F21DL

Data Mining and Machine Learning

GROUP COURSEWORK

Airline Survey Analysis

Done By:

Ajay Kumar Paidimarri - H00467552

Anugraha Antoo Kanjookaran - H00460688

Fiyifoluwa Shedrach Onafeso - H00466745

Hariharakumar Rathinar - H00463082

Syeda Mariam Danish - H00473375

GITHUB LINK: https://github.com/HWhr3000/F21DL_Coursework_grp2

TABLE OF CONTENTS

INTRODUCTION

This report is about a machine learning model that aims to predict customer loyalty in the airline industry. A predictive model which would assess the probability of loyalty of a passenger based on his very own response on the quality aspects of various services. The information obtained through this analysis can back strategic initiatives that target service areas responsible for loyalty and retention. At its core, the report aims to provide insights into customer loyalty trends and guide airlines on their journey toward improving customer retention and service offerings overall.

A unique feature of this study is its innovative use of image analysis. By analyzing pictures of passengers completing the survey, the research investigates how visual cues or facial expressions might affect loyalty outcomes. This approach offers fresh perspectives that go beyond the typical methods of data analysis.

RELATED WORK

Machine learning has gained significant momentum in recent decades for predicting customer loyalty, offering valuable insights into consumer behavior for businesses. Traditional methods focus on analyzing demographic and transactional data to identify patterns and make predictions. Increasingly more advanced techniques such as ensemble-based learning and neural networks are recently showing great promise in terms of improving prediction accuracy (Fritz.ai, n.d.). Customer segmentation plays a crucial role in loyalty prediction. Predictive analytics is another powerful application of machine learning in customer loyalty prediction. These algorithms can forecast future customer behavior, including purchase patterns, churn risk, and engagement levels (Build with Toki, 2023). Recent studies have incorporated deep learning models, such as neural networks, to capture non-linear relationships and complex patterns in customer data (Jadhav and Patil, 2020) (Amiri, 2021). As the field progresses, researchers are also investigating how to incorporate varying data sources, among them visual data and facial expressions, in customer loyalty predictions (Hamdan et al., 2023). This multifaceted approach could provide a comprehensive understanding of customer loyalty by combining behavioral data with emotional and visual cues to create accurate and insightful prediction models.

R2 DATASET DESCRIPTION AND ANALYSIS

Dataset 1 is excel data. The input CSV file contains 25 columns of customer data with variables capturing demographic details and service experience ratings across multiple aspects aimed to predict customer loyalty i.e segmented into four **Loyalty** classes.

2.1 Data Quality and Missing Values

We evaluated data quality and identified minor missing values: Arrival Delay (0.3%), Ease of Online Booking (4.37%), and Departure/Arrival Time Convenience (5.1%). Continuous features like delays and satisfaction metrics were filled using the median to handle gaps and outliers effectively. Categorical variables, including Gender, Customer Type, and Travel Type, were label-encoded for analysis.

2.2 Feature Engineering

Combining multiple feature selection techniques provides a more holistic view of feature importance. The binning of age and flight distance did not significantly enhance feature relevance, indicating the need for a more nuanced approach to numerical variables.

Feature / Rank	TC	Correlation	Kmean	Chi2Score	Lasso	RFE	Overall rank
Type of Travel	1	7	1	3	2	2	3
Online Boarding	2	4	2	7	1	6	4
In-flight Wifi Service	3	17	3	6	14	1	7
Ease of Online Booking	4	18	4	8	13	4	9
Age	7	3	8	2	4	18	7
In-flight Entertainment	8	6	9	11	5	7	8

The table above is a sample of top 5 features derived based on combined ranks from various feature selection methods from which top 13 features are selected. New critical features like Type of travel were selected based on this approach along with a decision to eliminate features like Departure Delay and Arrival Delay which showed limited utility across ranking methods.

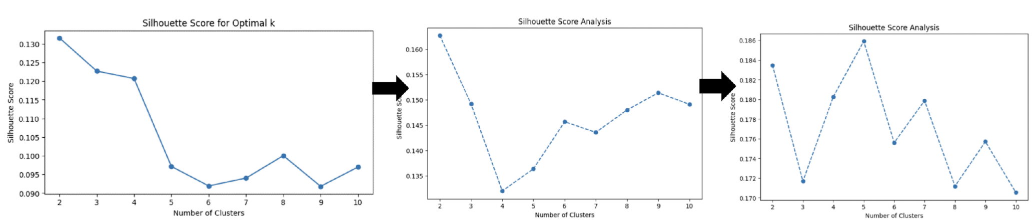
2.3 Image Dataset Description

The project used the Kaggle Face Emotions Recognition Dataset with 7 emotion classes: Angry, Disgust, Fear, Happy, Neutral, Sad, and Surprise. The dataset included 28,821 training images and 7,066 validation images. Class imbalance was addressed with weighted loss functions and augmentation.

EXPERIMENTAL SETUP AND RESULTS

R3 – CLUSTERING

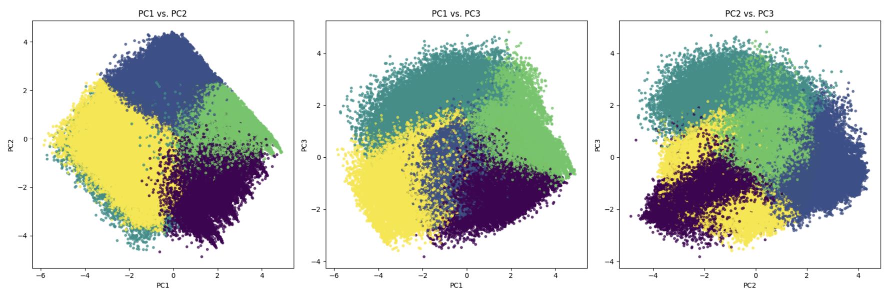
Feature selection improved clustering by isolating key attributes such as Type of Travel, Online Boarding, and In-flight Wifi Service, which had strong relevance to Loyalty. Removing redundant or uncorrelated features reduced noise and enhanced computational efficiency.



PCA further optimized the dataset by reducing the selected 14 features to eight principal components, capturing over 80% of the variance, a critical threshold for retaining the dataset's informational value. This dimensionality reduction not only improved clustering performance but also enabled clearer visualizations, such as scatter plots, to highlight meaningful cluster structures. For instance, PCA-reduced data allowed K-Means to achieve a peak silhouette score with five well-separated clusters, demonstrating the effectiveness of the process.

Clustering Methods

Among clustering methods, K-Means provided the most distinct and interpretable results, forming five well-separated clusters validated by the silhouette score peaking at five clusters. Using PCA-reduced data, scatter plots such as PC2 vs. PC3 revealed clear separations between groups, including loyal business travelers, dissatisfied economy travelers, and younger customers with moderate satisfaction.



GMM, assuming Gaussian distribution, struggled with overlapping clusters and failed to achieve a meaningful silhouette score peak, merging medium-satisfaction groups. Hierarchical clustering provided relationship insights via dendrograms but lacked the well-defined separations achieved by K-Means. Ultimately, K-Means demonstrated superior performance in creating actionable clusters, emphasizing the importance of dimensionality reduction and robust evaluation.

R4 - BASELINE TRAINING AND EVALUATION EXPERIMENTS

4.1 Bayes Net Algorithm

Naive Bayes Algorithm	**Accuracy (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**Recall (%)**	**ROC AUC**
Multinomial Naive Bayes	49.42	49.34	83.13	53.64	49.42	71.17
Gaussian Naive Bayes	78.72	78.71	92.91	80.22	78.72	94.59
Complement Naive Bayes	46.68	46.53	82.20	56.81	46.68	72.10
Bernoulli Naive Bayes	50.38	50.33	83.43	45.12	50.38	73.23
Categorical Naive Bayes	**82.37**	**82.37**	**94.13**	**82.86**	**82.37**	**95.47**

Best Performing Algorithm: Categorical Naive Bayes

4.2 Decision Tree

Tree Configuration	Training Accuracy	Test Accuracy	Precision	Recall	F1 Score	ROC AUC
Tree 1	0.76	0.75	0.77	0.76	0.76	0.88
Tree 2	0.76	0.76	0.77	0.76	0.75	0.88
Tree 3	0.76	0.76	0.76	0.76	0.75	0.88

The Decision Tree is with best parameters from Grid Search results. We observed that Tree 1 balances precision, recall, and F1 score better but has slightly lower accuracy. Tree 2 matches Tree 3 in accuracy but offers better precision and recall. Tree 3 lags in precision and F1, making it less optimal. For balanced precision and recall, Tree 1 is preferred. For tasks where balancing precision and recall is critical—such as minimizing both false positives and false negatives—we would likely prefer Tree 1 as the best option.

4.3 Logistic Regression

Solver	Max Iter	Batch Size	C	Accuracy	Precision	Recall	F1-Score	Observation
lbfgs	300	16	Default	81%	81.38%	81.18%	81.17%	Stable performance with default regularization and iterations; converges quickly.
lbfgs	1000	32	Default	81%	81.38%	81.18%	81.17%	No significant change in performance with increased iterations.
lbfgs	2000	64	Default	81%	81.38%	81.18%	81.17%	Larger batch size and more iterations did not significantly impact accuracy.
saga	300	16	0.1	81%	81.37%	81.18%	81.16%	Reduced regularization (C=0.1) yielded stable performance; effective for large datasets.
saga	1000	32	1	81%	81.37%	81.18%	81.16%	Default regularization (C=1) and more iterations maintained stability.
saga	2000	64	10	81%	81.37%	81.18%	81.16%	Increased regularization (C=10) did not negatively impact performance.
newton-cg	500	32	Default	81%	81.37%	81.18%	81.16%	Performs well with larger datasets; requires slightly more iterations to converge.
newton-cg	1500	64	1	81%	81.37%	81.18%	81.16%	Stable performance across increased batch sizes and regularization.
newton-cg	2000	16	0.5	81%	81.37%	81.18%	81.16%	Lower regularization (C=0.5) did not yield any noticeable improvement in accuracy.

R5 - NEURAL NETWORKS

5.1 Classification using Perceptron and MLP (Tabular data)

Analysis with 10-Fold Cross-Validation and Evaluation Metrics: The selected feature model performs, in comparison, much more evenly with less variance across classes, particularly with respect to precision and recall score against class 2.

Class	Precision	Recall	F1-Score	Support
1	0.89	0.91	0.9	10,455
2	0.9	0.91	0.9	10,659
3	0.93	0.93	0.93	10,561
4	0.92	0.88	0.9	10,572
Accuracy			**0.91**	42,247
Macro Avg	0.91	0.91	0.91	42,247
Weighted	0.91	0.91	0.91	42,247

Hyperparameter Tuning

Architecture	Accuracy	Sensitivity	Specificity	Precision	Recall	AUC	Number of Layers	Learning Rate	Iterations	Optimization Algorithm	Activation Functions	Layers (N × M × ...)
1	0.8643	0.8643	0.9334	0.865	0.8643	0.975	3	0.001	20	SGD	ReLU	10 × 32 × 32 × 4
2	0.9271	0.9271	0.9417	0.9271	0.9271	0.9931	3	0.001	20	SGD	ReLU	10 × 64 × 64 × 4
3	0.9133	0.9133	0.9409	0.9133	0.9133	0.9897	3	0.01	15	SGD	Tanh	10 × 128 × 128 × 4
4	0.9314	0.9314	0.9572	0.9315	0.9314	0.9939	3	0.001	10	Adam	ReLU	10 × 256 × 256 × 4
5	0.8938	0.8938	0.938	0.8942	0.8938	0.9838	3	0.001	20	SGD	ReLU	10 × 512 × 512 × 4
6	0.9387	0.9387	0.9628	0.9389	0.9387	0.9942	3	0.001	15	Adam	ReLU	10 × 1024 × 1024 × 4

Architecture 6 provides the best balance of high accuracy, specificity, sensitivity, and precision.

Performance Evaluation of MLP Classifiers with Varying Training Data Splits: Classifier 1 generalizes well. This means the model is likely to maintain similar performance when applied to new data**.**

Classifier	Train Accuracy	Test Accuracy	Precision	Recall	F1 Score	Specificity	AUC
Classifier 1	0.928778	0.922305	0.921902	0.922305	0.9221	0.96665	0.987987
Classifier 2	0.904143	0.898961	0.898696	0.898961	0.8981	0.955104	0.9813
Classifier 3	0.908328	0.902246	0.901412	0.902246	0.9011	0.956355	0.982515

Overfitting Analysis: We performed overfitting analysis and found that there was no overfitting. We also experimented with Early Stopping and drop out and found out that data was showing no signs of overfitting.

5.2 Image Classifier Comparison

The goal was to evaluate and compare the multiple classifiers (Perceptron, MLP, CNN, SGD and Random Forest) for emotion detection using image dataset, our dataset was already split to training and validation. The focus was on identifying the best performing architecture based on evaluation metrics like accuracy, precision, recall, F1 score and ROC AUC.

The comparison of machine learning models shows that the Convolutional Neural Network (CNN) achieves the highest accuracy among the evaluated models. Random Forest and Multilayer Perceptron (MLP) also perform well but fall short of CNN's accuracy. The SGD Classifier and Perceptron models have lower accuracy, with Perceptron models performing the worst.

Category	**Details**
CNN (Best Model)	- Accuracy: 37%, Precision: 34%, Recall: 37%, ROC AUC: 70% (best generalization and performance across classes)
Random Forest	- Accuracy: 30% - Balanced precision and recall - Less effective compared to CNN
SGD & Perceptron	- Accuracy: Below 25% - Perceptron struggled due to linear nature - SGD had slightly better recall but limited overall performance

MLP	- Accuracy: 26% - Addition of hidden layers improved performance over Perceptron - Still less effective than CNN
Evaluation of Architectures	- Simple architectures (Perceptron & SGD) struggled due to dataset complexity - Random Forest did well but faced data representation challenges - CNN outperformed all models by understanding shapes, edges, and textures effectively
Overfitting Analysis	- CNN: Early stopping & dropout prevented overfitting, ensuring better validation performance - MLP & Random Forest showed signs of overfitting (training accuracy > validation accuracy)
Class Imbalance	- Classes like “Disgust” underperformed due to fewer images - Augmentation and weighted loss function improved results but did not fully resolve the issue
Future Improvements	- Use pre-trained models (e.g., ResNet, VGG) for transfer learning - Explore SMOTE or class balancing algorithms - Use larger datasets for better accuracy and generalisation
Optimised CNN Model	- **Architecture**: 4 convolutional layers (64, 128, 512 filters), Batch Normalisation, Dropout (25-50%), Fully connected layers (256, 512 neurons), Softmax output - **Preprocessing**: Resized images to 56x56 (grayscale), normalised (0-1), applied augmentation (rotation, flipping, shifting) - **Result**: Achieved 65% validation accuracy with potential for further improvement through additional epochs and optimization

DISCUSSION AND CONCLUSION

The results show our data has both linear and nonlinear patterns, with MLP excelling (93.87% accuracy) at capturing complexity. Logistic Regression (81% accuracy) suggests some linear separability, while Naïve Bayes’ high specificity (94.13%) points to potential class imbalance. Decision Tree performance highlights varying feature importance. Overall, good preprocessing ensured consistent results, but there’s room for feature and model refinement.

On our image dataset, the results highlight the superior performance of the Convolutional Neural Network (65% optimized accuracy), which did well in capturing complex patterns like edges and textures. Random Forest (30%) and MLP (26%) showed moderate performance, while simpler models like SGD and Perceptron struggled (below 25%). Class imbalance affected underrepresented classes like "Disgust," despite augmentation efforts. Future improvements include using pre-trained models, larger datasets, and advanced augmentation techniques to enhance accuracy and generalization.

References

Amiri, M., 2021. *Customer Loyalty Prediction in Python*. GitHub repository. Available at: https://github.com/MohsenAmiri79/Customer-Loyalty-Prediction-in-Python [Accessed 19 Nov. 2024].

Build with Toki, 2023. *Use machine learning to improve customer loyalty*. Available at: https://www.buildwithtoki.com/blog-post/use-machine-learning-to-improve-customer-loyalty [Accessed 22 Nov. 2024].

Fritz.ai, n.d. *Machine learning techniques for predicting customer loyalty*. Available at: https://fritz.ai/machine-learning-techniques-for-predicting-customer-loyalty/ [Accessed 20 Nov. 2024].

Hamdan, I., Othman, M., Mohmad Hassim, Y.M., Marjudi, S., and Mohd Yusof, M., 2023. *Customer loyalty prediction for hotel industry using machine learning approach*. JOIV: International Journal on Informatics Visualization, 7(3), pp.695-703. Available at: https://doi.org/10.30630/joiv.7.3.1335 [Accessed 18 Nov. 2024].

Jadhav, S. and Patil, S., 2020. *Machine Learning Algorithms for Customer Loyalty Prediction*. International Journal of Advanced Technology and Computer Science Engineering, 9(4), pp.45-52. Available at: https://www.warse.org/IJATCSE/static/pdf/file/ijatcse143952020.pdf [Accessed 18 Nov. 2024].

Müller, A., 2024. *Introduction to Machine Learning with Python: Jupyter Notebook*. [online] Available at: https://github.com/amueller/introduction_to_ml_with_python/blob/main/01-introduction.ipynb [Accessed 19 Nov. 2024].

DLSU COMET, 2024. *Introduction to Machine Learning with Python (2017)*. [online] Available at: https://github.com/dlsucomet/MLResources/blob/master/books/%5BML%5D%20Introduction%20to%20Machine%20Learning%20with%20Python%20(2017).pdf [Accessed 20 Nov. 2024].

Shinde, A., 2019. *Feature Selection Techniques in Machine Learning with Python*. Towards Data Science. [online] Available at: https://towardsdatascience.com/feature-selection-techniques-in-machine-learning-with-python-f24e7da3f36e [Accessed 19 Nov. 2024].

Géron, A 2022, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly Media, Incorporated, Sebastopol. Available from: ProQuest Ebook Central. [20 November 2024].