

1. Introduction

This project aims to build and optimize a neural network model for binary classification using the *IBM HR Analytics Employee Attrition* dataset. The main objective is to predict whether an employee will leave the company (“Attrition”) based on various demographic, performance, and job-related features.

The assignment’s focus was to improve model accuracy through systematic hyperparameter tuning, testing different network architectures, and reducing overfitting. Three neural network configurations were compared by adjusting dropout rates, learning rates, and the number of hidden layers.

By analyzing the results and validation performance of each model, the goal was to identify a balanced configuration that achieves high accuracy while maintaining generalization ability.

2. Dataset Description and Preprocessing

The dataset used in this work is the IBM HR Analytics Employee Attrition & Performance dataset, publicly available on Kaggle.

The dataset contains various employee-related attributes such as age, department, job role, monthly income, years at the company, and work-life balance. The target variable, Attrition, indicates whether the employee left the company (Yes = 1, No = 0).

Before training, categorical variables were encoded using One-Hot Encoding, and numerical features were normalized with StandardScaler to ensure that all input features contribute equally to the model’s learning process. The dataset was then split into 80% training and 20% testing subsets using stratified sampling to preserve class proportions.

These preprocessing steps ensured that the model received clean, scaled, and balanced input data suitable for neural network training.

3. Model Design and Hyperparameter Selection

To explore different network architectures and minimize overfitting, three neural network configurations were designed and compared:

- **Model A (No Dropout):**

A simple baseline model with three dense layers (64, 32, 16 neurons). No dropout regularization was applied.

Purpose: Establish a reference accuracy without regularization.

- **Model B (Dropout):**

A moderately deep model with two hidden layers (64, 32 neurons) and dropout rates of 0.3 and 0.2.

Purpose: Prevent overfitting by introducing dropout and stabilize validation accuracy.

- **Model C (Deep + Slow):**

A deeper model (128, 64, 32 neurons) trained with a smaller learning rate (0.0005) and higher dropout rates (0.4, 0.3, 0.2).

Purpose: Test the impact of deeper architecture and slower learning on convergence and generalization.

The optimizer used for all models was Adam, and the loss function was binary cross-entropy since the task involves binary classification.

Hyperparameters such as the learning rate, batch size, and dropout percentage were chosen experimentally based on common practices for tabular datasets and through multiple test runs. Early stopping was also applied to prevent unnecessary training when the validation loss stopped improving for five consecutive epochs.

4. Training and Evaluation Results

Each model was trained for up to 50 epochs with early stopping enabled. The following table summarizes the final test accuracies:

Model	Description	Test Accuracy
Model A	No Dropout	0.8571
Model B	Dropout (0.3, 0.2)	0.8673
Model C	Deep + Slow	0.8639

Model B achieved the highest validation accuracy (0.8673), showing the best balance between bias and variance.

Validation loss and accuracy curves (Figures 1 and 2) further confirmed that dropout improved model stability and reduced overfitting.

While Model C showed competitive accuracy, it required longer training and exhibited minor fluctuations, likely due to increased network complexity. Model A, on the other hand, showed early signs of overfitting, as training accuracy continued to rise while validation performance plateaued.

Early stopping effectively prevented overfitting, and the dropout layers ensured better generalization on unseen data.

5. Discussion and Interpretation

The results show that dropout regularization and careful tuning of hyperparameters play a critical role in achieving better generalization.

Model A, without dropout, quickly reached high training accuracy but showed a smaller improvement in validation accuracy, indicating early overfitting.

Model B, with dropout layers and moderate learning rate, demonstrated more stable validation performance and achieved the highest test accuracy (0.8673).

This suggests that introducing dropout prevents the model from memorizing training samples, allowing it to generalize better to unseen data.

Model C, which used a deeper architecture with a lower learning rate, achieved comparable results but showed slower convergence. The deeper structure did not yield a significant performance boost, likely due to the limited complexity of the dataset.

Overall, Model B provided the best balance between complexity, training stability, and validation accuracy.

Additionally, early stopping effectively prevented unnecessary training and helped avoid overfitting. The validation loss curve flattened after approximately 12–15 epochs, confirming that the model reached its optimal capacity.

6. Conclusion

In this project, a binary classification neural network was built to predict employee attrition using the IBM HR Analytics dataset.

Three model configurations were designed and compared to analyze the effect of dropout, network depth, and learning rate on accuracy and overfitting.

The results showed that applying dropout (0.3–0.2) and early stopping produced the most balanced and reliable model.

Model B achieved a final test accuracy of 0.8673, outperforming both the simpler and deeper architectures.

These findings highlight that optimal performance in tabular binary classification tasks is often achieved through regularization techniques and careful tuning rather than excessive network depth.

The final model demonstrates good predictive power and generalization, making it suitable for real-world HR analytics applications such as employee retention analysis and workforce planning.

