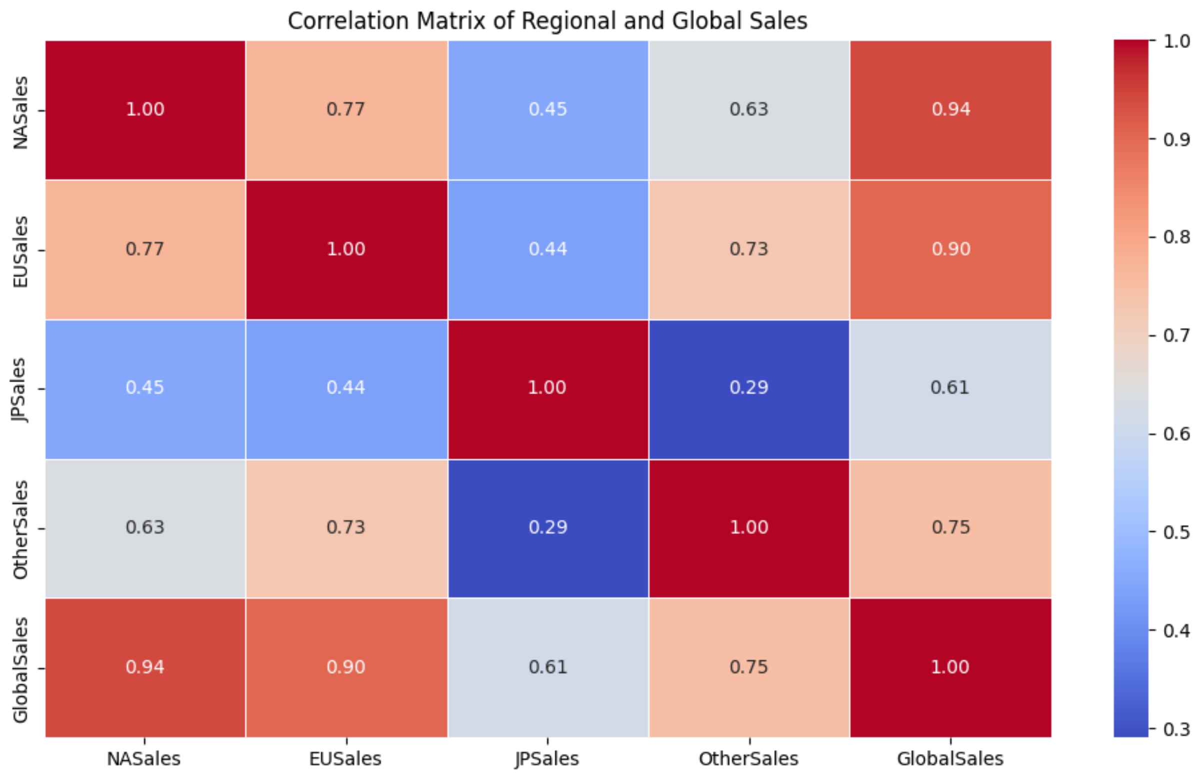


Figure 11:
Correlation heatmap; NASales shows the strongest link to GlobalSales.

1.5.Problem Definition

The basic difficulty of this study is predicting a continuous numerical goal specifically, worldwide video game sales which qualifies as a supervised regression problem. The aim was to predict a game's GlobalSales by using past sales data and metadata characteristics, hence helping stakeholders to make informed commercial decisions. The target variable, GlobalSales, was predicted using a mix of numerical and category characteristics. These were sales numbers from four key areas: NASales, EUSales, JPSales, and OtherSales. Apart from numerical information, categorical characteristics including Genre, Platform, Publisher, and Year were encoded and included since they were important for sales performance.

EDA findings showing notable links between worldwide sales and NA/EU sales guided the choice of features. Furthermore, the notable effect they had on success patterns supported the addition of genre and platform. By combining these elements, the model sought to find trends relevant to a broad spectrum of game types, historical eras, and geographic marketplaces.

1.6.Model Implementation

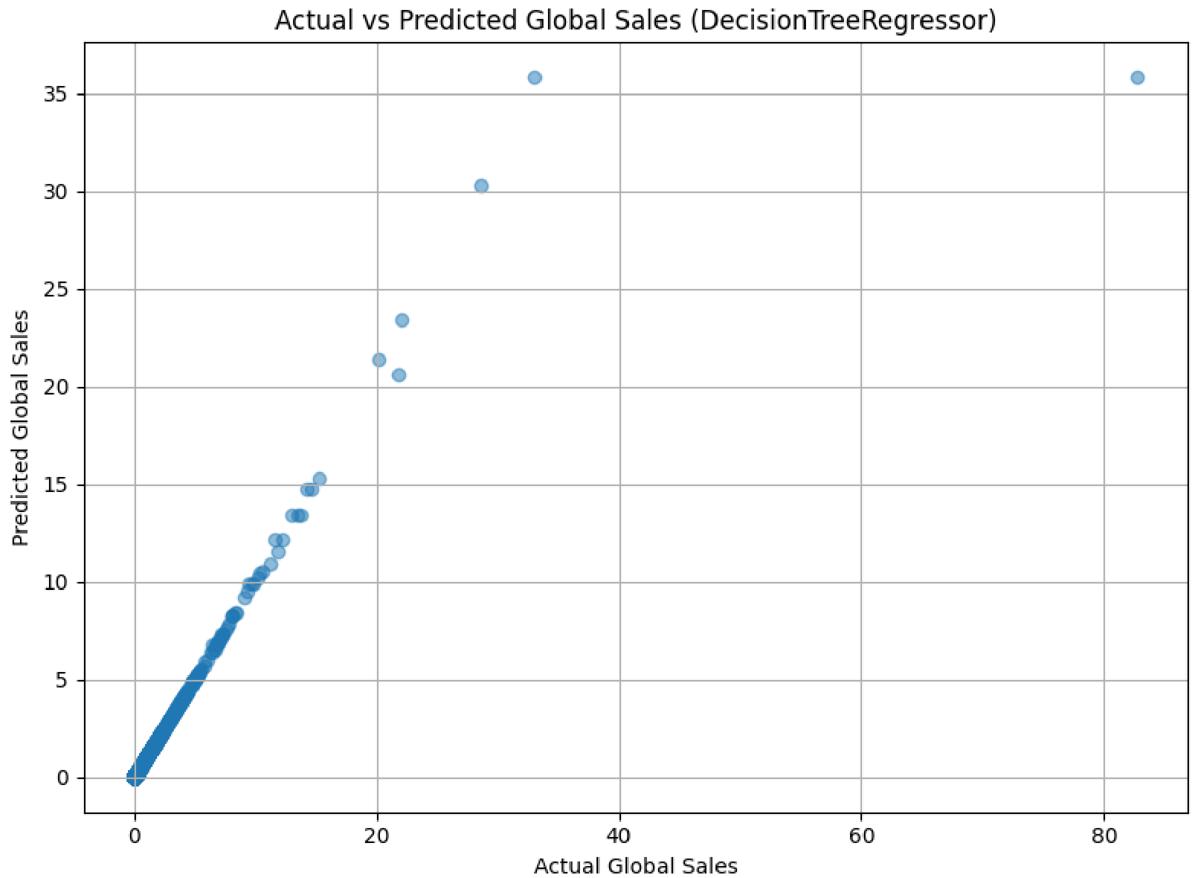
The Decision Tree Regressor was chosen to create a predictive model because of its capacity to capture non-linear correlations, manage categorical variables, and offer understandable decision routes. Efficient for datasets with varied kinds of input data, the model does not need previous feature scaling or normalisation.

An 80:20 split was used to separate the dataset into training and testing sets. While numerical sales data was used directly, categorical attributes were one-hot encoded to prevent ordinal consequences. Initially, the default Scikit-learn hyperparameters were employed to provide a baseline model for performance evaluation.

Further experimentation was based on this first model. Its tree-based architecture readily fit the complexity of the data, especially skewed distributions, and feature interactions. Future versions were intended to include hyperparameter tuning e.g., max depth, min samples split to enhance generalisation and lower overfitting.

1.7.Model Evaluation & Performance

Model evaluation was conducted using several regression metrics, including R^2 score, Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Squared Error (MSE). The Decision Tree Regressor achieved strong results across all metrics: an R^2 of 0.8614, RMSE of 0.6668, MAE of 0.0122, and MSE of 0.4446. These figures reflect high predictive accuracy with minimal deviation from true values.



A scatter plot of actual against expected sales proved the model's efficacy even further by indicating that most forecasts landed near the optimal diagonal line. Especially for games with low to moderate sales, most forecasts congregated around this line, suggesting a good fit. For games with extremely high sales, when the model underpredicted or overpredicted the result, certain variations did exist, though. The long-tailed distribution of the dataset, in which a small number of blockbuster titles disproportionately influence overall sales, explains these variations naturally.

Future actions might include decision tree pruning, grid search-based hyperparameter tuning, or application of ensemble methods such Random Forest or Gradient Boosting to lower sensitivity to such outliers and enhance overall performance, which usually provide more strong and generalisable performance.

Other models were investigated as well to evaluate the Decision Tree's performance. Linear Regression underperformed and neglected the complicated, non-linear patterns in the data since it expects a linear relationship between characteristics and the target variable. Particularly when comparing high-variance regional sales, the k-Nearest Neighbours algorithm found it difficult to generalise in the presence of overlapping values and distorted distributions. Offering a mix of accuracy, interpretability, and compliance with the dataset's structure, the Decision Tree Regressor finally proved to be the most successful choice.

1.8.Bias , Fairness and Ethical Considerations

During the modelling phase, attempts were made to spot, reduce prejudice, and guarantee ethical, accountable results. Initial data investigation revealed over-representation of some publishers, namely Nintendo, and a strong emphasis on the North American market. These disparities run the danger of distorting forecasts in favour of over-represented groups.

Regional sales were managed as distinct characteristics to prevent any one area controlling the forecast, thereby addressing fairness. Preventing the algorithm from inferring non-existent hierarchies, categorical variables like platform, genre, and publisher were one-hot encoded. These actions were meant to keep model performance while lowering prejudice.

Ethically speaking, the dataset raised little issues. It had no user interaction data, personally identifiable information (PII), or material connected to vulnerable populations. All entries were commercially available public data provided under open license, so guaranteeing conformity with university research regulations and data protection norms.

Still, there are restrictions. Excluding mobile and digital-only game markets, the dataset omits a sizeable portion of the contemporary gaming ecosystem. Furthermore, depending on past actual sales data restricts the generalisability of the model in predicting future trends. A more ethically whole model would include more recent types of distribution and more general market participation.

1.9.Conclusion

This paper showed how a methodical machine learning process may be used on a commercial dataset to produce predictive insights. Every stage added to a strong regression model able to predict worldwide video game sales with great accuracy from exploratory data analysis to model training and ethical consideration. Its interpretability, non-linear learning capability, and competitive performance led to the Decision Tree Regressor. It outperformed more basic models and underlined the need of genre/platform characteristics and regional data. Though there are several data coverage restrictions, the model demonstrated good generalisation inside the dataset.

Incorporating more varied data sources, fine-tuning model parameters, and using ensemble learning strategies would help future developments even more. Given the circumstances, the initiative emphasises how data science helps strategic decision-making in the video game industry.

2.Natural Language Processing and Deep Learning

This section explains how Natural Language Processing (NLP) and deep learning were used to build a model that can tell whether a product review is positive or negative. The goal was to analyse the written content of Amazon reviews and use it to predict the customer's overall opinion. The project shows how raw text can be cleaned and turned into useful data for machine learning.

To do this, the Amazon Product Reviews dataset (Rumi, n.d.) was downloaded from Kaggle. It contains millions of reviews along with star ratings. For this project, only the review text and ratings were used. The dataset was chosen because it contains a lot of useful opinion-based content, follows ethical standards, and does not include any personal information.

The next parts of this section explain how the data was cleaned, how word meanings were turned into numbers using GloVe, how the neural network was built, and how the model's results were measured using accuracy scores and other evaluation methods.

2.1. Text Dataset Selection and Preprocessing

The dataset selected for this project was the Amazon Product Reviews dataset, sourced from Kaggle (Rumi, n.d.). This dataset contains over three million customer reviews, providing both unstructured textual data (in the form of written product feedback) and structured metadata such as star ratings. It was selected due to its rich sentiment-bearing content, large volume, and alignment with the educational and ethical standards required by this coursework.

The dataset is ideal for binary sentiment classification tasks as it includes both the natural language input (review text) and a corresponding numerical label (star rating), which can be used to infer the sentiment polarity. Additionally, the dataset does not contain personally identifiable information, making it ethically compliant with institutional research requirements and GDPR regulations.

To ensure the quality of the sentiment classification model, the raw Amazon review data underwent a structured and rigorous preprocessing pipeline. This process transformed noisy, unstructured textual feedback into clean, consistent, and numerically interpretable inputs suitable for deep learning. The preprocessing steps were carefully designed to support the feedforward neural network architecture used in this study, which required fixed-length numerical input vectors for training and prediction.

The initial stage involved data cleaning. Irrelevant columns such as product-related metadata (e.g., product ID, category, or brand), timestamps, and user identifiers were removed, as they did not provide useful information for sentiment analysis. The review text was then normalised by converting it to lowercase, removing punctuation, special characters, numbers, and excess whitespace using regular expressions. This helped create a consistent and clean textual dataset suitable for further processing.

Stopword removal was then conducted using the Natural Language Toolkit (NLTK), a widely used library for text processing. Common stopwords such as “the,” “is,” “and” and “at” were filtered out, as these high-frequency words offer little semantic value and may dilute sentiment-relevant terms. After stopword removal, the cleaned text was tokenised using NLTK’s word tokeniser. This step splits each review into individual word tokens, which allows the text to be converted into a machine-readable format for embedding.

Next, pre-trained GloVe (Global Vectors for Word Representation) embeddings were downloaded and used. The 100-dimensional version was chosen as it offered a good balance between capturing word meaning and keeping things efficient. Each word in a review was converted into a GloVe vector, and the average of these vectors was used to represent the whole review. While this approach removes word order, it still gives a useful summary of the review’s meaning and works well with a basic neural network.

After text cleaning and embedding, sentiment labels were constructed based on the star ratings. Reviews with a rating of 1 or 2 stars were labelled as negative (0), while those rated with 4 or 5 stars were labelled as positive (1). Neutral reviews with 3 stars were excluded to eliminate ambiguity and sharpen the decision boundary between classes. This binary labelling approach enabled the model to focus on clear sentiment distinctions, improving classification performance and reducing training noise.

Finally, the processed feature vectors and associated sentiment labels were split into training and test sets using an 80/20 split. This ensured the model had sufficient data for learning while preserving an unseen dataset for performance evaluation. The resulting cleaned and structured dataset laid a robust foundation for model training, enabling the deep learning classifier to generalise effectively across previously unseen customer reviews.

2.2. Model Training Configuration

A deep learning model was built using a feedforward neural network to classify Amazon product reviews as either positive or negative. The model was created using Keras, which offers a simple way to build and train deep learning models, while TensorFlow worked in the background to carry out the actual training process.

This task was set up as a supervised learning problem, where each review was labelled as either 0 (negative) or 1 (positive), based on the product’s star rating. Reviews with 3-star ratings were removed to keep the classification clear. The model was trained to output a number between 0 and 1, and anything above 0.5 was classified as positive.

To prepare the text data for training, each review was turned into a 100-dimensional vector using pre-trained GloVe word embeddings. These embeddings help capture the meaning of words. By averaging the vectors for each word in a review, we created a fixed-size input for the model, allowing us to use a basic neural network without needing more complex models that handle word order.

The network had two main layers. The first layer had 128 units (neurons) and used a ReLU activation function, which helps the model learn patterns effectively. A dropout layer followed to randomly turn off some neurons during training, helping prevent overfitting. The second layer had 64 units with the same structure. Finally, a single output neuron used a sigmoid activation function to produce a score between 0 and 1 the predicted sentiment.

The model used binary cross-entropy as the loss function, which works well for two-class problems. It was trained using the Adam optimiser, a popular method that adjusts itself to learn efficiently. Accuracy was measured during training and validation to keep track of progress and avoid overfitting.

The model trained for five rounds (epochs) with a batch size of 32, and 20% of the training data was used to check how well it generalised to new data. Training went smoothly, with stable accuracy and loss across both the training and validation sets.

This feedforward design was chosen for its simplicity and fast performance, especially on large datasets. While it doesn't understand the order of words like more advanced models (such as LSTMs or Transformers), it still gave good results with minimal training time. Dropout helped prevent the model from memorising the data too closely.

Although more advanced models could improve results by better understanding context and word order, this simple approach provided a strong and reliable starting point for large-scale sentiment analysis.

2.2.1 Evaluation and Insights

Following the training phase, the performance of the sentiment classification model was evaluated using a combination of statistical metrics and visual diagnostics. The aim was to assess the model's ability to generalise to unseen data, evaluate how well it performs on each sentiment class, and identify limitations and opportunities for further optimisation.

2.2.2 Accuracy Score

The model achieved an overall accuracy of 88.30% on the test set, calculated using Scikit-learns accuracy score () function. This indicates that the model correctly predicted sentiment for 9 out of every 10 reviews, suggesting solid overall predictive capability. However, accuracy alone can be misleading, particularly in imbalanced classification problems such as this one, where the number of positive reviews significantly exceeds the number of negative ones. Therefore, further evaluation using class-specific metrics was necessary to gain a more nuanced understanding of model performance.

2.2.3 Classification Report

A detailed classification report was generated to evaluate the model's performance for each class using precision, recall, and F1-score. For negative reviews (label 0), the model achieved a precision of 0.79, recall of 0.34, and an F1-score of 0.47. In contrast, for positive reviews (label 1), it achieved a significantly higher precision of 0.89, recall of 0.98, and an F1-score of 0.93. These results reveal a considerable imbalance in predictive power: while the model is highly effective at identifying positive sentiment, it struggles to correctly detect negative sentiment, misclassifying many negative reviews as positive. This imbalance is evident in Figure 2.1, which presents the classification report output, highlighting the model's skewed performance and its bias toward the dominant class.

```
...   3287/3287 ━━━━━━━━━━━━━━━━━━━━ 5s 2ms/step
      Classification Report:
                    precision    recall  f1-score   support

                 0       0.79      0.34      0.47     16379
                 1       0.89      0.98      0.93     88784

          accuracy                           0.88    105163
         macro avg       0.84      0.66      0.70    105163
      weighted avg       0.87      0.88      0.86    105163
```

Figure 2.1: Classification report generated from Scikit-learn showing class-wise performance. While the model achieves strong results for positive sentiment (Precision: 0.89, Recall: 0.98, F1-score: 0.93), it underperforms on negative sentiment (Recall: 0.34), highlighting class imbalance.

2.2.4 Confusion Matrix

To further investigate the model's misclassification patterns, a confusion matrix was plotted. As shown in Figure 2.2, the matrix reveals 87,321 true positives and 5,540 true negatives, indicating that the model performs well on positive samples. However, it also shows 10,839 false positives instances where negative reviews were incorrectly predicted as positive and 1,463 false negatives, where positive reviews were misclassified. The large number of false positives aligns with the low recall for the negative class and confirms that the model is biased toward predicting the majority class (positive sentiment). This imbalance limits the model's usefulness in contexts where detecting dissatisfaction (negative sentiment) is critical, such as customer support or quality assurance.

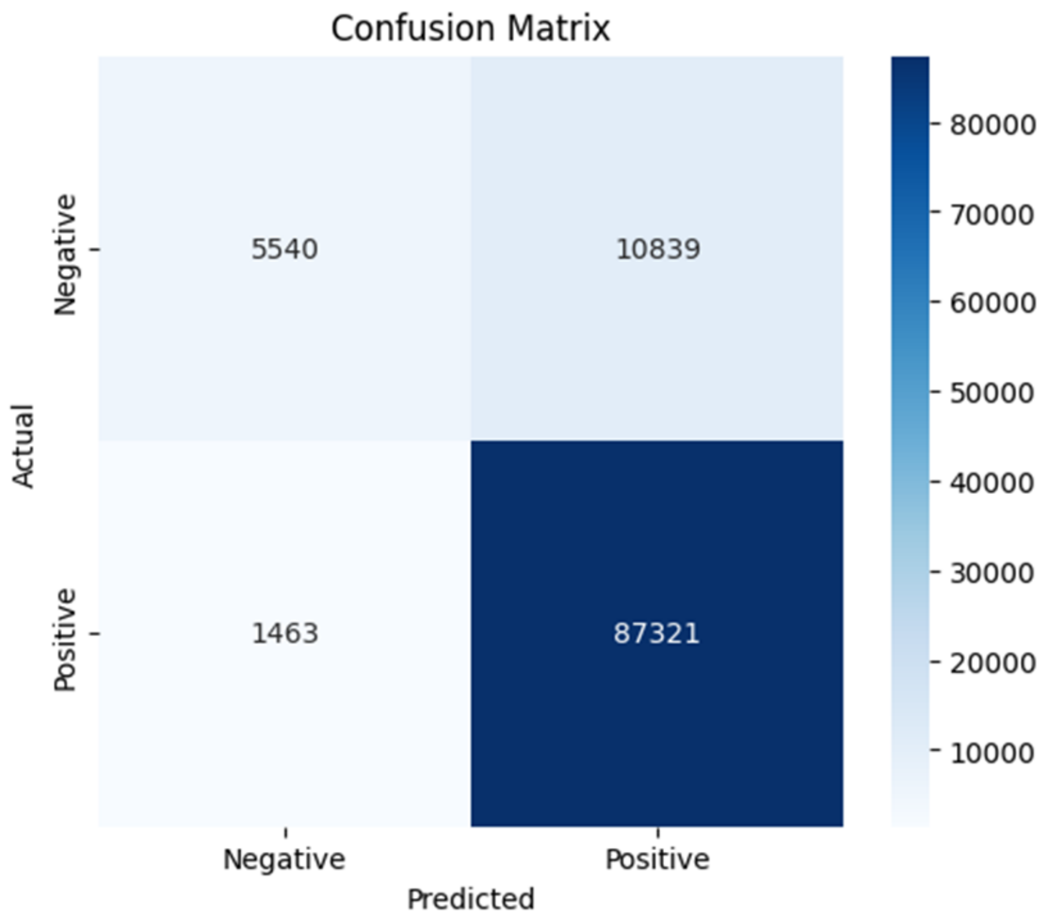


Figure 2.2: Confusion matrix visualising prediction outcomes. A high number of false positives (10,839) indicates the model's tendency to misclassify negative reviews as positive, reflecting the class imbalance observed in the dataset.

2.2.5. Training and Validation Curves

Learning curves of accuracy and loss throughout five epochs showed the model's training process. Two subplots are shown in Figure 2.3; the left one shows training and validation accuracy while the right one shows the related loss values. While validation accuracy stayed near and steady all through, showing continuous generalisation, training accuracy slowly rose to about 88.4%. Likewise, both training and validation loss fell consistently without divergence, implying the model did not experience overfitting. Even on a big dataset, this good learning behaviour shows how well dropout regularisation preserves generalisation performance.

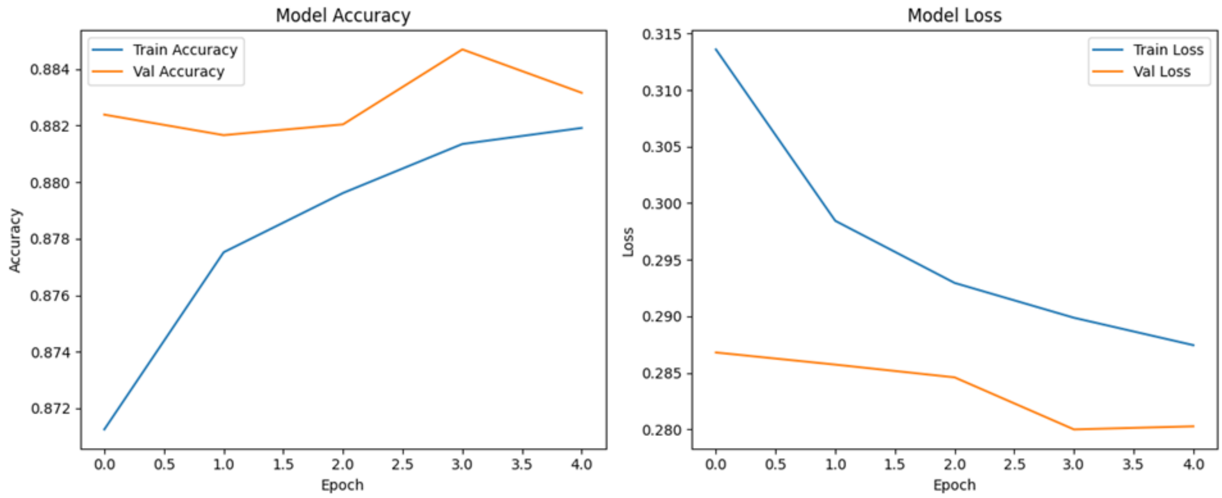


Figure 2.3: Training and validation accuracy (left) and loss (right) over five epochs. The plots show stable learning with consistent improvement and minimal divergence, indicating no major overfitting.

2.2.6 Discussion: Strengths and Limitations

The model demonstrates several key strengths. First, its overall accuracy of 88.3% on a large and diverse review dataset reflects strong generalisation ability. The consistent training and validation performance confirms that the model learned effectively without overfitting. Additionally, the simplicity and efficiency of the feedforward architecture made it suitable for large-scale training while still achieving high precision and recall for positive sentiment. However, the model also exhibits important limitations. A major limitation is its low recall of 34% for negative sentiment, which significantly reduces its ability to detect dissatisfied customers. Furthermore, the use of averaged GloVe embeddings means that local word order and syntactic structure are lost, limiting the model's ability to interpret nuanced phrases or sarcasm. As a result, reviews with subtle negative expressions may be incorrectly classified.

2.2.7 Recommendations for Improvement

To address the observed class imbalance and improve negative sentiment detection, several strategies are recommended. First, class rebalancing techniques such as SMOTE (Synthetic Minority Oversampling Technique), assigning class weights, or oversampling negative examples during training could help reduce bias toward the majority class. Secondly, upgrading the model architecture to incorporate sequential models like LSTMs or GRUs would allow it to capture word order and contextual meaning more effectively.

Alternatively, Transformer-based models such as BERT (Bidirectional Encoder Representations from Transformers) or RoBERTa (Robustly Optimised BERT Approach) could be adopted. These models are pre-trained on massive text and capture deep contextual relationships between words by considering both left and right context simultaneously. Unlike GloVe, which uses fixed word embeddings and ignores word order, BERT and RoBERTa generate dynamic, context-aware embeddings. This makes them significantly better at understanding complex sentiment, sarcasm, and subtle emotional cues in text.

Finally, performance could be further improved through longer training, early stopping, and hyperparameter tuning, including learning rate and dropout adjustments. These enhancements would increase the model's robustness and its ability to detect underrepresented or nuanced sentiment signals.

2.3. Conclusion

Using Natural Language Processing (NLP) and deep learning methods, this part has offered a whole end-to-end sentiment analysis pipeline. The work concentrated on categorising customer reviews from the Amazon Product Reviews database into binary sentiment classifications: positive or negative.

Starting with a large-scale, ethically sound dataset from Kaggle, the project Reviews were pre-processed by means of painstaking text cleaning techniques including lowercasing, stopword removal, punctuation stripping, and tokenisation. Averaged per review to provide fixed-size input vectors appropriate for neural network training, 100-dimensional GloVe embeddings were used to translate reviews into numerical representations. Excluding neutral (3-star) evaluations helped to clarify the classification border and lower label uncertainty.

A feedforward neural network was created using Keras, a high-level deep learning framework that simplifies the building and training of neural networks. Running on top of TensorFlow, an open-source machine learning framework developed by Google, it provides the backend infrastructure for model computation. In the architecture, a sigmoid output unit for binary classification follows two ReLU-activated dense layers with dropout regularisation. Constructed utilising the Adam optimiser and binary cross-entropy loss, the model was then trained over five epochs using a validation split.

The evaluation criteria showed that the model had a good test accuracy of 88.30% and did especially well in detecting positive emotion (Precision: 0.89, Recall: 0.98, F1-score: 0.93). On negative evaluations, however, it performed far worse, with a recall of just 0.34, indicating trouble in accurately spotting consumer discontent. These results were supported by the classification report and confusion matrix, which drew attention to a class imbalance problem whereby the model strongly preferred the majority class. With no indications of significant overfitting, visualisations of training and validation accuracy and loss curves verified that the model learnt efficiently and generalised well.

Although the model was simple, it served as a good start. Among other things, its disadvantages included averaging word embeddings resulting in loss of word order and context as well as problems forecasting the minority class. Among the suggestions to improve this were adopting class balancing techniques, sequence-based models such LSTM or Transformers, and contextual embeddings such BERT for better knowledge of complex language.

This sentiment analysis model shows that a quick and scalable method to text categorisation involves combining classical pre-trained word embeddings with deep learning. Although the baseline model is successful, the findings highlight even more the need of dataset balancing and context-aware modelling for capturing the entire range of emotion in natural language data.

3. References

Gregorut, V. (n.d.). *Video Game Sales Dataset*. [online] Kaggle. Available at: https://www.kaggle.com/datasets/gregorut/videogamesales [Accessed 20th Jan. 2025].

Rumi, A. (n.d.) *Amazon Product Reviews*. Kaggle. Available at: https://www.kaggle.com/datasets/arhamrumi/amazon-product-reviews (Accessed: 10 March 2025).