Optimizing Mileage Runs for Frequent Flyers

Yetunde Ademoyo, Robert Bennett, Thomas LaRock, Wei-Yung Liu, Christopher Matro & Sisira Saraswatula

Team 045

1 INTRODUCTION

Frequent Flyer Programs (FFPs) have become an effective marketing tool for airlines by enhancing loyalty customer loyalty and generating revenue [2][1] [12]. Several studies have highlighted FFPs as value-generating enterprises with significant contribution to airline revenue streams and competitiveness [10][4][6]. These programs incentivize customers to book higher-fare, more comfortable options—such as first-class tickets or shorter, direct flights—to optimize their loyalty status [16][10][9]. Airlines use these programs for gaining a competitive edge in retaining customer loyalty [9]. As a result, FFPs affect the price and competition of airlines in the aviation industry [14]. However, current flight search tools, most notably Expedia or Google Flights, are not tailored for the specific needs of frequent flyers seeking to maximize their loyalty status. They focus on cost minimization, allowing users to provide inputs such as intended dates of travel, origin and destination airports, and number of travelers. Yet, they fail to optimize the "mileage run" of frequent flyers who are purposefully purchasing flights to maintain or enhance their status tier. While Expedia provides users with information regarding the number of frequent flyer miles that can be earned for specific itineraries, no existing flight search tool or published studies have focused on optimizing metrics for maximizing a mileage run [1][12]. A similar use-case outlined in existing literature is route selection studies, which applies multi-objective optimization (MOO) with factors such as fare, travel time and comfort for rail systems. MOO is widely applied in transportation networks for route planning, including bus systems, vehicle routing, and even tour routing [15][17][8][5]. However, airlines have a greater number of fare classes and alliances compared to railways, adding complexity to frequent flyers decision-making. Additionally, another study suggests the dominance of proprietary aviation data can hinder reproducibility and limit the ability to extend research findings [11] [3] [7], which is a roadblock to broader consumer insights and to developing more comprehensive tools. As a result, there remains a gap in creating a streamlined tool to optimize flight searches for frequent flyers based on personalized status goals.

2 PROBLEM DEFINITION

Frequent flyers seeking to purchase a mileage run flight must iteratively use flight search tools and manually verify that these trips will qualify for the desired status level. Existing flight search tools prioritize cost minimization or generic travel preferences. They lack the capability to optimize flights for miles earned, segments flown, or fare class multipliers. To address this gap in existing flight search applications, we seek to develop an interactive tool targeted towards frequent flyers specifically seeking to perform a mileage run to attain a desired airline status level. This tool shall contain 2 key components: First is an interface which will allow the user to input current & desired frequent flyer miles, acceptable number of layovers and most notably weights to suggest the degree of important of flight itinerary cost and time. Second is a multi-objective algorithm which is constrained by user inputs to rank optimal available flights which fulfill the requirements for attaining frequent flyer status for the user's airline of choice.

3 LITERATURE SURVEY

A comprehensive literature review reveals a lack of a solution tailored to the flyer seeking to efficiently attain a list of itineraries suitable for performing mileage run. While most existing flight search platforms, such as Expedia, provide basic information on miles earned they do not optimize flight recommendations for maximizing status benefits [1].These platforms are limited to providing transactional data rather than strategic guidance, leaving frequent flyers without tools to make informed decisions about itinerary selection that aligns with their loyalty program goals. In contrast, studies in route optimization, such as for railways and bus systems, utilize MOO to balance fare, travel time, and comfort [18][15][17][8][5]. These frameworks, however, fall short of addressing the complexities of airline decision-making, where fare classes and alliances add additional layers of difficulty which are critical for frequent flyers aiming to maximize status accrual. The traveling salesman problem (TSP), a well-known optimization challenge, presents another relevant analogy. Widely used in logistics and network optimization, TSP's principles of minimizing travel distance while visiting specified locations could inform approaches to optimizing frequent flyer itineraries [13].

Additionally, the dominance of proprietary aviation data presents barriers to extending research findings and developing consumer tools [6]. Literature underscores the need for open, reproducible data to empower advancements in consumer-oriented tools. This roadblock highlights the importance of addressing gaps in current systems to create tools that support frequent flyers' strategic goals, focusing on status optimization rather than cost minimization. Our survey confirms a strong case for developing a flight search optimization tool that bridges these gaps, guided by the methodologies explored in related transportation and optimization studies.

4 METHODOLOGY

4.1 Data Sourcing and Preparation

The data was sourced from the FlightLabs API which provides pricing and logistic data for flights worldwide. The FlightLabs API was created from a variety of worldwide sources using both real time and historical data. While the FlightLabs dataset is over 1TB in size, we extracted only the data necessary to complete a search by leveraging the API search parameters. This includes 1.2 million records across a 5 day period containing attributes such as itinerary departure and arrival locations, total price, fare policy, travel time and connections. Additionally, from OpenFlights, we obtained a list of airport codes to query the FlightLabs API and airport coordinates used for spatial visualization in the user interface.

While each airline has unique rules for earning, redeeming and maintaining loyalty points or miles, building a system that encompasses all airline FFPs and their intricacies was out of scope of our 10 week project timeline. Therefore, this project focuses on Delta Airlines flights. However, the versatility of the chosen API would allow our framework to be scaled to other airlines. Delta's SkyMiles program offers numerous ways to earn miles, including flights booked through partner airlines, hotel bookings, and credit card spending. In this project, we focus exclusively on Mileage Qualifying Dollars (MQDs) directly related to airline purchases: the primary metric for achieving Medallion status. While credit card spending, hotel partnerships, and other methods also contribute to overall SkyMiles earnings, they fall outside of our project scope.

4.2 Exploratory Data Analysis

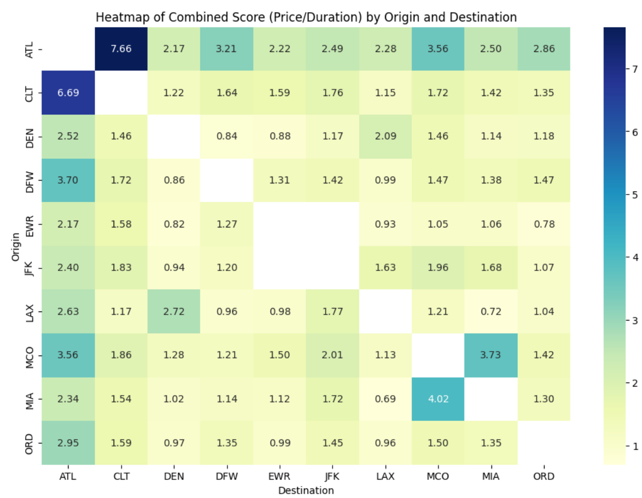

Figure 1: Flight Price vs. Duration Pain Matrix

Using a 5 day data subset contain 1.2 million records, performing exploratory data analysis highlighted the 10 most common routes for Delta Airlines to include the following origin or destination airports: Atlanta (ATL), Charlotte (CLT), Denver (DEN), Dallas Fort Worth (DFW), Newark (EWR), New York John F. Kennedy (JFK), Los Angeles (LAX), Orlando (MCO), Miami (MIA), and Chicago O'Hare (ORD). For Delta Airlines, flyers receive 1 MQD for each dollar spent on air travel for all seat classses except Basic Economy. As part of data exploration, we sought to understand routes which maximize MQD with minimal flight duration. Figure 1 above is a heatmap highlighting the price/duration ratios for the pairings of the 10 most popular routes. A higher intensity value indicates a higher price to duration tradeoff, suggesting it is an optimal route for maximizing MQD in less time. Per Figure 1, optimal routes would be between

(1) Atlanta and Charlotte (2) Orlando and Miami & (3) Atlanta and Orlando. Less optimal routes would be between (1) Los Angeles and Miami (2) Denver and Dallas Fort Worth & (3) Denver and Newark.

4.3 Multi Objective Optimization

For the purposes of this project, the algorithm to rank the optimal mileage run can be expressed as a multi objective optimization which, as previously highlighted, is commonly used to address route planning problems as they share the similar goal of identifying an optimal route from a finite set of objects [15]. Here, we take a scalarization approach using at least two conflicting user defined criteria which are the objectives and constraints:

$$\min Z(x) = w_1 S(x) + w_2 C(x) + w_3 T(x)$$

$$\text{s.t. } C(x) \geq target$$

The first step in the algorithm, Input Preparation (Algorithm 1), involves filtering the underlying flight data using user defined constraints: origin, dates, and maximum layovers. This ensures that the duration, price and connections will be extracted only for flights which meet the specified criteria.

Once the input data is prepared, the next step is to generate qualifying routes (Algorithm 2). This involves initializing routes with flights departing from the identified origin, and then iteratively identifying and extending these routes with connecting flights that satisfy the layover constraints within the provided dates. Routes are only considered valid if they meet the target criteria, such as completing the journey or achieving the required total miles.

After generating a list of valid routes, the next step is to normalize the criteria per route (Algorithm 3). This involves transforming the raw values of each route's attributes (such as duration, price, and number of layovers) to a scale that is comparable across all routes. The normalization process ensures that each criterion has the same weight when combined into the final score.

Next, we calculate the weights for each criterion (Algorithm 4). The weights can either be dynamically computed using an entropy based method, which quantifies the uncertainty or variability of the criterion, or they can be user defined based on adjusted Cost Weight and Time Weight. In the entropy based method, criteria with higher uncertainty are given lower weights, while those with more consistency are assigned higher weights.

With the normalized criteria and corresponding weights, we can now calculate the weighted score for each route. The score reflects the relative desirability of each route based on the user's preferences. To account for diversity in the routes, an adjustment is made to reduce the score for routes that share common layovers. Finally, the routes are ranked in ascending order based on their adjusted scores.

Algorithm 1 Input Preparation

Require: Flight data F
Require: User Preferences P
Require: Constraints C
Ensure: Filtered flight data FFD
Filter flights based on constraints C (e.g., origin, dates, & maximum layovers).
Extract relevant attributes: duration, price, and connections.

Algorithm 2 Route Generation

Require: Filtered flight data FFD
Ensure: List of valid routes R
Initialize routes with flights departing from the origin.
for each partial route r in current routes **do**
    Identify valid connecting flights based on layover constraints.
    Extend route r by appending valid connections.
    if route r meets the target (e.g., total miles, returns to origin) **then**
        Add to the list of valid routes R.
    end if
end for

Algorithm 3 Normalization of Criteria

Require: List of valid routes R
Ensure: Normalized route attributes nR
for each criterion c **do**
    Compute normalized value:

$$\text{Normalized Value} = \frac{\text{Actual Value} - \text{Minimum Value}}{\text{Maximum Value} - \text{Minimum Value}}$$

end for

Algorithm 4 Weight Calculation

Require: Normalized route attributes nR
Require: User Preferences P
Ensure: Weighted criteria wC

if using dynamic weighting (entropy based) **then**
    Compute entropy for each criterion:

$$H_c = -\frac{1}{\log(N)} \sum_{i=1}^{N} p_i \log(p_i)$$

where p_i is the distribution of normalized values for criterion c.
    Assign weights inversely proportional to entropy:

$$w_c = \frac{1 - H_c}{\sum_c (1 - H_c)}$$

else
    Use user defined weights w_c from preferences P.
end if

Algorithm 5 Score Calculation and Ranking

Require: Weighted criteria wC
Ensure: Ranked list of Routes R

for each route r_i **do**
    Compute weighted score:

$$\text{Score}_i = \sum_c \text{Normalized Value}_{i,c} \cdot w_c$$

    Adjust for route diversity:

$$\text{Final Score}_i = \text{Score}_i \cdot \frac{1}{1 + \text{Common Stops}}$$

end for
Sort routes R by Final Score in ascending order.

4.4 Mileage Run Visualization

For the mileage run visualization, we built an interactive tool displaying qualifying flights for frequent flyers seeking flight itineraries to perform a mileage run (Figure 2). The main Python libraries used to build the mileage run tool are streamlit as the main web framework and pydeck for the map visualization. There are several notable components to the visualization tool which are either consistent with existing flight search applications, such as Expedia and Google Flights, or are enhancements to these applications to account for the needs of frequent flyers seeking flight itineraries to perform a mileage run.

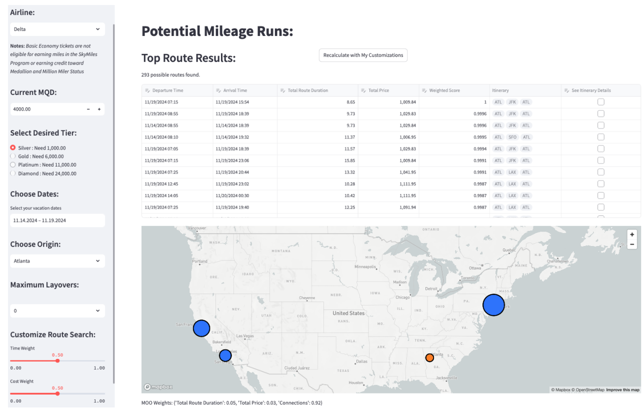

Figure 2: Mileage Run Tool User Interface

In alignment with existing flight search applications, users can optionally enter their travel requirements such as origin and dates of travel. Using these inputs, the mileage run tool will begin to filter flights using relevant criteria However, the enhancements upon existing flight applications are features which are targeted towards frequent flyers seeking flight itineraries to perform a mileage run. First, are the "Current MQD" and "Select Desired Tier" fields which require the user to input current MQD and target tier to finetune route recommendations. The difference between the MQD required for a particular status level and current MQD is the primary constraint for the underlying multi objective optimization algorithm which will output an initial set of optimal routes. Second, is the "Customize Route Search" feature which allows users to input "Cost Weight" and "Time Weight". This enables frequent flyers to define the relative importance of time and cost which is used to rerank the route recommendations. This is a significant enhancement upon existing flight search applications which fail to consider the degree of importance of the various variables they make available for filtering.

4.5 Innovation

The mileage run tool offers a handful of key innovations, including most notably: the utilization of a multi-objective optimization algorithm, which uniquely addresses the challenge of optimizing mileage runs by balancing multiple competing objectives, unlike traditional flight search platforms which often focus on a singular objective of finding the cheapest flight. Enhancing the interface to provide users with the ability to identify current MQD, desired tier, and the degree of importance of travel cost & time minimization, which directly impacts the weight of the values in the underlying multi-objective optimization algorithm to help personalize the outputs to the frequent flyer.

5 EXPERIMENTS & EVALUATION

5.1 Estimate of API Reliability

A key challenge is that the FlightLabs API frequently returns incomplete queries without a detailed reason. To acount for this, we included a backoff strategy that waits between iterations and periodically randomizes the remaining search points. We can iterate over the elements of $\mathcal{S}$, which is a set of distinct grid search points, each having a fixed number of features, with a few considerations. When a query returns complete data, the current point is removed from the queue as normal. When a query returns incomplete data, the current remains in the queue, which is shuffled to allow other dimensions of the search to be prioritized. The data obtained is preserved along the way, and duplicates are removed after search completion. If $\tilde{k}$ denotes the expected number of API calls required to have at least one success before moving to the next search point, then $K_{\max} = |\mathcal{S}| \times \tilde{k}$. An estimation of $\tilde{k}$ was completed using a Bayesian Beta-Binomial framework, noting that $\tilde{k} \geq \frac{\ln(1-\alpha)}{\ln(1-p_{\text{success}})}$, where α is the desired degree of confidence. The empirical estimate of p_{success} was calculated based on repeated batches of API calls, which had a sample mean of 0.22 and standard deviation of 5. In order to be 95% certain of at least one successful API call for a single point in $\mathcal{S}$ before moving onto the next point, we need to use on average 12 API calls. This becomes cost-prohibitive for an arbitrarily large grid search, without a competitive pricing agreement. Therefore, this approach, although reliable, may leave behind a more optimal solution due to computational constraints.

5.2 Numerical Experimentation

Numerical experimentation were performed to evaluate the multi objective flight route optimization algorithm's ability to rank routes based on varying constraints for time, cost, and layovers. For this experiment, the assumed current MQD value is 4,000 with the desired tier set as Silver, requiring 1,000 additional MQD with the Origin set to Atlanta. 4 scenarios were evaluated in the order shown: (1) cost priority (2) time priority (3) cost/time balanced (4) optimized:

Parameters	Top 3 Best Routes
Time Weight: 0.0 **Cost Weight:** 1.0 **Stopover Weight**: 0.0	(1) Destination: SFO | Cost: $1,006.95 | Time: 11.37 | Score: 1.0000 (2) Destination: SFO | Cost: $1,006.95 | Time: 131.37 | Score: 1.0000 (3) Destination: SFO | Cost: $1,006.95 | Time: 133.32 | Score: 1.0000
Time Weight: 1.0 **Cost Weight:** 0.0 **Stopover Weight**: 0.0	(1) Destination: JFK | Cost: $1,009.84 | Time: 8.65 | Score: 1.0000 (2) Destination: JFK | Cost: $1,029.83 | Time: 9.73 | Score: 0.9973 (3) Destination: JFK | Cost: $1,029.84 | Time: 9.73 | Score: 0.9973
Time Weight: 0.5 **Cost Weight:** 0.5 **Stopover Weight**: 0.0	(1) Destination: JFK | Cost: $1,009.84 | Time: 8.65 | Score: 0.9995 (2) Destination: SFO | Cost: $1,006.95 | Time: 11.37 | Score: 0.9950 (3) Destination: JFK | Cost: $1,029.83 | Time: 9.73 | Score: 0.9948
Time Weight: 0.05 **Cost Weight:** 0.03 **Stopover Weight**: 0.92	(1) Destination: JFK | Cost: $1,009.84 | Time: 8.65 | Score: 1.0000 (2) Destination: JFK | Cost: $1,029.83 | Time: 9.73 | Score: 0.9996 (3) Destination: JFK | Cost: $1,029.84 | Time: 9.73 | Score: 0.9996

Figure 3: Route Optimization Numerical Analysis

In the cost priority scenario, the algorithm ignored travel time entirely, focusing solely on cost minimization. Likewise, the time priority scenario provided routes with the shortest duration with slightly higher expense. This distinction highlights the importance of weight configurations when optimizing for multiple objectives. While the cost/time balanced scenario shared similar routes to the optimized scenario, the returned routes were scored lower highlighting the importance of the stopover/layover weight. Finally, the optimized scenario gave a significantly higher weight to stopovers (0.92) as opposed to time & cost suggesting the greater importance given to the number of stopovers to produce a diverse set of highly ranked routes.

5.3 User Feedback

Survey results were gathered from 8 participants who were tasked with the following: (1) Performing a mileage run search using the interactive visualization tool and Google Flights and reporting back on the task completion time using each tool (2) Ranking 5 of the key features available on the mileage tool (3) Providing, on a scale of 1 through 10, satisfaction with the mileage run tool. The results of the task completion time can be found in Figure 4 which indicates that 7 out of 8 participants found that the mileage run tool took 5 minutes or less to find a optimal mileage run whereas it took 7 out of 8 participants more than 5 minutes to

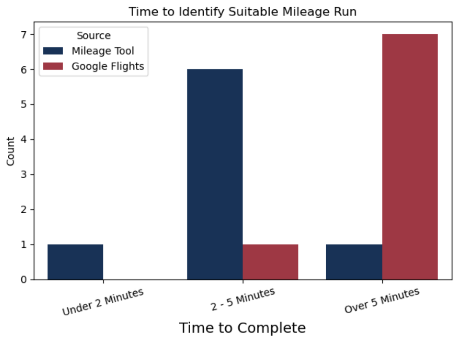

Figure 4: Time to Identify Suitable Mileage Run

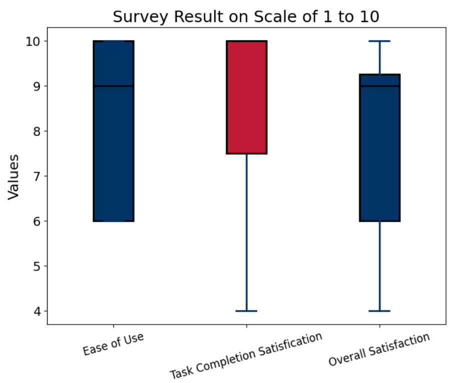

Figure 5: User Satisfaction on Scale of 1 to 10

find an optimal mileage run. The results of the key features ranking found in Table 1 suggests that, in order of importance, participants found the key features to be: mileage run outputs, route recommendations, search speed, filtering options & interface design. This feedback suggests that participants responded positively to the novelty of performing a mileage run optimization quickly and efficiently and outputting recommended routes accordingly. When asked to rate their satisfaction on a scale of 1 through 10, on average, users indicated high satisfaction, 9 out of 10, with ease of use and overall satisfaction as documented in Figure 5. Finally, to identify areas of future improvement, we requested participants to provide general subjective feedback. The most critical feedback highlighted adding more airline options, integrating direct booking links to facilitate the purchase of identified mileage runs, and improving the GUI. This suggests that the tool's core functionality is accessible, but the user interface could be improved in future studies.

Feature	Average User Ranking
Interface Design	3.625
Filtering Options	3.125
Search Speed	2.875
Route Recommendations	2.714
Mileage Run Outputs	2.625

Table 1: Ranking of Interactive Tool Features

6 CONCLUSION

This project highlights strides to apply route optimization algorithm to help frequent flyers identify flight itineraries for maximizing a mileage run. While we limited the scope of this study to a single airlines & few key airports, we have proven that a route optimization approach can be applied to the airlines industry. Numerical analysis highlighted the algorithm's flexibility in accommodating diverse user inputted preferences, suggesting that future work could explore optimizing on additional constraints. Additionally, user feedback highlighted that key areas of future development would be to (1) improve the user interface to provide users with more search, filter and sort capabilities (2) extend beyond Delta Airlines and integrate with airlines to facilitate direct booking. However, due to inconsistency of sourcing a complete query using FlightLabs API and associated cost constraints, a key milestone would be to identify a methodology for data sourcing which is scalable across all airlines, airports and dates. Additionally, while a cached dataset was used for the scope of this project, connecting to a real-time API would allow for up-to-date flight prices which are ever-changing due to market demand.

6.1 Distribution of Effort

Bennett, Liu, & Saraswatula are the primary authors of this report with contribution from the entire team. Saraswatula lead the poster, supported by Bennet & Liu. Bennett lead the front-end and back-end of the mileage run tool, supported by Liu. Liu lead the algorithm development, supported by Adeyemo & Matro. LaRock lead the data collection and exploratory data analysis. Saraswatula & Adeyemo worked on the experiments and evaluation. Adeyemo worked on the demo video and README.txt. Liu & Matro worked on the literature survey.

REFERENCES

[1] E. R. De Boer. 2017. *Future of the frequent flyer program.* 227–244 pages. https://doi.org/10.1007/978-3-319-62600-0_9

[2] E. R. De Boer and S. V. Gudmundsson. 2012. 30 years of frequent flyer programs. *Journal of Air Transport Management* 24 (2012), 18–24. https://doi.org/10.1016/j.jairtraman.2012.05.003

[3] G. Dudás, L. Boros, V. Pál, and P. Pernyész. 2015. Mapping cost distance using air traffic data. *Journal of Maps* 12, 4 (2015), 695–700. https://doi.org/10.1080/17445647.2015.1061463

[4] Y. Gao. 2020. A conceptual framework for valuating airline frequent flyer program miles. *International Journal of Aviation Aeronautics and Aerospace* (2020). https://doi.org/10.15394/ijaaa.2020.1447

[5] A. W. Holt, A. J. Reichenbach, and R. Millar. 2004. Impact of frequent flyer program benefits on consumer preferences. *Journal of Air Transport Management* 10, 2 (2004), 129–138. https://doi.org/10.1016/j.jairtraman.2003.09.002

[6] C. Lee-Anant. 2022. The Importance of Frequent Flyer Programs: An in-depth analysis. *Journal of Aerospace Technology and Management* 14 (2022). https://doi.org/10.1590/jatm.v14.1254

[7] M. Z. Li and M. S. Ryerson. 2018. Reviewing the DATAS of aviation research data: Diversity, availability, tractability, applicability, and sources. *Journal of Air Transport Management* 75 (2018), 111–130. https://doi.org/10.1016/j.jairtraman.2018.12.004

[8] G. Mason and N. Barker. 1996. Buy now fly later: an investigation of airline frequent flyer programmes. *Tourism Management* 17, 3 (1996), 219–223. https://doi.org/10.1016/0261-5177(96)00016-6

[9] S. Mumbower, L. A. Garrow, and J. P. Newman. 2015. Investigating airline customers' premium coach seat purchases and implications for optimal pricing strategies. *Transportation Research Part A: Policy and Practice* 73 (2015), 53–69. https://doi.org/10.1016/j.tra.2014.12.008

[10] S. M. Nako. 1992. Frequent flyer programs and business travelers: An empirical investigation. *Logistics and Transportation Review* 28, 4 (1992), 395. https://www.proquest.com/scholarly-journals/frequent-flyer-programs-business-travellers/docview/197442195/se-2

[11] C. N. Reales and J. F. O'Connell. 2017. An examination of the revenue generating capability of co-branded cards associated with Frequent Flyer Programmes. *Journal of Air Transport Management* 65 (2017), 63–75. https://doi.org/10.1016/j.jairtraman.2017.08.001

[12] H. Sahin, A. O. Kusakci, and B. Mbowe. 2021. The effects of frequent flyer programs in the airline industry on customer loyalty. *Heritage and Sustainable Development* 3, 2 (2021), 130–147. https://doi.org/10.37868/hsd.v3i2.69

[13] M. Seelhorst and Y. Liu. 2015. Latent air travel preferences: understanding the role of frequent flyer programs on itinerary choice. *Transportation Research Part A: Policy and Practice* 80 (2015), 49–61.

[14] N. S. Terblanche. 2014. Customers' Perceived Benefits of a Frequent-Flyer Program. *Journal of Travel & Tourism Marketing* 32, 3 (2014), 199–210. https://doi.org/10.1080/10548408.2014.895694

[15] X. Yang, L. Liu, Y. Li, and R. He. 2013. Route selection for railway passengers: a multi-objective model and optimization algorithm. *Journal of Transportation Systems Engineering and Information Technology* 13, 5 (2013), 72–100. https://doi.org/10.1016/s1570-6672(13)60124-5

[16] M. Zakir Hossain, H. Kibria, and S. Farhana. 2017. Do customer loyalty programs really work in airlines business? A study on Air Berlin. *Journal of Service Science and Management* 10, 4 (2017), 360–375. https://doi.org/10.4236/jssm.2017.104029

[17] J. Zhang and Y. Wang. 2022. Aircraft Traffic Visualization System for Airline Companies. In *7th International Conference on Information Science, Computer Technology and Transportation.* 1–5. https://doi.org/10.1109/ISCTT54194.2022.10071870

[18] Jinhang Zhang and Yi Wang. 2022. Aircraft Traffic Visualization System for Airline Companies. In *ISCTT 2022; 7th International Conference on Information Science, Computer Technology and Transportation.* 1–5.