Predictive Modeling Report: ANN Model Architecture and Training

1. Architecture of the Artificial Neural Network (ANN) Model

1.1. Input Layer

Number of Neurons: 16
Description: The input layer consists of 16 neurons, corresponding to the number of features in the dataset after preprocessing and splitting it to training and testing sets. This layer is responsible for receiving and processing the input data.

1.2. Hidden Layer

Number of Neurons: 64, Middle layer
Activation Function: Rectified Linear Unit (ReLU)
Description: The hidden layer, with 64 neurons, deploys the ReLU activation function. ReLU introduces nonlinearity into the model, allowing it to capture complex patterns and relationships within the data. The choice of 64 neurons is optimal to balance model complexity and computational efficiency.

1.3. Output Layer

Number of Neurons: 1
Activation Function: Sigmoid
Description: The output layer consists of a single neuron with a sigmoid activation function. This configuration is suited for binary classification tasks, providing a probability estimate of customer churn. The sigmoid function outputs a value between 0 and 1, indicating the likelihood of the positive class (churn).

1.4. Optimizer

Algorithm: Adam
Description: The Adam optimizer is employed to update the model's weights. It combines the advantages of two other extensions of stochastic gradient descent, namely AdaGrad and RMSProp. Adam adapts the learning rate for each parameter, improving convergence and reducing training time.

1.5. Loss Function

Function: Binary Crossentropy
Description: Binary cross entropy is used as the loss function, measuring the difference between the predicted probabilities and the actual binary outcomes. This loss function is appropriate for binary classification problems and facilitates effective model training.

2. Training of the ANN Model

2.1. Training Parameters

Epochs: 20
Batch Size: 32
Validation Split: 20% of the training data

The ANN model was trained for 20 epochs with a batch size of 32. A validation split of 20% was used to evaluate the model's performance during training and make adjustments as necessary. This approach helps in monitoring the model's generalization ability and preventing overfitting.

2.2. Performance Metrics

Training Accuracy: 79.73%
Training Loss: 0.4192
Validation Accuracy: 81.19%
Validation Loss: 0.4057

The performance of the trained model was evaluated based on accuracy and loss metrics. The training accuracy of 79.73% and validation accuracy of 81.19% indicate the model's effectiveness in predicting customer churn. The corresponding loss values reflect the model's error rate.

3. Resources used

Document on Model Architecture:
https://github.com/Pratapuchai2056/Churn_Analysis_Project_WIL/blob/main/Predictive_Modeling/Ann_Modelling.md

Trained ANN Model File:
https://github.com/Pratapuchai2056/Churn_Analysis_Project_WIL/blob/main/Predictive_Modeling/model_trained_on_resampled_data.keras