Predicting Airline Passenger Satisfaction using Deep Learning

JONAH GONZALO
Institute of Computer Studies
Philippine State College of Aeronautics
Fernando Air Base, Lipa City, Batangas
PHILIPPINES

Abstract: -This paper aims to evaluate the performance of predictive algorithms for the prediction of flight passenger satisfaction. Naive Bayes, Logistic Regression, Deep Learning, Decision Trees, Random Forest, and Support Vector Machine were simulated using Invistico Airline datasets. The six predictive algorithms were tested and evaluated based on accuracy, classification error, accuracy, Precision, Recall, F-measure, sensitivity, specificity, and ROC-AUC comparison. Deep Learning has the highest accuracy at 93%, Gains at 28,406, and training time at 21ms. Arrival Delay in Minutes and Customer Type causes dissatisfaction while Departure Delay Time, Baggage Handling, Ease of Online Booking, and Inflight Entertainment are the predictors for flight satisfaction. The results of this study can be used as a guide for managers to understand their customers, determine the factors to enhance passenger satisfaction, improve airline service quality, and formulate numerous alternatives.

Keyword: Flight Passenger Satisfaction, Data Analytics, Artificial Intelligence, Airline Management

Received: May 11, 2024. Revised: January 22, 2025. Accepted: March 23, 2025. Published: June 2, 2025.

1 Introduction

The emergence of artificial intelligence is a major leap for airline industries particularly handling flight passenger operations, managements, sales, monitoring and service evaluation. Automation in the aviation industry was USD 112.3 Million in 2017 and projected to increase to USD 2,222,5 Million by 2025, at CAGR of 46.65% [1]. With the proliferation of gigantic flight data and the cloud-based applications of airlines and other aviation-related companies, precise anticipation of a number of individuals who won't appear for a flight can be predicted. Boarding passenger predictions [1] were models designed by dissecting and analyzing historical customer data, changing climate information, and other attributes for a passenger to rebook, change flights, a or no-show. There is a dire need for the development of AI applications that will analyze big data coming from airline information systems, AI can predict if a passenger will change flight or have no show. These predictions can be accessed by ground staff via a web application or mobile application in a real-time manner which is a crucial moment for decision making. Data analytics components can visualize insights such as customer behavior and frequent flyer updates.[1].

This paper contributes to the development of flight well-being of passenger through the analysis and evaluation of flight passenger data and designing the best algorithms that predicts flight passenger satisfaction. These algorithms could be used to assist airline managers that needs proactive management strategies such as predicting the behavior and satisfaction of passengers though visualization and data analytics.

This paper discovered the hidden insights that can be used to improve customer experience, common problems encountered in boarding and departure, in-flight experiences, passenger sentiments, food preferences, and other issues that contributes to well-being. Information mining can be applied to both printed information and numerical data. The issues on aircraft passenger satisfaction were addressed in this paper such as numerous influencing factors due its diverse and multi-cultural populations that needs exploration to be able to gain advantages for competition, customer loyalty and even to prepare difficult situations. The results in this study may be used as a reference for these companies as they choose which machine learning algorithm will have the highest probability of accurately predicting consumer satisfaction.

2 Problem Formulation

Just like in many papers [2],[4], [5], [6], [7], [8], [9], [10], [11], this study will explore different algorithms and compare their performance. Many types of deep learning algorithms have been applied to solve address issues in passenger satisfaction. There is a need to understand the behavior of delayed passengers who often displays disruptive at airport terminals which affects personal safety and negatively affects civil aviation efficiency and passenger satisfaction.

Many algorithms were tested to predict airline passenger satisfaction. This research will address some of the unsolved problems in the previous works of literature, specifically, the following research questions will be addressed:

1. What are the factors that predict the satisfaction and dissatisfaction of passengers?
2. What algorithms can be used to gain high accuracy of prediction for passenger satisfaction in terms of Gains, Total Prediction Time, and Training Time?
3. How will the performance of prediction algorithms be tested?
4. What are the factors that evaluate the accuracy of the prediction algorithm?
5. What model can be developed to predict airline passenger satisfaction?

3 Problem Solution

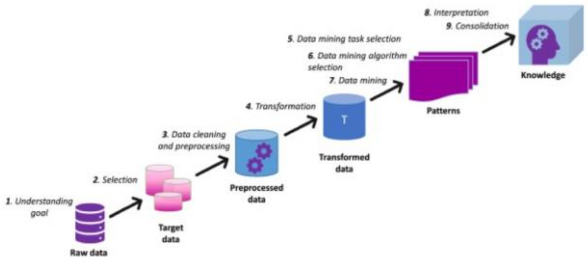
To find answers to research questions, scientific research approaches were utilized in search for solutions to the problems identified in the previous section. Such approaches are the research methods, algorithms, and evaluation strategies used to design the final output which is a prediction model/algorithm for flight passenger satisfaction.

3.1 Methods

3.1.1 Knowledge Discovery in Data Bases

In designing predictive models, various data mining frameworks were applied based on the type of prediction problems. This paper is based on one of the commonly used frameworks, Knowledge Discovery in Databases (KDD). It is the process that involves series of phases to that discover insights or hidden knowledge in the massive amounts of data coming from social media websites, company data, emails, mobile applications, databases, flatfiles, and even legacy databases. KDD is a dynamic framework that can be used to design predictive, descriptive and prescriptive models. Most of datasets are in raw form and KDD supports data preparation strategies such as data transformation, data cleaning, dimensionality reduction and data augmentation to ensure that the data to be used in training models are accurate [3]. The Knowledge Discovery in Databases Process involves steps as depicted in the figure 1.

Figure 1. Knowledge Discovery in Databases(KDD)



The processes in KDD can be accomplished through a series of steps that is performed as part of the simulation activities in datamining and machine learning. The selection of datasets requires both technical and ethical concerns with regards to its quality, legitimacy, validity, relevance and most importantly legitimacy of source. The source data has strong influence on the performance of algorithms that researchers should observe ethical standards in acquiring datasets. Raw datasets are commonly "noisy" which means data contain inaccurate or missing data. This problem can be solved through data pre-processing.

Datasets were cleaned by removing outliers, handling missing data, handling time sequence information and performing normalization. Data transformation can be done by transforming the format of data and by reducing the dimension of data while maintaining the patterns within the dataset.

Feature extraction can also be a part of data processing by identifying relevant features or variables as determinants to the prediction.

A good quality of dataset is the most appropriate data to be used to train predictive models such classification models, regression models and clustering models. The data mining components of KDD involves repetitive training and testing algorithms to ensure accurate and highly performing predictive and data mining models. For example, to predict whether a magazine subscriber will renew their subscription, clustering can be applied to group the subscriber database according to some patterns, and then create classification for each desired cluster by applying rule induction.

The final step is the interpretation of results that consists of translating the new model into a knowledge that is understandable to the users. Visualization is an interpretative technique in KDD that presents extracted patterns into understandable and meaning graphs [3].

was implemented to solve problems with massive data because it is capable of automatically extracting the important features from data. Furthermore, due to the fact that most of flight delay data are noisy, a technique based on stack denoising autoencoder is designed and added to the proposed model. A technique using Levenberg-Marquart algorithm was applied to find the weight and bias proper values resulting into an optimized and high accuracy results.

Due to Covid-19, airline businesses around the world suffer losses caused by very flights and passengers as required by many countries. The biggest income source of airline companies was suspended like the case of Thai Airways which filed for bankruptcy. A study [9] investigated on the demand for air travel and analyzed the competition in the aviation industry, and tried to determine the key factors of successful airline business. It uses KNN, Logistic Regression, Gaussian NB, Decision Trees, and Random Forest which will later be compared. The results of the study ranked Random Forest Algorithm as the highest in terms of accuracy using a threshold of 0.7 having an accuracy of 99%. The study also extracted the most important factor in getting customer satisfaction, which is Inflight Wi-Fi Services [9].

3.2 Algorithms for Predicting Flight Passenger Satisfaction

3.2.1Naive Bayes

The Naive Bayes algorithm is a classification algorithm based on the Bayes rule and a set of conditional independence assumptions [12]. Given the goal of learning P(Y|X) where X = hX$_1$ ..., X$_{ni}$, the Naive Bayes algorithm makes the assumption that each X$_i$ is conditionally independent of each of the other Xks given Y, and also independent of each subset of the other Xk's given Y. The value of this assumption is that it dramatically simplifies the representation of P(X|Y), and the problem of estimating it from the training data. Consider, for example, the case where X = hX$_1$, X^2_i. In this case,

$$\begin{aligned} P(X|Y) &= P(X_1, X_2|Y) \\ &= P(X_1|X_2, Y)P(X_2|Y) \\ &= P(X_1|Y)P(X_2|Y) \end{aligned}$$

where the second line follows from a general property of probabilities, and the third line follows directly from our above definition of conditional independence [12]. More generally, when X contains n attributes that satisfy the conditional independence assumption, we have

$$P(X_1 \ldots X_n|Y) = \prod_{i=1}^{n} P(X_i|Y)$$

Noticed that when Y and the X$_i$ are Boolean variables, we need only 2n parameters to define P(Xi = xik|Y = y$_j$) for the necessary i, j, k. This is a dramatic reduction compared to the 2(2 n −1) parameters needed to characterize P(X|Y) if we make no conditional independence assumption [12]. Let us now derive the Naive Bayes algorithm, assuming in general that Y is any discrete−valued variable, and the attributes X$_1$ ...X$_n$ are any discrete or real-

valued attributes. Our goal is to train a classifier that will output the probability distribution over possible values of Y, for each new instance X that we ask it to classify [12]. The expression for the probability that Y will take on its kth possible value, according to Bayes rule, is

$$P(Y = y_k|X_1 \ldots X_n) = \frac{P(Y = y_k)P(X_1 \ldots X_n|Y = y_k)}{\sum_j P(Y = y_j)P(X_1 \ldots X_n|Y = y_j)}$$

where the sum is taken over all possible values y j of Y. Now, assuming the X_i are conditionally independent given Y, we can use the equation (1) to rewrite this as

$$P(Y = y_k|X_1 \ldots X_n) = \frac{P(Y = y_k)\prod_i P(X_i|Y = y_k)}{\sum_j P(Y = y_j)\prod_i P(X_i|Y = y_j)}$$

is the fundamental equation for the Naive Bayes classifier. Given a new instance X new = hX1 ...Xni, this equation shows how to calculate the probability that Y will take on any given value, given the observed attribute values of X new and given the distributions P(Y) and P(Xi |Y) estimated from the training data. If we are interested only in the most probable value of Y, then we have the Naive Bayes classification rule:

$$Y \leftarrow \arg\max_{y_k} \frac{P(Y = y_k)\prod_i P(X_i|Y = y_k)}{\sum_j P(Y = y_j)\prod_i P(X_i|Y = y_j)}$$

which simplifies to the following (because the denominator does not depend on yk). Y ← argmax yk P(Y = yk)∏ i P(X_i |Y = yk)

$$Y \leftarrow \arg\max_{y_k} \; P(Y = y_k)\prod_i P(X_i|Y = y_k)$$

3.2.2 Logistic Regression

Logistic Regression is an approach to learning functions of the form f : X →Y, or P(Y|X) in the case where Y is discrete-valued, and X = hX1 ...Xni is any vector containing discrete or continuous variables [12]. In this section we will primarily consider the case where Y is a Boolean variable, in order to simplify notation. In the final subsection we extend our treatment to the case where Y takes on any finite number of discrete values [12]. Logistic Regression assumes a parametric form for the distribution P(Y|X), then directly estimates its parameters from the training data(figure 2).

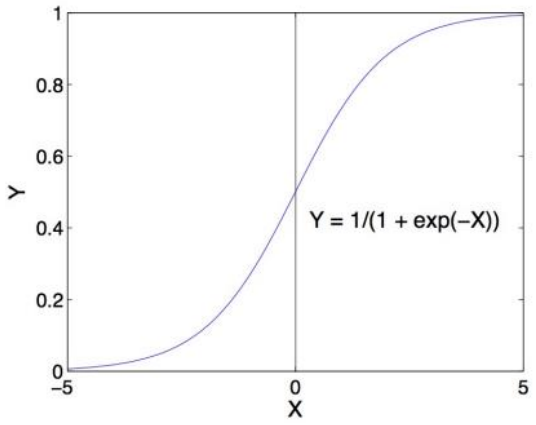


Figure 2. Form of the logistic function.

3.2.3 Deep Learning

The Deep Learning algorithm embedded in Rapid Miner Studio is deterministic only if the reproducible parameter is set to true. In this case, the algorithm uses only 1 thread. It is based on a multi-layer feed-forward artificial neural network that is trained with stochastic gradient descent using back-propagation. The network can contain a large number of hidden layers consisting of neurons with tanh, rectifier, and max out activation functions. Advanced features such as adaptive learning rate, rate annealing, momentum training, dropout, and L1 or L2 regularization enable high predictive accuracy. Each compute node trains a copy of the global model parameters on its local data with multi-threading (asynchronously), and contributes periodically to the global model via model averaging across the network [14].

The operator starts a 1-node local H2O cluster and runs the algorithm on it. Although it uses one node, the execution is parallel. You can set the level of parallelism by changing the Settings/Preferences/General/Number of threads setting. By default, it uses the recommended number of threads for the system. Only one instance of the

cluster is started and it remains running until you close RapidMiner Studio [14].

3.2.4 Support Vector Machine

The support vector machine (SVM) is a popular classification technique. However, beginners who are not familiar with SVM often get unsatisfactory results since they miss some easy but significant steps. Using 'm' numbers to represent an m-category attribute is recommended. Only one of the 'm' numbers is 1, the others are 0. For example, a three-category attribute such as Outlook {overcast, sunny, rain} can be represented as (0,0,1), (0,1,0), and (1,0,0). This can be achieved by setting the coding type parameter to 'dummy coding' in the Nominal to Numerical operator. Generally, if the number of values in an attribute is not too large, this coding might be more stable than using a single number.

To get a more accurate classification model from SVM, scaling is recommended. The main advantage of scaling is to avoid attributes in greater numeric ranges dominating those in smaller numeric ranges. Another advantage is to avoid numerical difficulties during the calculation. Because kernel values usually depend on the inner products of feature vectors, e.g. the linear kernel and the polynomial kernel, large attribute values might cause numerical problems. Scaling should be performed on both training and testing data sets. In this process the scale parameter is checked. Uncheck the scale parameter and run the process again. You will see that this time it takes a lot longer than the time taken with scaling.

You should have a good understanding of kernel types and different parameters associated with each kernel type in order to get better results from this operator. The gaussian combination kernel was used in this example process. All parameters were used with default values. The accuracy of this model was just 35.71%. Try changing different parameters to get better results. If you change the parameter C to 1 instead of 0, you will see that accuracy of the model rises to 64.29%. Thus, you can see how making small changes in parameters can have a significant effect on overall results. Thus, it is very necessary to have a good understanding of parameters of kernel type in use. It is equally important to have a good understanding of different kernel types, and choosing the most suitable kernel type for your ExampleSet. Try using the polynomial kernel in this Example Process (also set the parameter C to 0); you will see that accuracy is around 71.43% with default values for all parameters. Change the value of the parameter C to 1 instead of 0. Doing this increased the accuracy of the model with Gaussian combination kernel, but here you will see that the accuracy of the model drops.

We used default values for most of the parameters. To get more accurate results these values should be carefully selected. Usually, techniques like cross-validation are used to find the best values of these parameters for the ExampleSet under consideration.

3.2.5 Decision Trees

Decision tree is a hierarchical data structure that represents data through a divide-and-conquer strategy. In classification, the goal is to learn a decision tree that represents the training data such that labels for new examples can be determined. Decision trees are classifiers for instances represented as feature vectors (e.g. color=?; shape=?; label=?;). Nodes are tests for feature values, leaves specify the label, and at each node there must be one branch for each value of the feature.[15]

When building a decision tree, the goal is to produce as small of a decision tree as possible. However, finding a minimal decision tree that is consistent with the data is NP−hard. We thus want to use heuristics that will produce a small tree, but not necessarily the minimal tree. Consider the following data with two Boolean attributes (A,B), which attribute should we choose as root?: < (A = 0, B = 0), − >: 50 examples < (A = 0, B = 1), − >: 50 examples < (A = 1, B = 0), − >: 0 examples < (A = 1, B = 1), + >: 100 examples If we split on A, we get purely labeled nodes. If A=1, the label is +, − otherwise. If we split on B, we do not get purely labeled nodes. However if we change the number of (A = 1, B = 0) from 0 to 3, we will no longer have purely labeled nodes after splitting on A. Choosing A or B gives us the following two trees

3.2.6 Random Forest

As with nonparametric regression, simple and interpretable classifiers can be derived by partitioning the range of X. Let Πn = {A1, . . . , AN } be a partition of X . Let Aj be the partition element that contains x. Then bh(x) = 1 if P Xi∈Aj Yi ≥ P Xi∈Aj (1 − Yi) and bh(x) = 0 otherwise. This is nothing other than the plugin classifier based on the partition regression estimator [16]

$$\widehat{m}(x) = \sum_{j=1}^{N} \overline{Y}_j \, I(x \in A_j)$$

where $\overline{Y}_j = n_j^{-1} \sum_{i=1} Y_i I(X_i \in A_j)$ is the average of the Y_i's in A_j and $n_j = \#\{X_i \in A_j\}$. (We define $\overline{Y}_j$ to be 0 if $n_j = 0$.) Recall from the results on regression that if $m \in H^1(1, L)$ and the binwidth b of a regular partition satisfies $b \asymp n^{-1/(d+2)}$ then

$$\mathbb{E}||\widehat{m} - m||_P^2 \leq \frac{c}{n^{2/(d+2)}}.$$

We conclude that the corresponding classification risk satisfies $R(bh) - R(h^*) = O(n^{-1/(d+2)})$. Regression trees and classification trees (also called decision trees) are partition classifiers where the partition is built recursively.

For continuous Y (regression), the split is chosen to minimize the training error. For binary Y (classification), the split is chosen to minimize a surrogate for classification error. A common choice is the impurity defined by I(t) = P2 s=1 γs where

$$\gamma_s = 1 - [\overline{Y}_s^2 + (1 - \overline{Y}_s)^2].$$

This particular measure of impurity is known as the Gini index. If a partition element as contains all 0's or all 1's, then γs = 0. Otherwise, γs > 0. We choose the split point t to minimize the impurity. Other indices of impurity besides the Gini index can be used, such as entropy. The reason for using impurity rather than classification error is because impurity is a smooth function and hence is easy to minimize. Now we continue recursively splitting until some stopping criterion is met. For example, we might stop when every partition element has fewer than n0 data points, where n0 is some fixed number. The bottom nodes of the tree are called the leaves. Each leaf has an estimate mb (x) which is the mean of Yi 's in that leaf. For classification, we take bh(x) = I(mb (x) > 1/2). When there are several covariates, we choose whichever covariate and split that leads to the lowest impurity. The result is a piecewise constant estimator that can be represented as a tree.

3.3 Simulation and Testing of Algorithms

Rapid Miner Studio was used as simulation and data analysis platform to test, analyze and design predictive algorithms. The simulation is composed four main parts, namely: Data Preparation, Data Mining, Evaluation, Visualization.

3.3.1 Data Preparation

Description of Datasets

The dataset used in this study is an open dataset from kaggle.com, specifically Invistico_Airline data with 129,880 rows and 22 columns. The column names are described as follows: satisfaction, Customer Type, Age, Type of Travel, Class, Flight Distance, Seat comfort, Departure/Arrival time convenient, Food and drink, Gate location, Inflight Wi-Fi service, Inflight entertainment, Online support, Ease of Online booking, On-board service, Leg room service, Baggage handling, Check-in service, Cleanliness, Online boarding, Departure Delay in Minutes, Arrival Delay in Minutes.

Data preparation was first conducted through feature selection, data cleaning and normalization. After data preparation, an Auto Modeller in Rapidminer was used to simulate and test different algorithms. It involves selection, data preprocessing, determining class of higher interest, mapping classes to new values and transformation.

3.3.2 Selection

After starting Auto Model, the first step is selecting “invistico_airline.csv” dataset from the repositories.

3.3.3 Data Preprocessing

Since "Satisfaction" has only two values, "Yes" or "No", we consider it as a classification problem. Auto Model displayed a bar chart with only the 10 classes with the most data points.

3.3.3.1 Class of Highest Interest

In the presentation of results, Class of Highest Interest is important because performance values such as "Precision" and "Recall" depend on knowing which of the classes should be interpreted as a "positive" result. In our experiment on the Flight Passenger datasets, the Class of Highest Interest in “YES”.

3.3.4 Transformation

Transformation of raw dataset helps to speed up the process of modelling and improve the performance of algorithms. Some of the columns in our dataset may not help to make prediction and key point we are looking for is the common patterns that may contribute to good prediction of passenger satisfaction. Transforming data such as dimensionality reduction which reduces the size of data but retaining the patterns is good practice for data pre-processing.

In transforming the datasets, the following data were removed and thus our dataset was cleansed:

- Columns that too closely mirror the target column, or not at all (Correlation),
- Columns where nearly all values are different (ID-ness),
- Columns where nearly all values are identical (Stability),
- Columns with missing values (Missing).
- Automatic Feature Selection
- Automatic Feature Generation

3.3.5 Data Mining

Rapid miner Studio has selections of data mining algorithms that can be used to design and simulate prediction models. It also has Auto Model features which suggest algorithms fit for type of data mining tasks. It can perform prediction, cluster detection and outlier detection as its default capabilities. Based on our problem domain, our main task is prediction problem which means we need to use classification algorithms to design prediction models.

In training the Flight Passenger data, the following algorithms were used:

- Naive Bayes
- Generalized Linear Model
- Logistic Regression
- Deep Learning
- Decision Tree
- Random Forest

3.3.6 Evaluation and Performance Measures

During the simulation, the algorithms were trained along another data components such as classification error, accuracy, Precision, Recall, F-measure, sensitivity, specificity, and ROC-AUC comparison [13].

Classification error -relative number of misclassified examples or in other words percentage of incorrect predictions.

Accuracy - *t*he accuracy is calculated by taking the percentage of correct predictions over the total number of examples. Correct prediction means the examples where the value of the prediction attribute is equal to the value of label attribute.

Recall - *t*he weighted mean recall is calculated by taking the average of recall of every class.

Precision -the weighted mean precision is calculated by taking the average of precision of every class.

F-measure - in statistical analysis of binary classification and information retrieval systems, the F-score or F-measure is a measure of predictive performance. It is calculated from the precision and recall of the test, where the precision is the number of true positive results divided by the number of all samples predicted to be positive, including those not identified correctly, and the recall is the number of true positive results divided by the number of all samples that should have been identified as positive. Precision is also known as positive predictive value, and recall is also known as sensitivity in diagnostic binary classification.

TP = True Positive = Correctly predicted Positive class
TN = True Negative = Correctly predicted Negative class

FP = False Positive = Predicted positive, but actually negative

FN = False Negative = Predicted negative, but actually Positive

PPV and precision = TP / (TP + FP) *Positive Predicted Value*

NPV = TN / (TN + FN) *Negative Predicted Value*

sensitivity and recall = TP / (TP + FN)

specificity = TN / (TN + FP)

F Measure F1 = 2TP/(2TP + FP + FN) An equal balance of precision and recall.

AUC (**Area Under The Curve**) ROC (**Receiver Operating Characteristics**) curve. It is one of the most important evaluation metrics for checking any classification model's performance. It is also written as AUROC (**Area Under the Receiver Operating Characteristics**)

UC - ROC curve is a performance measurement for the classification problems at various threshold settings. ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. Higher the AUC, the better the model is at predicting 0 classes as 0 and 1 classes as 1. By analogy, the Higher the AUC, the better the model is at distinguishing between patients with the disease and no disease.

AUC is the Area Under the Curve of the ROC -which is created when TP rate is plotted against the FP rate across the thresholds. For a given threshold the TP rate is like the PPV at that threshold and the FP rate is like the NPV at that threshold.

3.3.7 Description of Dataset

The datasets used in this paper comes from a public kaggle database. It is sourced from Invistico_Airline. It has 129,880 and: 22 columns. The attributes/columns are described below:

Satisfaction, Customer Type, Age, Type of Travel, Class, Flight Distance, Seat comfort, Departure/Arrival time convenient, Food and drink, Gate location, Inflight Wi-Fi service, Inflight entertainment, Online support, Ease of Online booking, On-board service, Leg room service, Baggage handling, Check-in service, Cleanliness, Online boarding, Departure Delay in Minutes, Arrival Delay in Minutes.

3.4 Results

This is section discuss the results of the predictive algorithms tested in this study. The main task is to determine which of the algorithms has highest accuracy and the best performance.

3.4.1 Accuracy of Algorithms

The accuracy of predictive algorithms is expressed in terms of the collective results of percentage of training accuracy, gains, total time and training time. As depicted in table 1, Deep Learning has the highest percentage of accuracy at 93.0% and

highest gain of 28, 406. The total time it takes for Deep Learning to train, test and build the model is 13 minutes and 34 seconds which includes the training time of 21 milliseconds.

Table 1. Summary of Accuracy of Algorithms Tested

Model	Accuracy	Gains	Total Time	Training Time
Naive Bayes	80.8%	19,320	16 mins 17 s	3 ms
Decision Tree	89.4%	25,616	7 mins 54 s	2 ms
Logistic Regression	82.4%	20,510	8 mins 37 s	6 ms
Deep learning	93.0%	28,406	13 mins 34 s	21 ms
Random Forest	89.8%	12,526	34 min 36 s	323 ms
Support Vector Machine	54.8%	10	31 min 49 s	32s

Gain is used to measure how much better our prediction model compared without the model. It evaluates the model prediction and the benefit to the business. It visualized the performance of a model in terms of benefits we could get in using the model in a portion of the population.

Deep Learning, having the largest Gain value means that we can visualized to cover a greater area between gain and baseline population.

3.4.2 Performance of Algorithms

Table 2 and 3 depicts the values corresponding the performance measures of a predictive algorithms, specifically in terms of classification error, AUC(ROC), Precision, Recall, F-measure, sensitivity, and specificity. It is evident that the values corresponding to Deep Learning proved a good performance with the lowest classification error of 7.0% which means that among all algorithms tested, Deep learning has the smallest relative numbers of misclassified examples or in other words percentage of incorrect predictions. In terms of AUC(ROC) which is one of the most important evaluation metrics for checking any classification model's performance, 99.5% is a remarkable performance because the higher the AUC, the better the model is at predicting 0 classes as 0 and 1 classes as 1. By analogy, the higher the AUC, the better the model is at distinguishing between a satisfied passenger and a dissatisfied passenger.

Furthermore, precision of 98.8%, recall at 88.5% and F-measure at 93.3% is an indication of best performance of Deep Learning. This value represents the weighted mean precision calculated by taking the average of precision of every class, the weighted mean recall is calculated by taking the average of recall of every class and a measure of predictive performance respectively.

Evaluating the sensitivity measure of Deep Learning at 88.5% is the proportion of actual positive cases (satisfied passengers), which got predicted correctly, while specificity at 98.5% is the proportion of actual negative cases (dissatisfied passengers), which got predicted correctly.

Table 2. Performance Table

Algorithm	Classification Error	AUC	Precision	Recall	Sensitivity	Specificity	F-Measure
Naive Bayes	19.2%	88.9%	86.7%	76.7%	78.4%	85.5%	81.4%
Logistic Regression	17.6%	90.3%	91.4%	74.8%	74.8%	91.5%	82.3%
Deep Learning	7.0%	99.5%	98.8%	88.5%	88.5%	98.5%	93.3%
Decision Tree	10.6%	92.5%	92.0%	88.2%	88.2%	90.7%	90.1%
Random Forest	10.2%	95.8%	90.6%	90.7%	90.7%	88.6%	90.6%
Support Vector Machine	45.2%	55.2%	54.8%	100%	100%	0.3%	70.8%

As discussed above, accuracy measures present the total number of correct classifications divided by the total number of cases. However, using this metric as a standalone comes with limitations, such as working with imbalance data and error types. Through these limitations, the confusion matrix in

table 3 described a more detailed insight on how to improve a model's performance.

Table 3. Confusion Matrix

Algorithm	Predicted Dissatisfied			Predicted Satisfied			Class Recall	
	True Dissatisfied	True Satisfied	Class Precision	True Dissatisfied	True Satisfied	Class Precision	True Dissatisfied	True Satisfied
Naive Bayes	14397	4737	75.24%	2391	15584	86.70%	85.76%	76.69%
Logistic Regression	15365	5110	75.04%	1433	15201	91.39%	91.47%	74.84%
Deep Learning	16540	2337	87.62%	258	17974	98.58%	98.40%	88.40%
Decision Tree	15203	2395	88.30%	1552	17959	92.064 %	90.74%	88.23%
Random Forest	7167	904	88.80%	924	8862	90.58%	88.58%	90.74%
Support Vector Machine	0	1	86.71%	1935	2343	54.77%	0.31%	99.96%

3.4.3 Prediction Model for Airline Passenger Satisfaction using Deep Learning

Deep Learning Model

```
Model Metrics Type: Binomial
 Description: Metrics reported on temporary training frame with 9894 samples
 model id: rm-h2o-model-model-1
 frame id: rm-h2o-frame-model-1.temporary.sample.14.26%
 MSE: 0.033220917
 RMSE: 0.18226606
 R^2: 0.8659472
 AUC: 0.9924613
 pr_auc: 0.994314
 logloss: 0.10981998
 mean_per_class_error: 0.044685278
 default threshold: 0.4449719190597534
```

Figure 2. Deep Learning Model for Flight Passenger Prediction

Figure 2 shows the Deep Learning Model for Prediction Airline Passenger Satisfaction. A binary prediction model was designed using 9894 samples for temporary training frame.

MSE is an absolute value unique to the Invistico Airline Passenger Dataset. The value of **0.033220917,** for MSE, being close to 0, means that the model has become more accurate compared to previous experiments found in the literature.

RMSE should be low as possible. A value of **0.18226606** for RMSE means it is closer to zero that can be interpreted to be a good model.

R-squared is not a measure of how accurate the prediction is, but it is a measure of fit. The higher the R-squared value, the better. The model shows that the value of **R2= 0.8659472** implies that there is a significant amount of variance of the independent variable is explained by the independent variable of this regression model. It represents strong correlation for most use case.

The probability that the model will assign a larger probability to a random positive example than a random negative example is **0.9924613.** This AUC value is interpreted as very good performance and accuracy that more likely the correct predictions will be made.

To optimize machine learning algorithms, the **mean_per_class error value of 0.044685278** indicates a good performance and represents the average of errors of each class in a muti-class dataset. This speaks towards low chances of misclassification of the data across classes.

A decision threshold of 0.4444719190597534 is a good indication could balance the precision value and value of recall.

Table 4. Important Factors for Passenger Satisfaction and Dissatisfaction

Type of Travel/Type of Passenger	% Satisfied	% Dissatisfied	Factors for Satisfaction	Factors for Dissatisfaction
Business/Loyal Customer	43%	57%	• Customer Type • Departure Delay in Minutes • Baggage Handling • Class • Ease of Online Booking • Inflight Entertainment	Arrival Delay in Minutes
Economy/Loyal Customer	10%	90%	• Customer Type • Departure Delay in Minutes • Baggage Handling • Class • Ease of Online Booking • Inflight Entertainment	Arrival Delay in Minutes
Economy Plus/Loyal Customer	9%	91%	• Customer Type • Departure Delay in Minutes • Baggage Handling • Class • Ease of Online Booking • Inflight Entertainment	Arrival Delay in Minutes
Business/Disloyal Customer	7%	93%	• Departure Delay in Minutes • Baggage Handling • Class • Ease of Online Booking • Inflight Entertainment	-Arrival Delay in Minutes -Customer Type

Deep Learning clearly identified that Customer Type is the most important factor in passenger satisfaction. Business/Loyal Customer has the highest level of satisfaction while business/disloyal customer has the highest level of dissatisfaction. Other factors for satisfaction are Departure Delay Time, Baggage Handling, Ease of Online Booking, and Inflight Entertainment. The

factor for passenger dissatisfaction is Arrival Delay in Minutes.

Satisfaction Model for Business Class/Loyal Customer

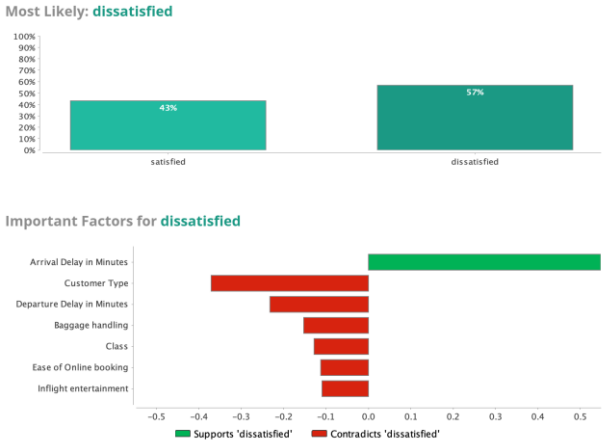
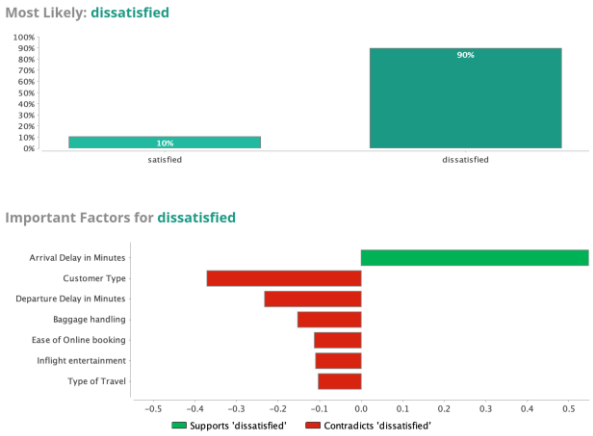


Figure 3. Satisfaction Model for Business Class/Loyal Customer

Figure 3 shows that business class and loyal passengers has 57% probability of dissatisfaction. The lower bar chart explains what is against satisfaction, above all, it is Arrival Delay in Minutes, displayed as green bar. Green in this context implies that Arrival Delay in Minutes agree with the prediction of Satisfaction, namely "No". The red bars for Customer Type, Departure Delay in Minutes, Baggage Handling, Class, Ease of Online Booking and Inflight Entertainment imply a disagreement with the prediction, and hence a positive correlation with Satisfaction.

Satisfaction Model for Economy Class/Loyal Passenger



Fiure 4. Satisfaction Model for Economy Class/Loyal Passenger

Figure 4 shows that economy class and loyal passengers has 90% probability of dissatisfaction. The lower bar chart explains what is against satisfaction, above all, it is Arrival Delay in Minutes, displayed as green bar. Green in this context implies that Arrival Delay in Minutes agree with the prediction of satisfaction, namely "No". The red bars for Customer Type, Departure Delay in Minutes, Baggage Handling, Class, Ease of Online Booking, Inflight Entertainment and Type of Travel imply a disagreement with the prediction, and hence a positive correlation with Satisfaction.

Satisfaction Model for Economy Plus/Loyal Customer

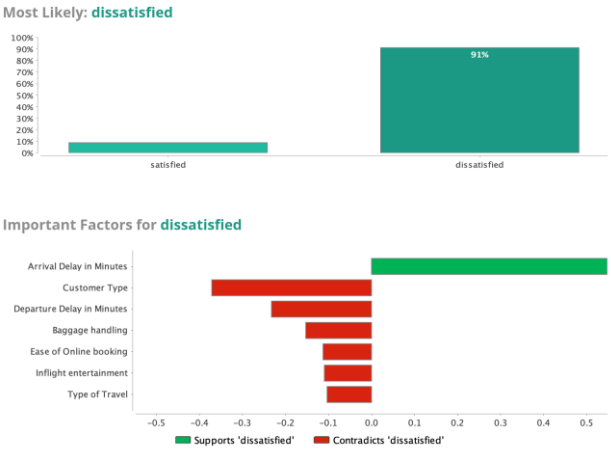


Figure 5. Satisfaction Model for Economy Plus/Loyal Passenger

Figure 5 shows that economy plus and loyal passengers has 91% probability of dissatisfaction. The lower bar chart explains what is against satisfaction, above all, it is Arrival Delay in Minutes, displayed as green bar. Green in this context implies that Arrival Delay in Minutes agree with the prediction of satisfaction, namely "No". The red bars for Customer Type, Departure Delay in Minutes, Baggage Handling, Class, Ease of Online Booking, Inflight Entertainment and Type of Travel imply a disagreement with the prediction, and hence a positive correlation with Satisfaction.

Satisfaction Model for Business Class/Disloyal Customer

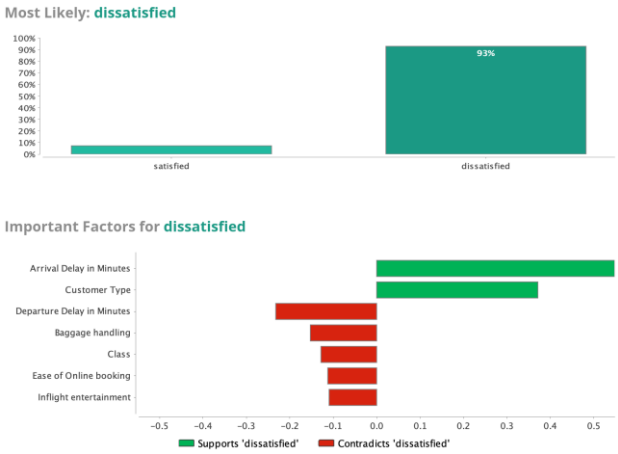


Figure 6. Satisfaction Model for Business Class/Disloyal Passenger

Figure 6 shows that business class and disloyal passengers has 93% probability of dissatisfaction. The lower bar chart explains what is against satisfaction, above all, it is Arrival Delay in Minutes and Customer Type displayed as green bars. Green in this context implies that Arrival Delay in Minutes agree with the prediction of satisfaction, namely "No". The red bars for Departure Delay in Minutes, Baggage Handling, Class, Ease of Online Booking, and Inflight Entertainment imply a disagreement with the prediction, and hence a positive correlation with Satisfaction.

4 Conclusion

With the dataset used in this experiment, we can conclude that Deep Learning has the best performance for predicting inflight passenger satisfaction with the highest accuracy among 6 commonly used prediction algorithms with 93.0% accuracy and the largest gain value of 28,406 that cover a greater area between gain and baseline population. It has the best performance at predicting 0 classes as 0 and 1 classes as 1 in distinguishing between a satisfied passenger and a dissatisfied passenger with its AUC(ROC) 99.5%.

Deep learning has the smallest relative numbers of misclassified examples or in other words percentage of incorrect predictions represented by its classification error of 7.0%. It predicted 88.5% of actual satisfied passengers and 98.5% dissatisfied passengers.

Using Deep Learning, four models were designed for predicting flight passenger satisfaction namely (1) business class /loyal customer model (2) economy class and loyal passengers, (3) economy plus/loyal customer, and (4) business class/disloyal customer model. Arrival Delay in Minutes and Customer Type causes dissatisfaction for all types of passengers and types of class. Departure Delay Time, Baggage Handling, Ease of Online Booking, and Inflight Entertainment are the predictors for flight satisfaction.

Such prediction models were evaluated as more accurate compared to previous experiments with an MSE of 0.033220917, RMSE of 0.18226606, R-square of 0.8659472 and AUC of 0.9924613.

Acknowledgement:

I would like to express my deepest appreciation to my family, ONA family and most especially to my husband, Ronwald Gonzalo and my son, Roald Johann. Also, my warmest thanks to my adviser Dr. Menchita F. Dumlao, without her guidance and expertise, this paper will not be possible. To my colleagues at PHILSCA, Ma'am Divine, Ma'am Elden, Jinky G. and to my close friends Shella and Tess D., thank you for the moral support. The belief of my family and friends has kept my spirits and motivation high during this process.

References:

[1] Kashyap, Ramgopal . 2019. "(PDF) Artificial Intelligence Systems in Aviation." ResearchGate. January 2019. https://www.researchgate.net/publication/330573828_Artificial_Intelligence_Systems_in_Aviation.

[2] Jiang, Xuchu, Ying Zhang, Ying Li, and Biao Zhang. 2022. "Forecast and Analysis of Aircraft Passenger Satisfaction Based on RF-RFE-LR Model." *Scientific Reports* 12 (1). https://doi.org/10.1038/s41598-022-14566-3.

[3] Nwagu, Chikezie Kenneth, Omankwu, Obinnaya Chinecherem, and Inyiama, Hycient. Knowledge Discovery in Databases (KDD): An Overview. 2017.
International Journal of Computer Science and Information Security (IJCSIS)
Vol. 15, No. 12.

[4] M. Ali and F. Esposito (Eds.): IEA/AIE 2005, LNAI 3533, pp. 1–5, 2005.Springer-Verlag Berlin Heidelberg.

[5] Gu, Yunyan, Jianhua Yang, Conghui Wang, and Guo Xie. 2020. "Early Warning Model for Passenger Disturbance due to Flight Delays." Edited by Qiang Zeng. *PLOS ONE* 15 (9): e0239141. https://doi.org/10.1371/journal.pone.0239141.

[6] Dr.Sumitha.K, and Mr.Santhosh.K.V. 2023. "'Exploring the Role of Artificial Intelligence in Improving Passenger Satisfaction in the Airline Industry: An Analysis of Customer Feedback and AI-Driven Solutions.,'" March. https://doi.org/10.5281/zenodo.7827975.

[7] Hong, A.C.Y., Khaw,K.W., Chew, X.Y., and Yeong, W.C. 2023. Prediction of US airline passenger satisfaction using machine learning algorithms. DATA ANALYTICS AND APPLIED MATHEMATICS.VOLUME 4, ISSUE 1, 2023, 8 – 24. E-ISSN: 2773-4854.
DOI: https://doi.org/10.15282/daam.v4i1.9071.

[8] Yazdi, M.F., Kamel, S.R., Chabok, S.J.M. *et al.* Flight delay prediction based on deep learning and Levenberg-Marquart algorithm. *J Big Data* 7, 106 (2020). https://doi.org/10.1186/s40537-020-00380-z

[9] Hulliyah, Khodijah. (2021). Predicting Airline Passenger Satisfaction with Classification Algorithms. IJIIS: International Journal of Informatics and Information Systems4(1), 82-94. 10.47738/ijiis.v4i1.80.

[10] Noviantoro, T., Huang, J.P. 2022. Investigating airline passenger satisfaction: Data mining method.Research in Transportation and Business Management. From https://doi.org/10.1016/j.rtbm.2021.100726

[11] Cen Song, Xiaoqian Ma, Catherine Ardizzone, Jun Zhuang, The adverse impact of flight delays on passenger satisfaction: An innovative prediction model utilizing wide & deep learning, Journal of Air Transport Management,Volume114,2024,102511, ISSN0969-6997 from https://doi.org/10.1016/j.jairtraman.2023.102511.

[12] Mitchell, Tom M. 2017. Book Chapter 3 Generative and Discriminant Classifiers: Naive Bayes and Logistic Regression.

[13] Gadi, Paulo and Tagliaferri, Roberto. 2018. Data Mining: Accuracy and Error Measures for Classification and Predictions. Elsevier Preprint.
DOI:10.1016/B978-0-12-809633-8.20474-3

[14] Rapidminer Documentation Manual. 2024. Deep Learning Synopsis.

[15] Roth, Dan. 2016. Lecture Notes in CS 446 Machine Learning Fall 2016. From https://www.cis.upenn.edu/~danroth/Teaching/CS446-17/LectureNotesNew/dtree/main.pdf.

[16] Biau, Gérard, and Erwan Scornet. 2016. "A Random Forest Guided Tour." *TEST* 25 (2): 197–227. https://doi.org/10.1007/s11749-016-0481-7.