Lab Report

Title: (Draft 1) Air Traffic Analysis
Notice: Dr. Bryan Runck
Author: Megan Marsolek
Date: 11/10/21

Project Repository: https://github.com/mmarsole/GIS5571FinalProject
Google Drive Link: NA
Time Spent: 16 hours

Abstract

Air Travel has become an increasing and popular means of travel, with airlines expanding and offering more flights to more destinations, it becomes a challenge to monitor and safely manage airspace to avoid collisions around highly frequented areas like airports. There are many regulations in place to assure travel safety, but it has the potential to benefit from incorporating Machine Leaning (ML) to better estimate a plane's near future location based on constantly changing variables (weather) and fixed restrictions (airspace classifications).

Using a plane's current and past locations, the surrounding weather, and current airspace I intend to track and predict a plane's location in 3D space over time (lat., lon., and alt.). Relying on three main sources for data, I will convert and prep the data to read into a ML model. I will then train and test the accuracy by comparing it to the recorded flight path. In the end I hope to visually display a sample of the ML predicted flight path with that of the real flight path and any surrounding weather and airspace classes for user interoperability.

Problem Statement

The intention is to predict a plane's flight path based on data that may influence a plane's course, such as Airspace classification or weather. Given I have access to its present and past locations as it flies (via ADS-B data) I hope to predict where it will likely be in 3D space for some specified time in the future (seconds to minutes).

Table 1. Relevant (attainable) variables for the problem

#	Requirement	Defined As	(Spatial) Data	Attribute Data	Dataset	Preparation
1	Air craft locations	Point data (Lat., Long., Alt.) from ADS-B (currently in csv file)	Point data in csv (extracted from TAR files)	lat, lon, geoaltitude, heading, velocity, time	ADS-B Data	Subset the data to isolate a couple of flight paths (for instance flights that travel through Chicago Class B airspace). Convert lat lon from excel to point data in Arc.
2	Weather	Measurements within each cell is 8 bit Base Reflectivity 0.5dbz (used to measure the severity of weather, the	Raster Sizes : 12200x5400 pixels (for the continental US),	Pixel dbz measure in each cell	IEM NexRad Mosaic	Possible web scrapping (extracting archived raster images

		higher the dbz the more the severe the weather)	pixel resolution is 0.005 degrees by 0.005 degrees (approximately 55mx555m)			over a specified time period) Matching CRS with rest of data, subsetting/cropping data to relevant spatial extant
3	Airspace Classification B	Regulated airspace across the US with flying permissions that dictat flight protocol (subset to focus only on class B airspace)	2D shapefile (vector) or access to a 3D Google kmz file	UPPER_VAL, LOWER_VAL, NAME, SECTOR, LEVEL	FAA Airspace Shapefiles	Matching CRS with all other data, Subsetting data to just class B

Input Data

This project's initial steps focused on subsetting and extracting corresponding samples for all the data (for instance making sure that the weather data corresponds with the same time frame that the ADSB data was collected). The flight data is a snapshot of all global flights between 12:00:10 AM and 12:59:50 AM on May 25th of 2020 (recorded in GMT time), and 13 corresponding rasters (GeoTiffs) have recorded the weather in each cell as dBZ measurements which represent the strength of returned energy to the radar (US Department of Commerce, n.d.). The raster files match the same time frame as the ADSB data but are recorded in 5 min increments.

For simplicity, the flight data (ADSB data) was subsetted to several smaller datasets, first by reducing the data to only contain flight paths that intersected with Chicago's Class B airspace (this airspace surrounds O'Hare International Airport and Chicago Midway International Airport). This step reduced the csv file from its original 723,098 points (over 675,000 flight paths) to 11,697 points (57 flight paths), a more manageable dataset size. Then I subsetted the data to individual flight paths (based on the icao24 identifier flight number). These isolated flight paths were then fed into the ML model to predict their immediate trajectory. These individual flight paths contained between 150 points to 450 points.

Table 2. Data Access

#	Title	Purpose in Analysis	Link to Source
1	ADS-B	ADS-B data records the location of plane at a given time in points (lat., long. Alt.). Its location can help predict where it might be at some time in the future. It also contains other attributes: make/model of aircraft, time, flight number, etc.	Open Sky Network
2	IEM NexRad Mosaic	This is Raster data for weather (radar). It will indicate where severe weather is located that may influence some flight paths. (smallest measure of time is 5 mins)	Iowa State University
3	Class B,C,D,E Airspace Shape Files	Define where Airspace classifications are in 3D space that may affect flight paths (represented as 2D but contains attributes that describe its min and max altitudes).	Federal Aviation Administration (FAA)

Methods

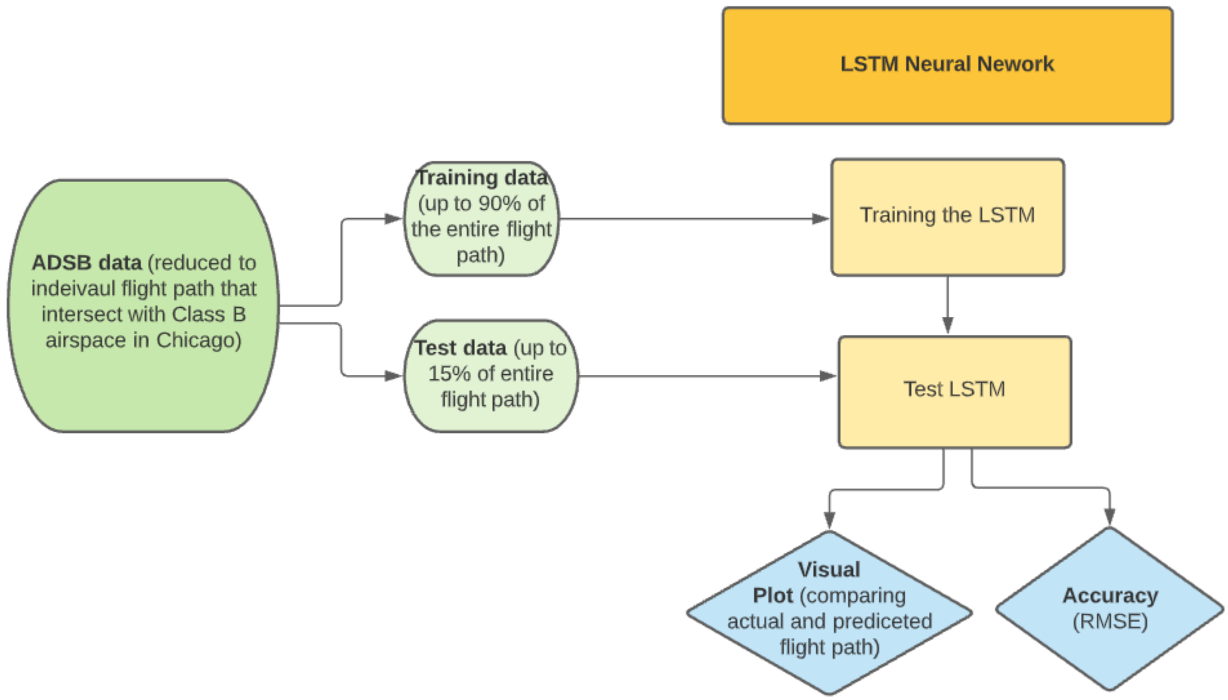

Figure 1. Data Flow Diagram

Based on research I have identified 2 possible Machine Learning (ML) algorithms that are suitable for predicting time series data (1) Long Short Term Memory Neural Network (LSTM) and (2) Linear Regression. The latter is a little easier to understand as it is uses predictors (independent variables) to predict independent variables based upon the linear formula y = c +b(x). Meanwhile, LSTM neural networks are rooted in deep learning and "capable of learning and memorizing long terms dependencies" (Biswal, 2021). My initial results are only for LSTM based models. The data fed into the LSTM was based upon each individual flight data (no weather context or Airspace classification were added at this time). The input variables for the LSTM data included: velocity (m/s), heading (track angle measured up to 360 degrees), vertrate (vertical speed m/s), lastposupdate (the age of the current position), and the lastcontact (the last recorded time of the previous signal/point). These variables were then used to predict latitude and longitude (only a 2D prediction output).

Based on *Figure 1* you can see how I divided the flight data into training and testing data. I often used a majority of the data as training data (between 85-90% of the available flight's point data) and the remainder to test the model. Once the model was trained, I fed in all the data (training and testing) so that I could plot the predicted flight path and the actual flight path to visually compare the results. As an added measure to test the LSTM accuracy, I computed the Root Mean Squared Error (RMSE) for entire model and then just for the testing data. What I hope to see is a similar RMSE value for the test data and the training data, and ideally at low values (all my RMSE values would be measured in decimal degrees since this is the unit of measure for my latitude and longitude). The smaller the RMSE the better the LSTM.

Results

Figure 2 depicts the predictions and the actual flight path for flight a198e5. This flight is arriving at Chicago Midway International Airport and thus most of its coordinates for the end of its flight are stationary. You can see this reflected in the graph on the right, where we see two dots. During this flight's last 4.5 mins it was sitting in a terminal unloading passengers, thus the blue point are stacked consecutive points over time that do not move. The prediction (seen in orange) is within 1 decimal degree of the actual plane's location, and is stationary but still reflects a poor accuracy as seen in the RMSE values for both the latitude: 0.02564239501953125 decimal degrees and longitude: 0.0373382568359375 decimal degrees. For a general interpretation 2 decimal places is approximately equivalent to measurement in Kilometers (so 0.01 decimal degrees is about 1 km). Here, I can see my LSTM prediction for Figure 2 is off by kilometers.

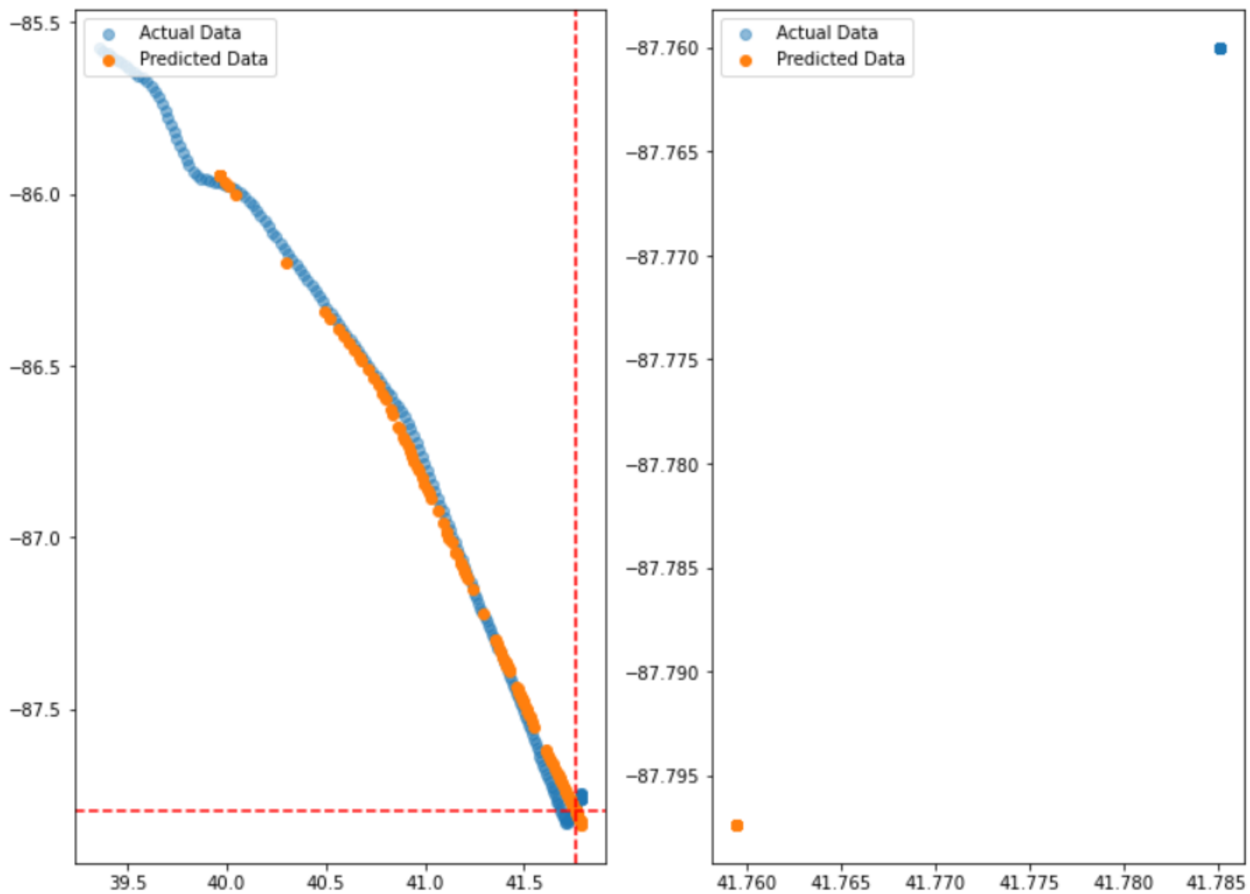

Figure 2: Flight path for a198e5. The left graph displays all the point data for a198e5, where blue is the actual flight path and orange the predicted flight path based on the LSTM. The red dashed lines are the markers that differentiate where the training data and testing data (the lower right quadrant was the testing data, while all the data elsewhere represents the training data). The image on the right is a close up look at the lower right quadrant which is based solely on testing data.

Figure 3 depicts the LSTM prediction for flight 89902f. Which is neither an arrival nor a departure from any airport in Chicago, rather it appears to be a flight with a consistent cruising altitude and a NW bound trajectory. Based on the graph on the left I would be inclined to think it was performing well but looking closer at the testing data (see the graph on the right) it is apparent that the predicted data seems to be moving in the wrong direction over time (the dark red indicates older time intervals while the yellow reflects most current time intervals, based on this it appears the prediction in the right graph moves South before backtracking North again). The RMSE error for the testing latitude data: 0.13827374217023908 and the longitude error: 0.17702699681397477. This indicates that the LSTM model is having more difficulty with the longitude than the latitude since the error for the latitude testing data is higher than the latitude (but both are still within a similar range of error within the 2nd decimal place).

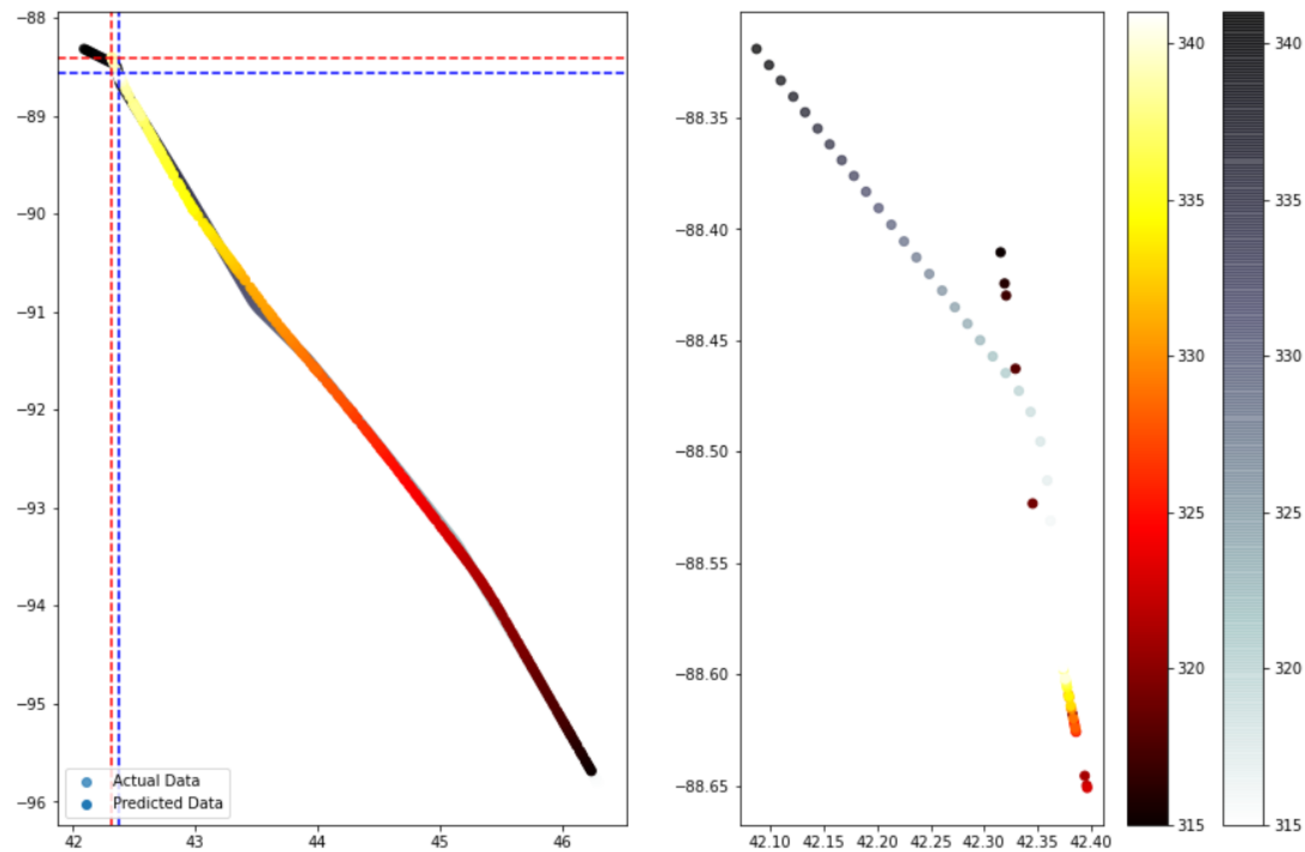

Figure 3: Flight path for 89902f. The left graph displays all the data associated with flight 89902f. In this case, the actual data is displayed as a gradient that starts as white and progresses to black (displays flight travel over time), while the predicted data based on its trained LSTM data starts with dark red and progresses to yellow over time. In this graph the lower right quadrant contains all of the training data (this can explain why the predicted path looks very similar to the actual data) whereas the upper left quadrant conveys the test data. You can see a close up of the test data in the graph on the right. On close inspection you can see that the prediction seems to change direction. It predicts the aircrafts path to a downward (South) direction and then flips its heading and predicts it to traveling north bound.

The last LSTM model trained was based on additional data for flight 89902f (same flight path form *Figure 3*). In this case, I made 5 new variables based on the already existing latitude, longitude, and velocity variables (see the equations). Three new variables were for the differences between 1 timestep and calculated for Latitude, Longitude and Velocity. This means I subtracted the 'current time step' from its immediate president timestep. This would provide the distance covered along the x and y axis as well as any change in velocity. Additionally, I made 2 more variables for longitude and latitude that reflect the difference between the current timestep and the timestep after the immediate previous (essentially, looking at the distance between locations over twice as much time as the previously made variables).

Equations for new Variables.

$$Latitude_2\text{-}Latitude_1 = \text{Difference between 1 timestep}$$
$$Latitude_3\text{-}Latitude_1 = \text{Difference between 2 timestep}$$

$$Longitude_2\text{-}Longitude_1 = \text{Difference between 1 timestep}$$
$$Longitude_3\text{-}Longitude_1 = \text{Difference between 2 timestep}$$

$$Velocity_2\text{-}Velocity_1 = \text{Difference between 1 timestep}$$

After making these additional variables, I then fed in the first 300 points from 89902f flight path as training data and used the remaining 40 data points at the end of its flight as testing data. Based on Figure 4, you can see all of the actual flight path (in blue) and the predicted flight path (in orange) in the left graph. Once again, it appears the training data adheres or follows the actual path well (as seen in the lower right quadrant of the left graph) but begins to diverge when it comes to the test data (see the right- hand graph). The data in the right graph has been colored as a gradient to better convey time and show how the LSTM predicts the flight path. The scale bar that starts red and turns to yellow represents the predicted path, while the grey scale bar represents the actual flight path. If we follow the color trail for the predicted flight, we start very close to the actual flight path (look for the darkest red points) and then proceed to travel South but backtrack Northward and finally jump to the middle and begin a new linear path that is generally NW bound (as seen in the lighter yellow values). This path doesn't seem to follow a general linear path but fluctuates in direction. Looking once again at the RMSE for the test data we see the highest error of all three of our LSTM models yet with latitude's error at 0.55010530 decimal degrees and longitude's error at 1.08232746 decimal degrees. We can also see longitude has almost twice the amount of error when compared to the latitude, and thus corroborate that the LSTM model is struggling with its vertical axis predictions (this explains the large jumps in the North and South directions.

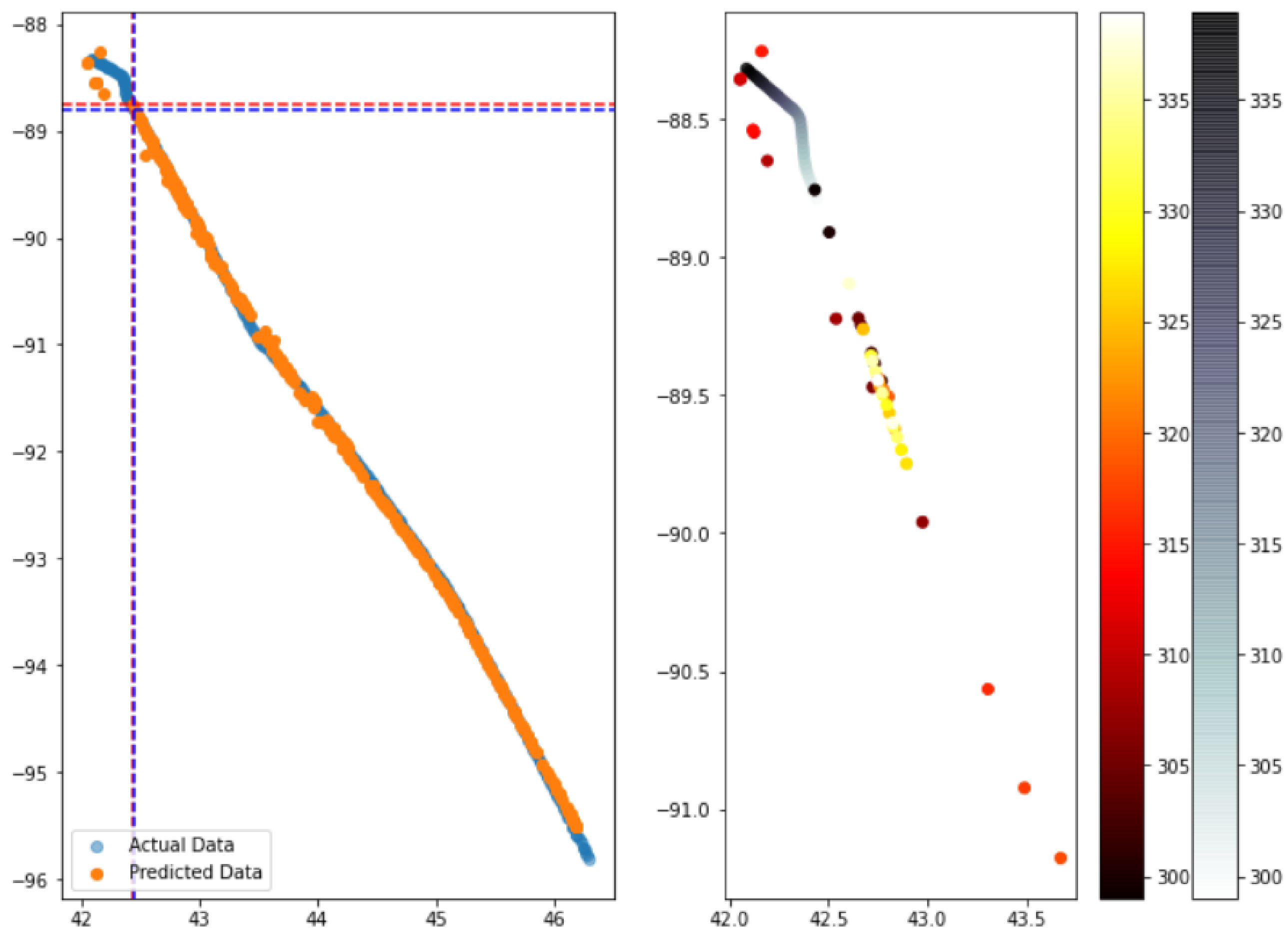

Figure 4: Flight Path for 89902f with supplemental data. This LSTM was given extra data for flight 89902f, 5 more variables were constructed based on the change between the current latitude and its immediate previous latitude (lat2 -lat1), the change between current latitude and its position 2 intervals past (lat3-lat1) and similarly for longitude (long2-long1 and long 3-long1) and velocity (v2-v1, but only for its immediate past interval).

Results Verification

The results for this report are predictions made by LSTM models for 2D space (latitude and longitude). Based on the visuals (*Figure 2 - Figure 4*) the LSTM flight predictions are not as good as I would have hoped, but I did corroborate the accuracy by calculating the RMSE for each Model (both for the testing data and then for all the input data within the model).

Discussion and Conclusion

Overall, I was excited to see results from the LSTM, but am disappointed with its lack of accuracy. As seen in the figures above (Figure 3 and Figure 4) it appears that the LSTM predicts flight trajectory in the opposite direction than the general flight's path (and actual flight 89902f appears to be relatively linear). I don't think adding the extra variables (expressing the difference between time intervals for latitude, longitude, and velocity) were of any significant benefit. Based on the performance of the LSTM in these three examples, I think it might be worthwhile investigating the use of linear regressions, since the flights depicted tend to be linear themselves. The LSTM seems to have the most trouble with predicting Longitude values (these tend to have the largest RMSE error in the testing data).

I do not expect that I'll be able to improve the LSTM performance, but desire to experiment more with changing current parameters in the LSTM such as changing the number of hidden layers, the optimizer, and/or learning rate. I do not know exactly how these parameters affect the predictions, but know the current LSTMs are built with two layers, the Adam optimizer, and have a learning rate of 0.001. I suspect that I may find some possible relationships if I continue to fiddle with these values. Another proposition is to reduce my predictions to a smaller time interval. In the above figures I predicted trajectories 27 to 40 timesteps into the future (approximately 4.5 to 7 mins in the future), maybe I should focus on 1 minute into the future (about 6 timesteps) and see if the accuracy improves or at the very least is related to time (I want to know if accuracy decrease as time increases).

I would also like to try training the LSTM on multiple flight paths instead of just one. The results explained in this report were only based on each individual flight's data. It might improve the LSTM if I fed it more flight data and then tested it on each flight after (this means I'd have to add the 'icao24' variable to the input data to distinguish between flights). Given more time, I would then experiment with predicting the altitude as mentioned in the beginning of this report and integrating weather data. The intention is to track the plane in 3D space (since a plane's flight can interest in 2D space and not crash provided they are at different altitudes), and account for weather's effect on a flight's path. My current challenge is figuring out how to add the weather data to the LSTM, given that relevant weather data kilometers away from the plane's current position can cause a flight to change its path.

References

Biswal, A. (2021, October 28). *Top 10 Deep Learning Algorithms You Should Know in (2021)*. Simplilearn.com. https://www.simplilearn.com/tutorials/deep-learning-tutorial/deep-learning-algorithm

Sakyi-Gyinae, M. K. (2019). *A Machine Learning Approach to Evaluating Aircraft Deviations from Planned Routes* (p. 73) [Thesis]. https://repository.tudelft.nl/islandora/object/uuid%3A274b4386-539a-4193-80e9-f120c8d4832e

US Department of Commerce, N. (n.d.). *NWS JetStream - NWS Radar Images: Reflectivity*. Www.weather.gov. https://www.weather.gov/jetstream/refl

Self-score

Category	Description	Points Possible	Score
Structural Elements	All elements of a lab report are included **(2 points each)**: Title, Notice: Dr. Bryan Runck, Author, Project Repository, Date, Abstract, Problem Statement, Input Data w/ tables, Methods w/ Data, Flow Diagrams, Results, Results Verification, Discussion and Conclusion, References in common format, Self-score	28	
Clarity of	Each element above is executed at a professional level so that someone can understand the goal, data, methods, results, and their	24	

Content	validity and implications in a 5 minute reading at a cursory-level, and in a 30 minute meeting at a deep level **(12 points)**. There is a clear connection from data to results to discussion and conclusion **(12 points)**.		
Reproducibility	Results are completely reproducible by someone with basic GIS training. There is no ambiguity in data flow or rationale for data operations. Every step is documented and justified.	28	
Verification	Results are correct in that they have been verified in comparison to some standard. The standard is clearly stated **(10 points)**, the method of comparison is clearly stated **(5 points)**, and the result of verification is clearly stated **(5 points)**.	20	
		100	