Joseph Annand

DSE 6211

Final Project

## Executive Summary

This report outlines the potential capabilities of a feed-forward dense neural network classification model to identify and mitigate the risk of cancellations for bookings at ABC hotels. Using a dataset of 36,238 bookings, including information such as number of guests, room type, and arrival date, a classification model was trained to predict if a booking will be cancelled or not and evaluated on a held-out test set. Initially, the dataset underwent preprocessing, including splitting into training and test sets and feature engineering. Categorical features were one-hot encoded, rare cases are grouped, and additional features such as season and weekday/weekend arrival are derived from the arrival date. Numerical features are scaled to ensure fair representation. Two neural network models, named Model 1 and Model 2, were trained and evaluated. Model 1, initially designed to overfit the training data, was adjusted with techniques including L2 regularization, batch normalization, dropout, and early stopping to improve generalization. Model 2, with a smaller architecture, undergoes similar adjustments. The evaluation of both models involved analyzing loss and accuracy curves, ROC curves, and calibration curves. Model 1, with its larger architecture, demonstrates flexibility in solving the problem and achieves high accuracy, with well-calibrated predicted probabilities. Model 2, although having similar performance in terms of AUC, showed slightly less reliability in its predicted probabilities. Based on these evaluations, Model 1 was deemed the best for classifying future booking status at ABC hotels. Its flexibility, high AUC, and reliable predicted probabilities make it the preferred choice over Model 2. We suggest using the model to classify bookings into high and low-risk categories based on cancellation probabilities, allowing targeted promotional efforts to reduce cancellations. Overall, the report highlights the effectiveness of employing neural network models to predict and mitigate booking cancellations, providing valuable insights for ABC hotels to optimize their operations and revenue.

## Approach & Data

To help ABC hotels identify bookings that have a high risk of cancellation, a data set containing 36,238 bookings and a supervised feed-forward dense neural network are used to determine the probability of cancellation for a given booking. Each observation in the given data set is a different booking at ABC hotels and includes information about each booking, including number of guests, room type, arrival data, etc. A neural network that predicts whether a booking will be cancelled or not was trained using ABC hotels data set. Of the 36,238 bookings, 11,878 bookings were cancelled, approximately 33%. To mitigate such high rates of cancellation, a model that predicts the likelihood of cancellation for a given booking can be employed by ABC hotels to determine which customers are most in need of targeted advertisements or special promotions and offers to reduce the risk of them cancelling.

Prior to training the neural network model, the data set had to be processed for the purpose of extracting more useful information from what was already provided and transforming the data into a format that the neural network can take as input. To begin, the ABC hotels data set was divided into a training set and a test set. The training set contained 75% of the observations from the original data set and was used to train the neural network. The test set contained the remaining quarter of the original data set and was used to evaluate the neural network once it was trained. A series of feature engineering steps were performed on the training set to prepare the data for the neural. Initially, the categorical features, non-numeric variables, of the training set needed to be one-hot encoded before it could serve as meaningful input in the model. One-hot encoding will create a separate feature for each of the unique values of all the categorical features in the training set. Some features, however, have rare cases, such as one observation with nine children listed on the booking, for example. To reduce the number of features produced in the training set while also ensuring that the training and test set will have the same number of features, rare values in a handful of categorical features are grouped together. Specifically, number of previous bookings not cancelled, number of children, number of previous cancellations, number of week nights, number of weekend nights, and number of special requests were the categorical features with rare cases that were grouped together. The arrival date of each booking does not appear to be useful on its own, but more useful information can be inferred from it and used in the model. From the arrival date, the season of the arrival date and whether the date is a weekday or weekend was determined and added to the training set. After converting all categorical features to factors, they were one-hot encoded. The same steps were performed on the test set.

After engineering the categorical features in both the training and test set, the numerical features were scaled. The two numerical features, lead time and average price per room, are represented in days and dollars, respectively. To ensure that neither feature dominates the other simply because of the difference in units, these features are centered using the means and standard deviations of each feature in the training set. Finally, the target variable or the labels are 0/1 valued so that 0 represents the booking not being cancelled and 1 represents the booking being cancelled.

Two feed-forward dense neural networks with slightly different architectures were trained and evaluated. The relu function is the default activation function for hidden layers in a multilayer perceptron, so the relu function is used for all hidden layers of the neural networks. For the output layer, the number of units is equal to 1 because the aim of the neural network is to provide a single output for predicting the status of a given booking. For a binary classification problem, the sigmoidal activation function is used so that all outputs are between 0 and 1, with 0 corresponding to a 0% chance the booking will be cancelled and 1 corresponding to a 100% chance the booking will be cancelled. The optimizer for the models is RMSprop, which is an adaptive learning rate algorithm that helps reduce the amount of computational effort required to train the neural network compared to other optimizers. The loss function used for the neural network is binary cross-entropy; this loss function calculates the dissimilarity between true labels and predicted labels when the output is between 0 and 1. The metric used in training the model is accuracy to measure how well the model is performing in terms of predicting booking status.

Initially, a model that could easily overfit the data, which we will call Model 1, was created. This model consisted of three hidden layers, two of which had 100 nodes and the third had 50 nodes. This architecture is much larger than what is likely required for the training set, as there are more nodes in the first two hidden layers than there are features in the training set. Once the model that can overfit is created, the overfit model is modified to increase generalization, or the model’s ability to make predictions on unseen data. L2 regularization, batch normalization, and dropout are implemented to do so. In L2 regularization, overfitting is mitigated by placing constraints on the weights of the model through adding to the loss function of the model at a cost proportional to the square of the value of the weight coefficients. In our model, this coefficient is 0.002, and it is used in every hidden layer. Batch normalization standardizes the inputs at every layer of the neural network, a similar action that was manually performed during the feature engineering steps. Dropout is a regularization technique that drops out some output features in each layer of the model. As stated previously, the model is much larger than what is probably required for this problem; dropout reduces the size of the output of each hidden layer by 50% to reduce overfitting. Finally, early stopping is used to stop training the model once the validation loss stops decreasing after two epochs, so that the model does not overfit and the model with the best performance on the validation set is found. After training this first model, a second model with a smaller capacity, called Model 2, was created and trained. The smaller model consists of two hidden layers: one with 75 nodes and the other with 37 nodes. The same generalization techniques were employed with a change in the drop out rate from 0.5 to 0.2.

## Detailed Findings & Evaluation

To begin, Model 1 was developed to overfit the training data set. A model that can overfit the training set is sufficiently flexible for the problem. Figure 1 shows the loss and accuracy curves for the overfitting model. The validation loss curve quickly reaches an inflection point after which the loss increases gradually, while the training loss continuously decreases for all 200 epochs. Because the validation and training loss curves diverge to this extent, the model overfits the data and has the appropriate amount of flexibility needed for the problem; the focus of developing the model shifts to maximizing the generalization power.

After adding early stopping, batch normalization, and regularization techniques to the model; it no longer overfits the data. Figure 2 shows the loss and accuracy curves for Model 1 with improved generalization. The training loss and validation loss curves are nearly identical, and the accuracy curves are also similar, meaning Model 1 now performs equally well on training data and unseen data. Notice that the minimum validation loss in Figure 2 increased from Figure 1. This is expected considering that L2 regularization was implemented to reduce overfitting.

Now that Model 1 has improved generalization power, it is evaluated using the test set that was held out from the original data set. Model 1 is used to predict the booking status of all the observations from the test set. The true positive rates, the probability the model will accurately predict a true positive result to be true, and the false positive rates, the probability that the model will predict a positive result when it is actually negative, were calculated for a variety of threshold values between 1 and 0. These values are plotted on the receiver operating characteristic, or ROC, curve in Figure 3. The ROC curve shows how well the binary classification model performs on the test data. The curve in Figure 3 is far from the dashed, diagonal line; it increases almost perfectly vertically before sharply curving and plateauing. Its shape maximizes the area under the curve (AUC), which is 0.916. For reference, an AUC of 1.0 indicates a perfect classifier. Finally, a calibration curve for Model 1 is shown in Figure 4. This curve tells us how well the predicted probabilities match the actual performance of the model. In other words, it shows how reliable the predicted probabilities are. The diagonal line represents a perfectly calibrated model in which for any bin midpoint, the observed percentage of observations predicted true should be equal to the value of the bin midpoint. The calibration curve follows the dashed line closely, meaning Model 1 is well-calibrated.

Model 2 was evaluated using the same process as Model 1. Figure 5 shows the loss and accuracy curves from training the smaller capacity model. Model 2’s overfitting capabilities are weaker than those of Model 1. The validation loss curve does not diverge from the training loss curve as much as seen in Figure 1 for Model 1. Additionally, the minimum training loss is much higher in Figure 5 than in Figure 1. The minimum validation losses in Figures 1 and 5, however, are similar, meaning we might expect to see Model 2 have similar performance on the test set as Model 1.

A graph of data and data

Description automatically generated with medium confidence

Figure 1: Overfitted model, min training loss = 0.143, max training accuracy = 0.940, min validation loss = 0.329, max validation accuracy = 0.864

Figure 6 shows the loss and accuracy curves of Model 2 after the techniques to improve generalization are applied. The validation and training curves are now very similar, and the minimum validation loss is similar to that in Figure 2. Model 2 with improved generalization is used to predict the targets for the test set. The ROC curve for Model 2 is shown in Figure 7. The shape of the curve indicates very good performance on predicting the booking status of the test set observations. The area under the curve is 0.916, the same result Model 1 produced. The calibration curve in Figure 8 indicates that Model 2 is fairly well calibrated; however, its predicted probabilities are not as reliable as those in Model 1.

A graph of data and data

Description automatically generated

Figure 2: Improved generalization model using early stopping, batch normalization, and regularization techniques, min training loss = 0.368, max training accuracy = 0.851, min validation loss = 0.369, max validation accuracy = 0.850.

A graph of a function

Description automatically generated

Figure 3: ROC Curve for Model 1 with improved generalization; area under the curve = 0.916

A graph with a line

Description automatically generated

Figure 4: Calibration curve for Model 1 with improved generalization

A graph of data and data

Description automatically generated with medium confidence

Figure 5: Model 2 loss and accuracy curves; min training loss = 0.227, max training accuracy = 0.904, min validation loss = 0.339, max validation accuracy = 0.862.

A graph of data and a graph of data

Description automatically generated with medium confidence

Figure 6: Model 2 with improved generalization loss and accuracy curves; min training loss = 0.364, max training accuracy = 0.854, min validation loss = 0.385, max validation accuracy = 0.846.

A graph of a function

Description automatically generated

Figure 7: ROC curve for Model 2 with improved generalization, area under the curve = 0.916.

A graph with a line

Description automatically generated

Figure 8: Calibration curve for improved generalization smaller capacity model

## Recommendations

Based on the detailed findings from the model evaluations, Model 1 is the best feed-forward dense neural network for classifying future booking status of bookings at ABC hotels. It clearly demonstrates flexibility for solving the problem, its high AUC demonstrates excellent performance in the classification of booking status, and its predicted probabilities of cancellation are very reliable based on its calibration curve. Compared to Model 2, a neural network with a smaller architecture, Model 1 has equal accuracy in predicting test data and is better calibrated. Because Model 1 is well calibrated, ABC hotels can be confident that regardless of the threshold, the model will provide accurate predictions. The probability threshold to classify the booking as high-risk or low-risk may vary for several reasons. If the company has committed a plethora of money and resources into cancellation prevention, the threshold may be set relatively low like 50% chance of cancellation. If the company has limited capacity for extra advertisements and offers, then the threshold may be set to a higher probability value like 75%. The model could also classify bookings with a high cancellation risk into priority groups. Then, the hotel could easily distinguish between a booking with a 90% chance of cancellation and a booking with a 75% chance so that it prioritizes sending advertisements and offers to the booking at a higher risk. Nonetheless, the model will provide useful information to ABC hotels that will help keep their rooms filled with customers.

## Appendix

Figures 9A through 12A show how Model 2 is affected by the successive addition of each technique to improve generalization.

A graph of data and data

Description automatically generated

Figure 9A: Model 2 using only early stop and batch normalization; min training loss = 0.279, max training accuracy = 0.878, min validation loss = 0.347, max validation accuracy = 0.850

A graph of data and data

Description automatically generated

Figure 10A: Model 2 adding L2 regularization; min training loss = 0.346, max training accuracy = 0.879, min validation loss = 0.416, max validation accuracy = 0.846

A graph of data and data

Description automatically generated

Figure 11A: Model 2 adding in drop-out technique; min training loss = 0.401, max training accuracy = 0.839, min validation loss = 0.404, max validation accuracy = 0.830

A graph of data and data

Description automatically generated

Figure 12A: Model 2 removing L2 regularization; min training loss = 0.379, max training accuracy = 0.831, min validation loss = 0.376, max validation accuracy = 0.830