Learning ML

TensorFlow / Google Colab Intro

Machine Learning (ML) Models for Prospective Aircraft and City Servicing Decisions

---

Pat Gardner

Abstract: How can airlines and aircraft manufacturers leverage machine learning technologies in their respective businesses? With technology becoming ever more available through the advent of open-source projects and the cloud, not to mention digital storage becoming ever cheaper according to Moore's Law, one can reasonably expect wider application of previously complex technological methods in various industries. While complex topics such as big data, Internet of things (IoT), and machine learning (ML) remain in the wheelhouse of tech companies since they

develop the technologies themselves, it seems that the same ideas can stick out as overused buzzwords in other businesses. This report intends to confront one of these, ML, and determine whether players in the Large Commercial Aircraft (LCA) industry can benefit from it and to what extent those on both sides – manufacturers as sellers and airlines as purchasers – will employ it. The benefit from reading the research most directly applies to strategists within the aircraft and airline industries, but extends to any businessperson with an interest in knowing how ML will impact their future. See the Appendix to access and run the models under research.

BIO-STATEMENT

Pat Gardner, a third-year student, studies Computer Engineering with a certificate in Supply Chain and minors in Mathematics and Economics at the University of Pittsburgh. His internship experience ranges from a business analyst role at K&L Gates to software engineering roles at Capital One and Google. He has also developed software for a physical therapist's PhD project, which has won $175,000 in grant awards. On campus, he serves as President of both the Engineering Business Administration club and the Panther Algorithmic Trading club. Outside of the classroom, he has served as Resident Assistant for Pitt's underclassmen and upperclassmen engineering halls. Upcoming, he intends to intern as a Junior Trader at Belvedere Trading this summer and serve as a Community Assistant for Pitt's student athletes in the fall. He plans to graduate in either Winter 2020 or Summer 2021 with the intention of working full-time at a hedge fund, trading firm, or major technology company. His future aspirations include providing the greatest benefit to society and his employer, most likely within management, as well as potentially starting his own hedge fund. Email: pag73@pitt.edu.

INTRODUCTION

Both airlines and aircraft manufacturers operate in a complex competitive environment. To best determine machine learning's (ML) applicability to these businesses, this research takes the experimental approach of creating three exemplary, yet primitive, models with available data. Using Google's open-source software library TensorFlow in tandem with its free cloud development environment Google Colaboratory, the creation of these models is viewable to and modifiable by any reader. While many mathematical models in research remain confusing to the common person, the background and interactivity of this research should give more readers a decent grasp on how ML works and how one can apply it. By creating and training feasible models on public information, the report showcases the possibilities for professional companies and organizations which have greater resources and more detailed internal metrics.

Although most applicable to those involved in strategic decision-making in airline and aircraft manufacturing businesses, this research will aid any student or businessperson in better understanding ML and its potential future impacts. Throughout the following sections filled with outside evidence and model outputs, this report will answer the following question: how can airlines and aircraft manufacturers best leverage machine learning technologies in their respective businesses?

MACHINE LEARNING SIMPLE OVERVIEW

The wide range of pre-written ML tools abstract much of the process's complexity from the researcher; however, it does behoove one to understand the fundamentals in order to avoid incorrect assumptions. While a computer model attempts to learn a relationship between input(s) and output(s), it does not do so magically. In a series of iterations over the data, the computer attempts to make error-minimizing modifications – however slight or big according to the set learning rate – to its "function" that maps between input and output. One can imagine the difficulty of running this process without the powerful computers and pre-made tools of today's technology age.

Basic Mathematical Background

Without going too in depth on the math, one can understand the basic underpinnings of ML reliant upon algebra. Consider a simple linear regression: mathematically, one minimizes the sum of the squares of the residuals of all examples under study. Taking this to ML, examples are composed of input "features" and output "labels" or "targets". The sum of squares, which the model should minimize over time, becomes the "loss function" of the model. The least squares loss function is referred to as mean square error (MSE) and is commonly utilized in machine learning (Google, n.d.).

The reduction in loss of the model remains the complex part, and thus, one must venture into the realm of calculus. For the simple linear regression example, the coefficient for each variable represents the associated "weight" for each input feature in ML. Each weight has a corresponding loss value. When looking at a plot of weight versus loss for a convex problem, one notices that the plot will always be 'U'-shaped (Google, n.d.). The loss function converges at the minimum point in this plot, with a slope of zero; modifying the weight for an input at this point would only increase loss. To get to this point, one must realize that the derivative, or ML-termed "gradient", of the loss function will always point in the direction "of steepest increase in the loss function" (Google, n.d.). With the goal of minimizing loss, the model is programmed to move in the direction of the negative gradient, or steepest decrease in the loss function. Thus, the model reduces loss and finds the optimal weights to map the input features to the labels or targets.

Mapping Input Features to Targets

Within machine learning, this paper focuses on the topic of supervised learning, where one clearly informs the model of each example's input features and output labels. Worth mentioning is the idea of unsupervised learning, wherein the researcher does not input labelled examples. Instead, the model attempts to learn relationships among input features and create its own associations and groupings or clusters. The two types of learning can have very different use cases due to the requirement of "ground truth" labels in supervised learning.

Training and Validating the Model

After deciding upon the features and labels, a researcher must then train the model. In training, the investigator must set some parameters manually. These parameters, called hyperparameters, include the learning rate or step size, the batch size, the number of steps, and the training/validation/test data split. The learning rate dictates how big of an adjustment the model makes per data point. Through multiple training runs, one usually finds the right balance between learning too slowly and learning so quickly that the model overshoots the minimum repeatedly. The model will train over N batches of X examples (where N is number of steps, and X is batch size); with a high number of steps, the model might have to repeatedly go over the same data.

In order to confront the inevitable consequence of overfitting (i.e. learning only the very specific relationship of current data, and therefore not predicting new outputs well) one must set the data split for training, validation, and test data. The model learns on the training data each training run. When tweaking hyperparameters, one sees the error results from validation data, which is tested for loss but does not teach the model. After one completes all hyperparameter adjusting and model training, the model runs over the test set in order to see its effectiveness on completely removed examples (Google, n.d. “Machine Learning Crash Course”).

In training the model, various optimizers exist. Gradient descent, one of the simplest, just follows the idea of moving in the direction of the negative gradient of loss. The simple becomes ineffective in more complex scenarios where a problem does not fit the regular convex example (i.e. it has multiple minima). More recently developed optimizers make use of techniques such as modifying the learning rate over time and applying moving averages to gradients to avoid getting stuck in a sub-optimal minimum (Bengio 2012, 9). This research uses one such optimizer, Adam, as it has relatively great results in achieving a minimal loss quickly (Kingma and Ba 2014, 1-7).

Utilized Technologies

Past the theoretical portion of this transformational technology, this research shifts to the practical. A variety of ML software libraries exist for open-source use. The following models use TensorFlow, a very common one originating from Google. The TensorFlow code written for this research utilizes Python as its language of choice, also available to the general public. Finally, these models train and run on Google’s free cloud service Colaboratory, or “Colab” for short. Essentially, Google provides a limited and varying amount of free computational power for anyone in the form of remote graphics processing units (GPUs) (Google, n.d. “Colaboratory”). Cloud technology, or temporary use of a company’s compute, allows for operation on a code execution cost basis. Cloud services can take the place of a technologist’s burdensome pastime of having to buy and set up servers and GPUs. With such availability and ease of use, one should not find ML’s rapid adoption surprising.

AIRLINE AND AIRCRAFT MANUFACTURER BACKGROUND

Besides the massive amounts of capital necessary for research and development and operation in both the aircraft and airline businesses, the industries are also distinct in that the miniscule difference between "success and catastrophic failure [lies] in the subtlety and perfection of program management and systems integration" (Fallows 2013, 163). With potentially disastrous costs and difficult-to-achieve profits, these sectors can serve as an interesting environment for those empowered by ML to improve recognized relationships and even find unrecognized ones.

Industry-shocking events such as 9/11 or the recent Coronavirus (COVID-19) pandemic evidence the quick shifting in strategy required to compete and remain in these types of businesses (PricewaterhouseCoopers n.d.). Handcrafted models can falter at the occurrence of an unexpected world event, requiring researchers and analysts to head back to the metaphorical drawing board. On the other hand, ML models can actively learn from incoming data and make adjustments continuously, providing faster and more effective decision-making.

A Look at Airline Profitability

Airlines have quite a complex problem in their hands: maximize profit by trying to best predict ever-changing passenger demand and reduce operational and fixed costs. Connected with the aircraft industry, airlines must buy or lease airplanes for use. A private aircraft agent, which will remain anonymous, mentioned that an airline's choice of aircraft for profitability may best correlate with operating routes, seat revenue, and costs.

The complexity is magnified by the network of routes which is dependent upon each route's demand, the airline's pricing strategy, and "the capacity of the fleet it [operates]" (Hebert and Taleb 2011, 6). Further, carrier profitability depends on both the acquisition and exploitation costs of its airplanes, and performance “[is] essentially contingent on the effective control of [its] operating costs” (Hebert and Taleb 2011, 6).

Demonstrating the interdependence of considerations, three researchers found it beneficial to integrate the interplay of the crew-scheduling procedure “with the airline operations of schedule design, fleet assignment, and aircraft routing” in a consolidated model, yet determined it to require further work (Sherali, Bae, and Haouari 2013, 474). Overall, one can see how many potential relationships must be taken into account for profitable airline models. Since most relevant data for such models remains proprietary and thus internal to the airlines, this research intends to demonstrate ML’s potential power for mapping features to outputs.

The Large Commercial Aircraft (LCA) Sector

Before the Coronavirus tanked demand in the airline and, therefore, aircraft industries, geographically widespread opportunities for growth abounded. In 2005, 35 percent of global demand existed in the BRIC (Brazil, Russia, India, China) countries – up from less than 5 percent a few years earlier (Hebert and Taleb 2011, 7). The demand in developing countries causes companies such as Airbus and Boeing to consider the advantages and disadvantages of expanding operations. With China as a specific example, they must balance “the opportunities of the world’s fastest-growing market with the challenges posed by the world’s most rapidly expanding industrial base” (Fallows 2013, 145).

Keeping in mind how quickly their demand changes, aircraft manufacturers can benefit from applying machine learning technology to detect moving trends in the industry. Growing and innovating from a yet established business, competitors from developing areas such as China can potentially more quickly implement ML and reap its available competitive advantages.

MODEL 1: AIRCRAFT LIST PRICE

The first model attempts to determine an aircraft's list price based on a variety of input features. All of the data used in this model comes from 2017. First, the list prices originate from Airbus' official website and a financial report on Boeing's list prices in 2017 (Airbus 2017, Ausick 2017). With the goal of learning about profitability of different kinds of aircrafts, it makes sense to begin with determining what goes into their pricing.

Approach

The first model uses a linear regressor, one of the simplest estimators, in tandem with the aforementioned Adam optimizer. A linear regressor takes any number of numerical input features and attempts to predict a continuous numerical output target.

Much of the logic of this model, as well as of the subsequent ones, originates from one of Google's examples on model validation (Google n.d. "Validation"). The Colab trains the model over subsequent periods, keeping track of the RMSE of both the training set and validation set as a tracker of model error. Plotting the graph of both RMSE values over the ten periods shows the model learning the relationship between inputs and output.

In determining the input features, available characteristics on all of the planes with prices were found in a December 2017 booklet prepared by DVB Bank SE's Aviation Research department (DVB Bank SE 2017). These characteristics include standard 2-class seating quantity, maximum range in nautical miles, and year of first flight. In order to make the year of first flight a learnable input of more sensible range, this feature is transformed into years between the airplane's first flight and 2017. While ideally one should normalize all inputs within the same range of each other – this can help the model converge quicker – the model leaves out this extra step for simplicity's sake. Finally, each input feature can alone train the model, allowing for comparison in each feature’s effectiveness to predict price, and any combination of features can together train the model for potentially better results.

Due to the limited number of examples available, a test set was left out of this and subsequent models. Test sets mainly serve to ensure that hyperparameters do not specifically apply to the current data, but little hyperparameter tweaking occurs in these simple experiments. Adjustments in learning rate occur to attempt to create a leveling off in learning. To ensure that the models do not over-fit to the data, the programs utilize a simple split of approximately 80 percent training data and 20 percent validation data.

Results

Training with all three of the input features results in a very nice reduction in training and validation error over time, as seen in Figure 1 below.

```
Training model...
RMSE (on training data):
  period 00 : 173.95
  period 01 : 148.34
  period 02 : 123.07
  period 03 : 98.36
  period 04 : 74.86
  period 05 : 54.04
  period 06 : 40.35
  period 07 : 39.47
  period 08 : 39.77
  period 09 : 39.63
Model training finished.
```

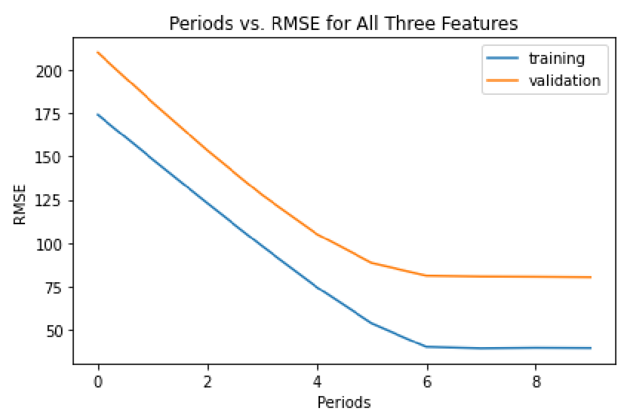

Figure 1: Training on All Three Features for Model 1

As expected, training error results turn out much lower than validation, as the model directly learns from the training data split. The graph evidences, though, that the model reduces validation error without ever seeing the validation data. Clearly, there exists a relationship between the three input features and the output of list price. After training, the model can output predictions within the Google Colab. However, since few input features of few examples make up the training data, these predictions do not result in very meaningful information.

Instead of trying to utilize predictions, another methodology can provide useful information. The model can selectively utilize any combination of features or even a single feature. Seeing how model error varies with each feature alone can demonstrate which factors have greater influence on price. Running the model with each of the single features produces the plots in Figure 2 below.

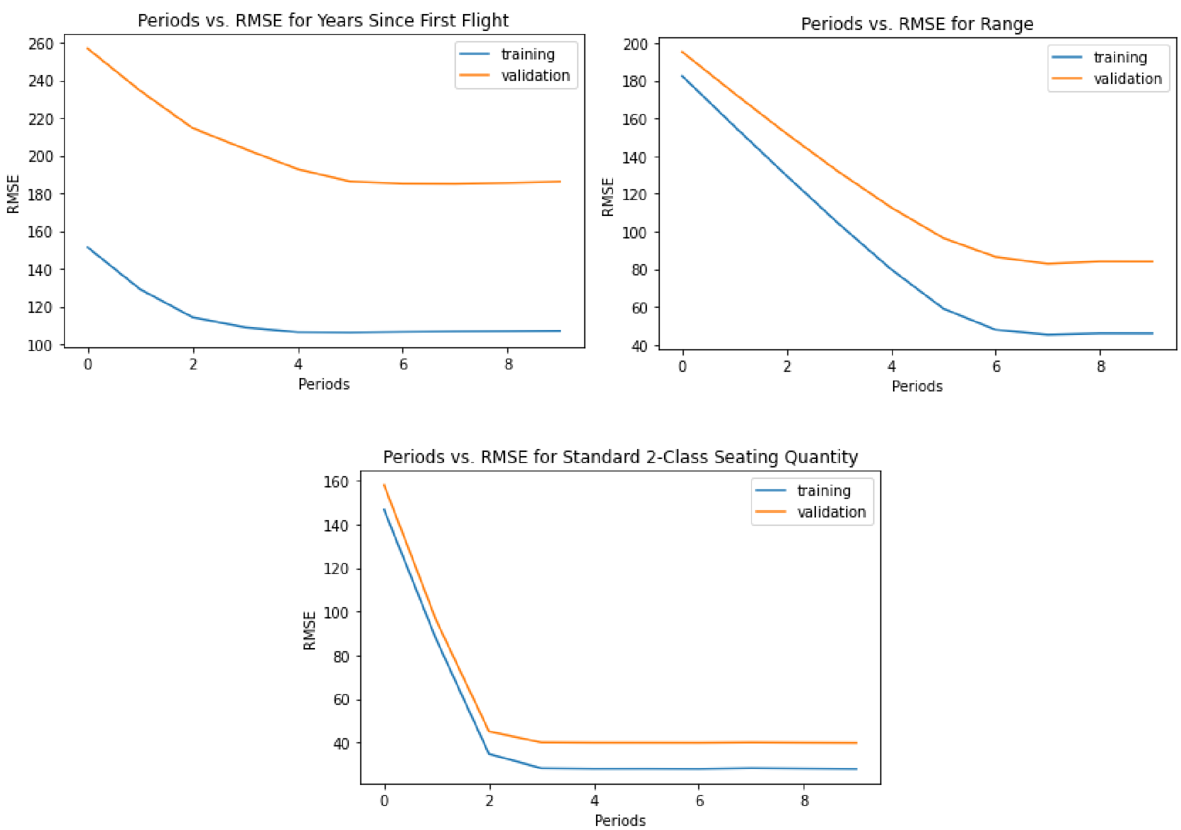

Figure 2: Training on Each of the Three Features for Model 1

Based on this experiment, standard 2-class seating quantity best predicts price, followed by range and then years since the model's first flight. Seating best reduces error for the model in both the training and validation data splits.

Takeaways

This model demonstrates that a learnable relationship exists between the three input features and the list price of an aircraft. Applying each feature alone to the training of the model illustrates how the best to worst predictors of price are standard 2-class seating quantity, maximum range, and years since the model's first flight. Due to lack of normalization and potential lack of usefulness of a feature, the combination of all three inputs does not result in the least amount of error. Clearly, ML can rank the importance of factors on predicting a continuous output. Such a utility can extend to many other use cases.

MODEL 2.1: ENTERING AIRLINE SERVICE OF CITY

The second model attempts to determine whether a major airline of a country should enter into servicing a city based on the city's passenger throughput and whether two other major airlines service said city. The United States serves as the country in running this model, while United Airlines, American Airlines, and Delta Airlines serve as the major airlines, and American airports with a throughput of at least two hundred thousand passengers serve as the cities. On the next page, Figure 3 geographically shows the airports under study, with locational data coming from a cross-reference between two sources (Partow n.d., Toolforge n.d.). Enplaned passengers numbers originate from a Bureau of Transportation Statistics report on scheduled enplanements on U.S. and foreign airlines in each airport (Bureau of Transportation Statistics 2019). Data on airlines servicing comes from using the Wayback Machine, looking at each airline's official website. In order to access sites from the past, the internet archive Wayback Machine consults its saved webpages over time. Such a system allows for data collection from 2017.

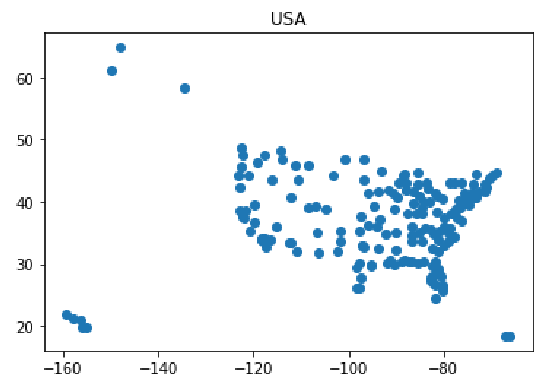

Figure 3: American Airports Under Study

Approach

The second model utilizes an Adam optimizer in a linear classifier with output of either zero or one, one meaning it predicts the third airline should service that city. Training and validation examples originate from holding out one airline at a time as the ground truth output. Meanwhile, enplaned passengers in 2017 and how many of the other two serviced the city in 2017 serve as input features. Geographical data can be useful in a model like this, but since this model has a linear basis, it cannot learn regional differences which have a two-dimensional nature.

Rather than RMSE, this experiment utilizes a different error feature: mean absolute error (MAE). With the coding of this Colab, the MAE value shows the average difference between the true output and the probability with which the model expects the output to be one (airline should service).

Results

Experiment 2.1 produces far less valuable results than the first one, as the model cannot find a great relationship between inputs and outputs, as evidenced by the following line plot in Figure 4.

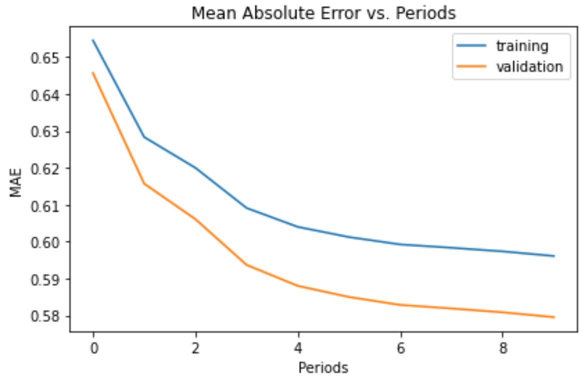

Figure 4: Training on Model 2.1

While the training reduces error over time, an average error of 0.60 at the end of training for a process which outputs somewhere between 0.0 and 1.0 remains lackluster. Oddly enough, validation error improves better than training error, further signifying an unideal relationship. Due to lackluster results, further investigation into a similar model's viability in China's ecosystem is not undertaken.

When using the model to predict outputs for a given city with an input number of passengers and an input number of major airlines servicing, one can see the relationships the model has found. Upon increasing passengers, the prediction goes up substantially. However, in changing the number of airlines servicing, the output changes by only about a ten millionth.

Takeaways

First and foremost, this model shows that there is not a great learnable relationship mapping from city passenger throughput and whether two major airlines service said city to whether a third major airline should service the city. Realizing this, there must exist many more reasons which affect whether a major airline services a city (e.g. acquisitions of smaller airlines). This model in particular tends to output only high values to limit error, which might be a result of selection of data, as only the top one hundred and seventy-nine U.S. cities serve as input.

The model does seem to weight passengers numbers with a higher importance than whether other airlines service a city. This model serves as an example of trying to find relationships with too little data (i.e. throwing any data at ML and hoping for a good result). It seems that without much more feature data, a mathematically derived empirical model will perform far better (Berry 1992, 914-915).

MODEL 2.2: TOTAL AIRLINE SERVICE OF CITY

Model 2.2 trains almost identically to Model 2.1 in its approach and data sources. This model attempts to determine how many of the three major airlines should service a city based on its passenger throughput.

Approach

This second version of the second model implements a linear classifier with four classes (i.e. 0, 1, 2, 3) using the Adam optimizer. Each class represents how many of the three major airlines should operate in the input city. Therefore, the output target is the sum of the three airlines' servicing values. The error detection for this model utilizes MAE, as well. In order to check the output, the Colab sums each class' value (i.e. 0, 1, 2, 3) multiplied by the model's expected probability of that class, resulting in a single value of how many of the three airlines should service the city.

Results

The model does reduce error somewhat, as seen in Figure 5 on the next page. However, just as in Model 2.1, one can point to a pretty dismal ending value of MAE. Considering the larger scale of Model 2.2, its MAE is slightly better than that of Model 2.1.

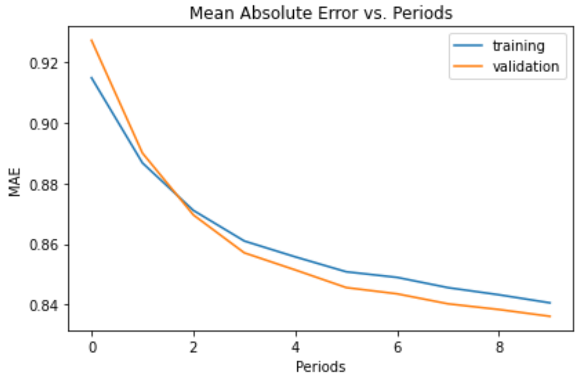

Figure 5: Training on Model 2.2

Similar to Model 2.1, the model has found that outputting high values reduces error. Again, the dataset is biased toward bigger airports. In modifying the prediction input of passengers, one can see the direct relationship between passenger count and number of servicing airlines. Pointing out the same biasing error of this model, inputting a value of zero passengers produces a predicted output of approximately 1.50 airlines.

Takeaways

Mirroring Model 2.1, this model shows that there is not a great learnable relationship mapping from city passenger throughput to how many of three major airlines should service a city. Nonetheless, the model does show that there is a direct correlation.

MODEL 3: AIRCRAFT UTILIZATION RATE

The third model attempts to determine an aircraft's daily utilization rate in hours based on a variety of input features. This experiment makes use of the same aircraft data as in Model 1, except all four data points comprise this model's input features. Utilization rates taken from a China Eastern Airlines SEC filing make up the model's targeted outputs (Securities and Exchange Commission 2016).

Approach

The aircraft utilization model employs a linear regressor and Adam optimizer combination, just as in Model 1. Input features include standard 2-class seating quantity, maximum range, years between its first flight and 2017, and its list price. Similar to Model 1, there are very few data points under study, making ML techniques much less effective.

Results

Training with the four input features on aircraft utilization rates results in a decent reduction in RMSE, as shown in Figure 6 below.

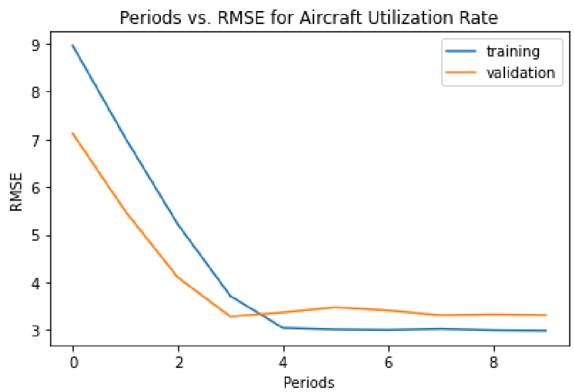

Figure 6: Training on Four Features with Model 3

As expected, the model reduces training error to a lower minimum than that of validation error; with so few examples, a sizable difference in unlikely. Undertaking the training with the sole use of each feature, the ability of each variable to reduce error is found – see Figure 7.

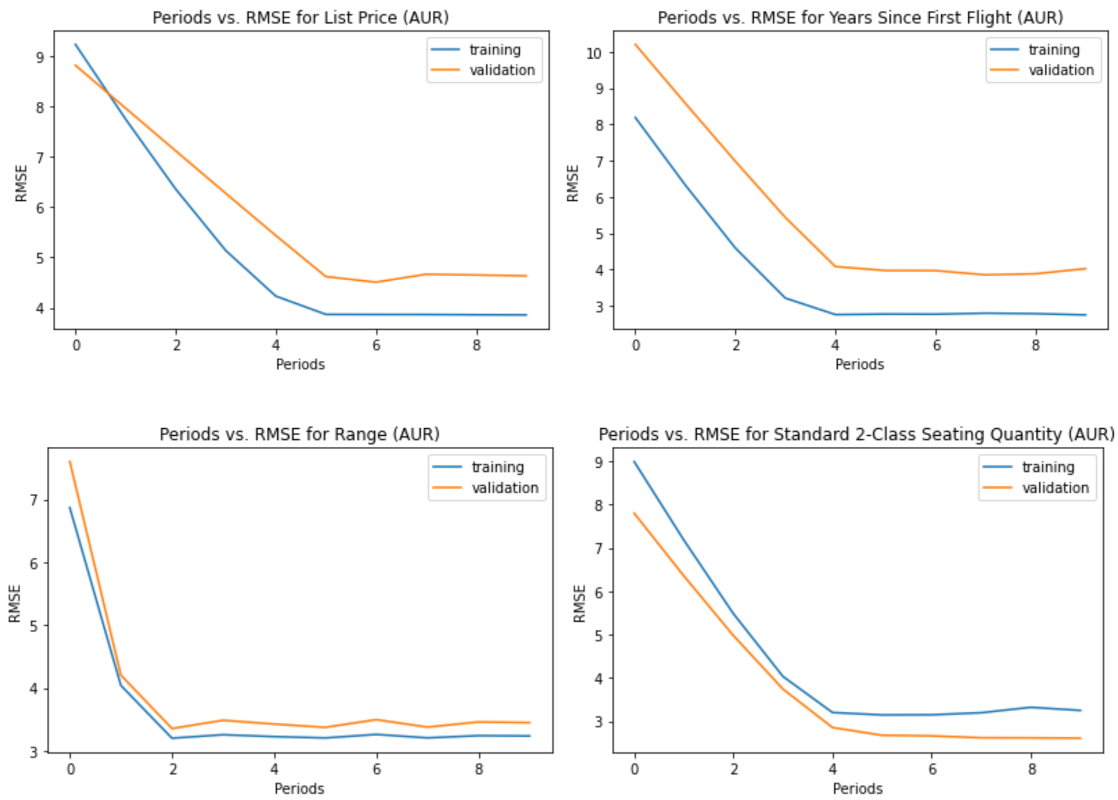

Figure 7: Training on Each of Four Features with Model 3

The best features in Model 1 turn out to be the best features in Model 3 as well. The best to worst features are seating, maximum range, years since first flight, and list price. The two best, seating and range, appear very similar to the plot which includes all four features.

Takeaways

Model 3 demonstrates that there is a somewhat learnable relationship mapping from the four features under study to the daily utilization rate of an aircraft for China Eastern Airlines. Although this experiment holds little data from which the model can learn, interesting insights still result. In tandem with the first model, this model demonstrates that the same factors which correlate with the cost of an aircraft may also correlate with its utilization rate.

OVERALL OBSERVATIONS AND POTENTIAL APPLICATIONS

While machine learning's power comes from looking at situations with many more data points, these simple experiments demonstrate the potential relationships that ML can find. If airlines published more industry data, then models similar in development to these could produce far more interesting and relevant relationships. Obviously, due to the competitive advantage of internal information, airlines do not release specific aircraft and route information. Along the same lines, aircraft manufacturers do not release their proprietary data. Therefore, only those within these companies can truly test and use ML's potentially profitable power within these tightly knit industries.

Improving Current Situations

In both the airline and aircraft industries, areas of potential improvement abound. The fastest and easiest applications of machine learning are within existing processes. In order to garner company-wide support for such a new methodology of determining relationships and predicting outputs, strategists should try to find improvements within current operations. Based on ML's already implemented and abstracted functionality, those with a technical background and involved in data analytics in these industries can easily supplement their methods with this technology. What will begin as a supplement may eventually lend itself to becoming the primary methodology.

Supplementing Future Growth

One can find another very interesting use case in that of airline and aircraft companies within areas of new growth. Without the pre-determined existing situations typical of highly developed nations, the aforementioned BRIC countries can utilize ML to establish best practices from the beginning. Fewer entrenched structural limitations should exist in these areas' industries. Seeing how one can view interesting relationships with simple SEC-filed data by China Eastern Airlines, the potential applications for China's industry insiders are numerous. Machine learning pipelines can actively provide data to models such that they continuously learn and serve more accurate predictions and correlations. With the air industries rapidly changing due to COVID-19, ML can prove invaluable.

SUGGESTED FUTURE RESEARCH

As the models in this research remain quite primitive, a variety of potential next steps can help explore all machine learning has to offer.

Diversity of ML Model Types

Within this research, the models only made use of the linear type of estimators, although they use both the regressor and classifier sub-types. Due to lack of data and intended simplicity of this paper, other more complicated networks are not applied. However, additional research on these subjects is warranted.

For example, multiple heads can combine to form a broader network – essentially a model with sub-models providing intermediate inputs. The TensorFlow documentation lists numerous additional types of estimators, including nonlinear ones and those intended for deeper neural networks (TensorFlow n.d.). Deeper neural networks apply many hidden layers of nodes which contribute to the final output. With that scale of a network, these typically require massive amounts of data in order to have all of their nodes learn enough to be applicable.

Greater Quantities of Data

Lack of data underlies most issues with potential ML applications. While this research focuses on data available from both the United States and China in order to look for comparisons, future research can look specifically at sources which have more data (e.g. data specifically from America's Bureau of Transportation Statistics). Further, those with insider access to airline or aircraft manufacturing data have much more interesting information with which to create ML models for this industry.

Policy-Making and Enforcing

Beyond both airlines and aircraft manufacturers, air policy makers and enforcers can benefit from machine learning's application. Since ML effectively detects relationships among data, one can utilize models to detect violations of policy by looking for abnormalities, or situations where an expected output vastly differs from the reality. For example, models can aid efforts in searching for anti-competitive behaviors such as price gouging among airlines.

CONCLUSION

Exploring the underpinnings and simple applications of machine learning greatly evidences its high availability and great number of use cases. The models under this research illustrate the finding of simple yet interesting correlations and subsequent prediction ability. With open-source tools such as Google's Colab in tandem with TensorFlow, students, researchers, or anyone interested in general have the ability to utilize this technology at extent.

While powerful and available, ML is not a panacea. The experiments in this study evidence the importance of large quantities of data in providing useful models and results. Further, one of the biggest downsides of ML is that of its inexplicable nature. Especially with regard to deeper neural networks, a model will learn appropriate correlations and find ideal weightings, yet a produced prediction does not come along with an explanation. Although machine learning produces interesting results, one cannot simply interpret them as truth.

Nonetheless, ML's abilities are promising for many industries and associated applications. As discussed in this research, the airline and aircraft manufacturing sectors hold many possibilities for this burgeoning technology. From improving current business and general public situations to aiding in future growth opportunities, machine learning holds a lot of potential for these areas. Yet determined is which organizations or corporations will most rapidly apply and extend ML. With China having experienced rapid growth in this sector with such a large population, strategists should be intent on seeing if involved players will widely adopt this methodology.

APPENDIX

In order to run a model, access its link while signed into a Google/Gmail account. Either copy the notebook to your own account, or open it in playground mode. See instructions on notebooks.

Access to Model 1:

https://colab.research.google.com/drive/135psItHoTswVIx8zRnf3QVqsOg1mu4Mw

Access to Model 2.1:

https://colab.research.google.com/drive/1f9C0cYowAIEPbrdctkU7QT6Yg1YE97dT

Access to Model 2.2:

https://colab.research.google.com/drive/1lf5f392jBMLm5PN_Wo2v9sJGcU8RVUV3

Access to Model 3:

https://colab.research.google.com/drive/1jcJ9GhAVtw9T-i0hNLaS14FFOkrWx0Oi

REFERENCES

Airbus. 2017. “2017 price adjustment for Airbus’ modern, fuel-efficient aircraft.” Accessed April 25, 2020. https://www.airbus.com/content/dam/corporate-topics/publications/press-release/new-airbus-list-prices-2017.pdf.

Ausick, Paul. 2017. “Here’s the List Price of All Boeing’s Commercial Airplanes.” Accessed April 25, 2020. https://finance.yahoo.com/news/list-price-boeing-commercial-airplanes-143500555.html.

Bengio, Yoshua. 2012. “Practical Recommendations for Gradient-Based Training of Deep Architectures.” *Arxiv*. 9. https://arxiv.org/pdf/1206.5533.pdf.

Berry, Steven T. 1992. "Estimation of a Model of Entry in the Airline Industry." *Econometrica* 60, no. 4 (1992): 889-917. doi:10.2307/2951571.

Bureau of Transportation Statistics. 2019. “Airport Rankings 2018.” Accessed April 25, 2020. https://www.bts.gov/airport-rankings-2018.

DVB Bank SE. 2017. “An Overview of Commercial Aircraft: 2018-2019.” Accessed April 25, 2020. https://www.dvbbank.com/~/media/Files/D/dvbbank-corp/aviation/dvb-overview-of-commercial-aircraft-2018-2019.pdf.

Fallows, James. 2012. *China Airborne: The Test of China’s Future*. New York: Vintage Books.

Google. n.d. “Colaboratory.” Accessed April 25, 2020. https://research.google.com/colaboratory/faq.html.

Google. n.d. “Machine Learning Crash Course.” Accessed April 25, 2020. https://developers.google.com/machine-learning/crash-course/.

Google. n.d. “Validation.” Accessed April 25, 2020. https://colab.research.google.com/notebooks/mlcc/validation.ipynb.

Hebert, Louis and Ali Taleb. 2011. “Bombardier Aerospace: The CSeries Dilemma”. 6-7. *Ivey Publishing*. https://hbsp.harvard.edu/product/W11176-PDF-ENG.

Kingma, Diederik, and Jimmy Ba. 2014. “Adam: A Method for Stochastic Optimization. International Conference on Learning Representations.” 1-7. *Arxiv*. https://arxiv.org/pdf/1412.6980v8.pdf.

Partow. n.d. “The Global Airport Database.” Accessed April 25, 2020. https://www.partow.net/miscellaneous/airportdatabase/index.html#Downloads.

PricewaterhouseCoopers. n.d. “COVID-19: What It Means for the Aerospace and Defense Industry.” Accessed April 25, 2020. https://www.pwc.com/us/en/library/covid-19/coronavirus-impacts-aerospace-and-defense.html.

Securities and Exchange Commission. 2016. “Form 20-F China Eastern Airlines Corp Ltd.” Accessed April 25, 2020. https://sec.report/Document/0001144204-16-095915/.

Sherali, Hanif, Ki-Hwan Bae, and Mohamed Haouari. 2013. “An Integrated Approach for Airline Flight Selection and Timing, Fleet Assignment, and Aircraft Routing.” *Transportation Science* 47, no. 4: 455-476. https://doi.org/10.1287/trsc.2013.0460.

TensorFlow. n.d. “TensorFlow Core v2.1.0.” Accessed April 25, 2020. https://www.tensorflow.org/api_docs/python/tf/estimator/.

Toolforge. n.d. “GeoHack.” Accessed April 25, 2020. https://tools.wmflabs.org/geohack/.