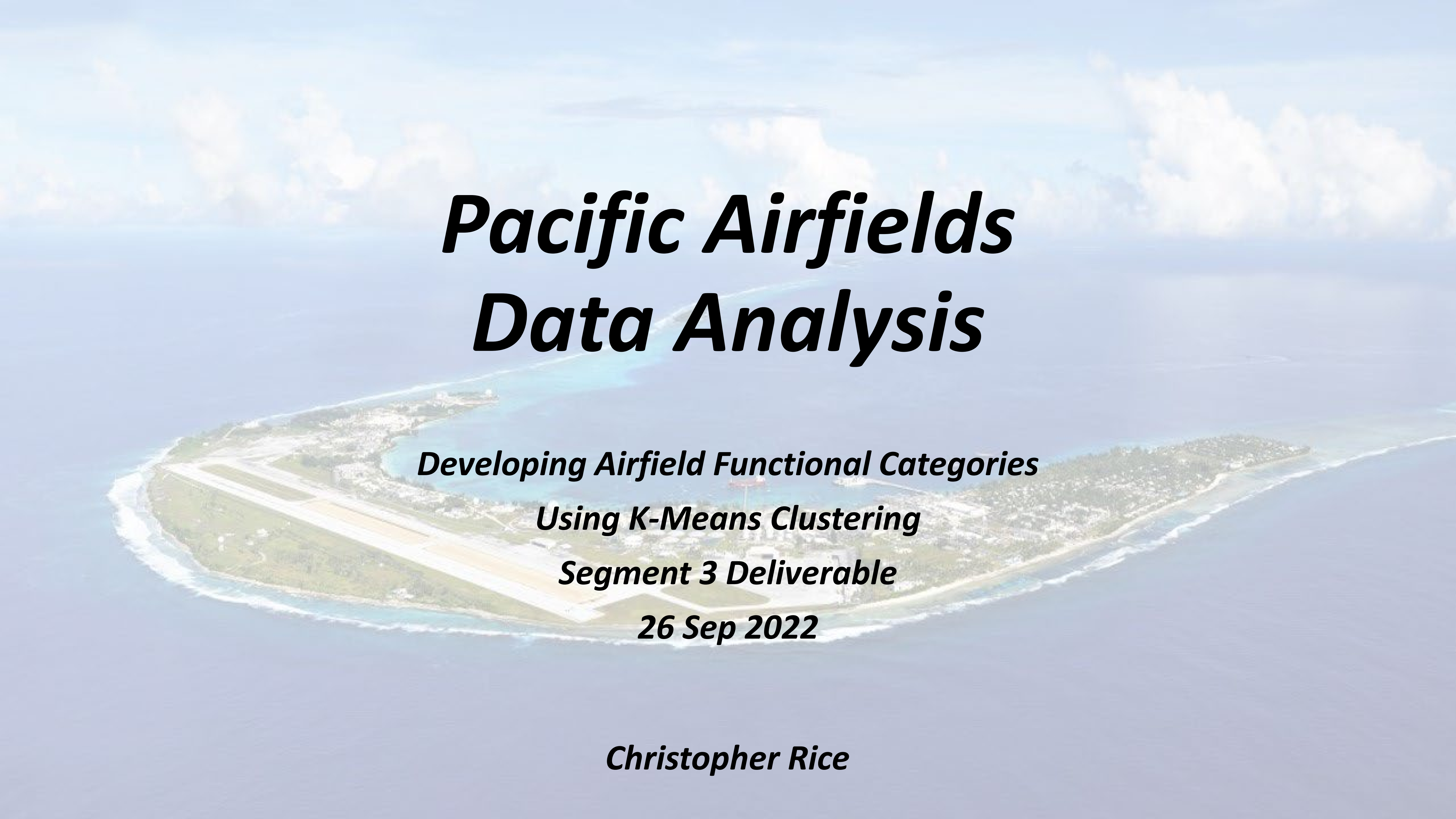


Pacific Airfields Data Analysis

Developing Airfield Functional Categories

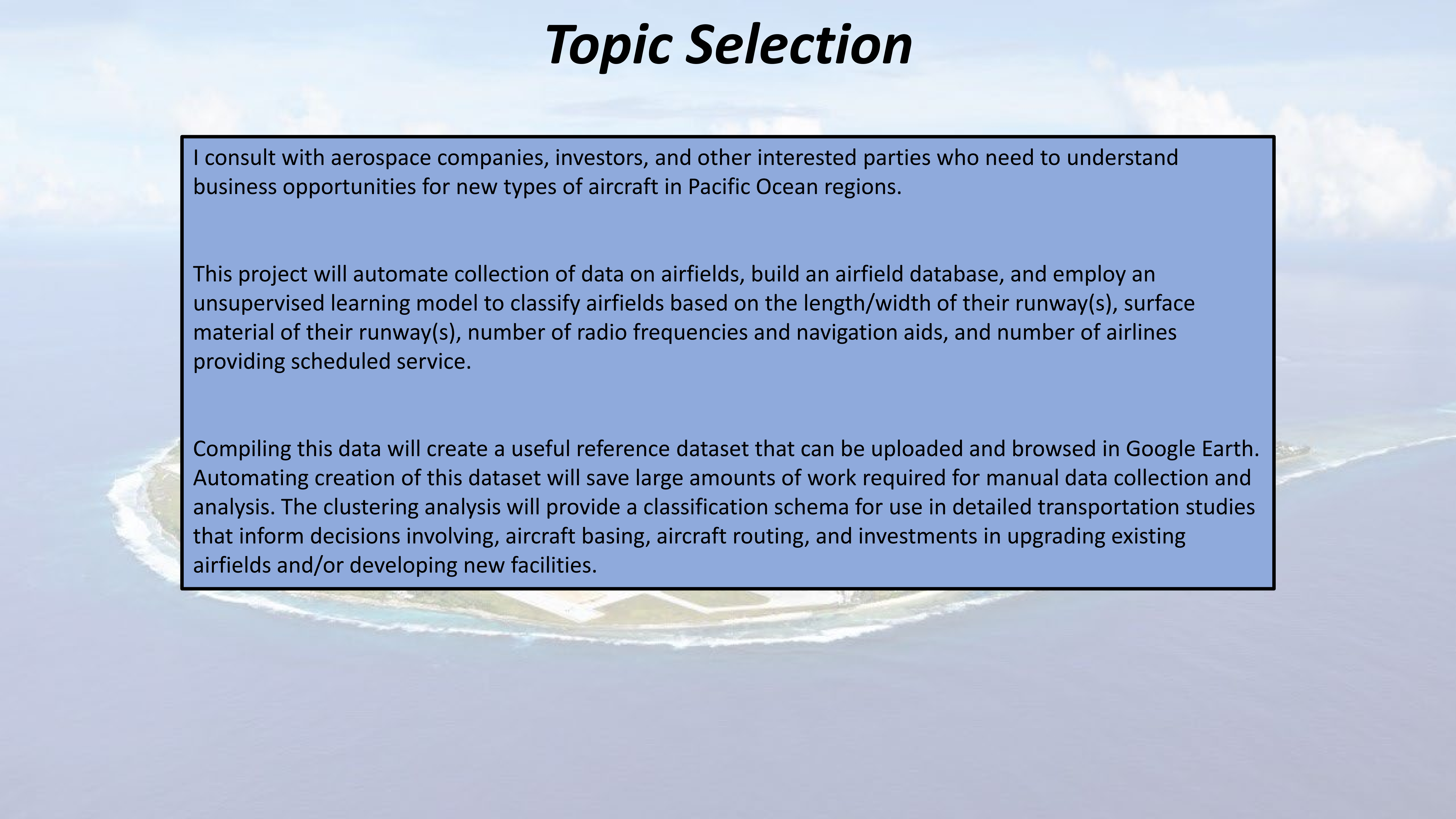
Using K-Means Clustering

Segment 3 Deliverable

26 Sep 2022

Christopher Rice

Topic Selection

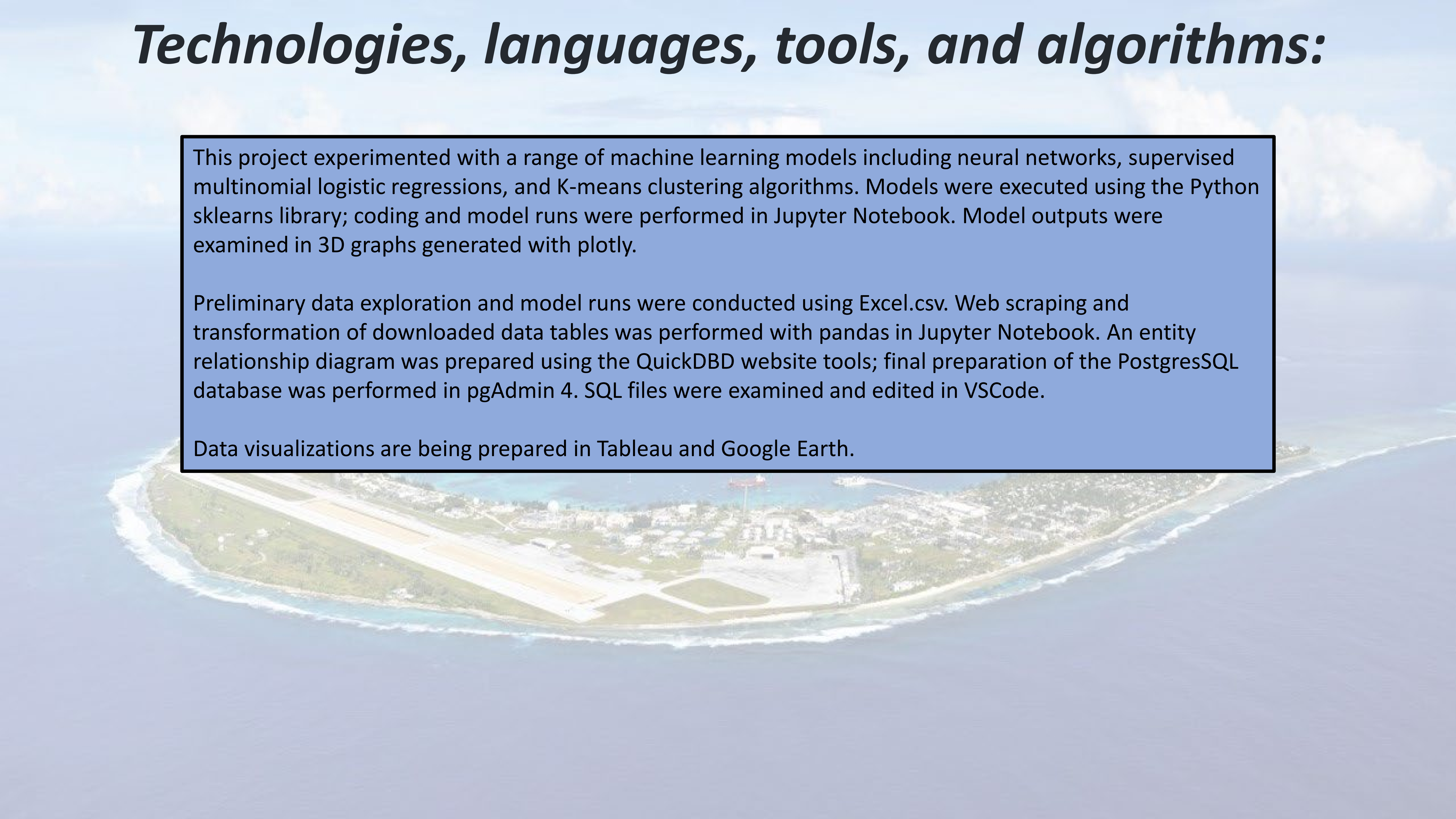


I consult with aerospace companies, investors, and other interested parties who need to understand business opportunities for new types of aircraft in Pacific Ocean regions.

This project will automate collection of data on airfields, build an airfield database, and employ an unsupervised learning model to classify airfields based on the length/width of their runway(s), surface material of their runway(s), number of radio frequencies and navigation aids, and number of airlines providing scheduled service.

Compiling this data will create a useful reference dataset that can be uploaded and browsed in Google Earth. Automating creation of this dataset will save large amounts of work required for manual data collection and analysis. The clustering analysis will provide a classification schema for use in detailed transportation studies that inform decisions involving, aircraft basing, aircraft routing, and investments in upgrading existing airfields and/or developing new facilities.

Technologies, languages, tools, and algorithms:



This project experimented with a range of machine learning models including neural networks, supervised multinomial logistic regressions, and K-means clustering algorithms. Models were executed using the Python sklearns library; coding and model runs were performed in Jupyter Notebook. Model outputs were examined in 3D graphs generated with plotly.

Preliminary data exploration and model runs were conducted using Excel.csv. Web scraping and transformation of downloaded data tables was performed with pandas in Jupyter Notebook. An entity relationship diagram was prepared using the QuickDBD website tools; final preparation of the PostgresSQL database was performed in pgAdmin 4. SQL files were examined and edited in VSCode.

Data visualizations are being prepared in Tableau and Google Earth.

Data Sources

The data required for this study comes from two principal sources: Airportdatabase.net and Wikipedia. Google Earth can also be used to visually inspect for missing variables.

Airportdatabase.net source allows us to identify all of the airfields in a selected country or region and to collect airfield data including:

Airfield name, GPS and IATA codes
Latitude, longitude
Elevation
Runway length, width
Number and frequencies of radio stations and navigation aids
Names of airlines providing scheduled service
URL for the airfields Wikipedia page

Wikipedia allows us to collect more detailed information including:

Runway surface type

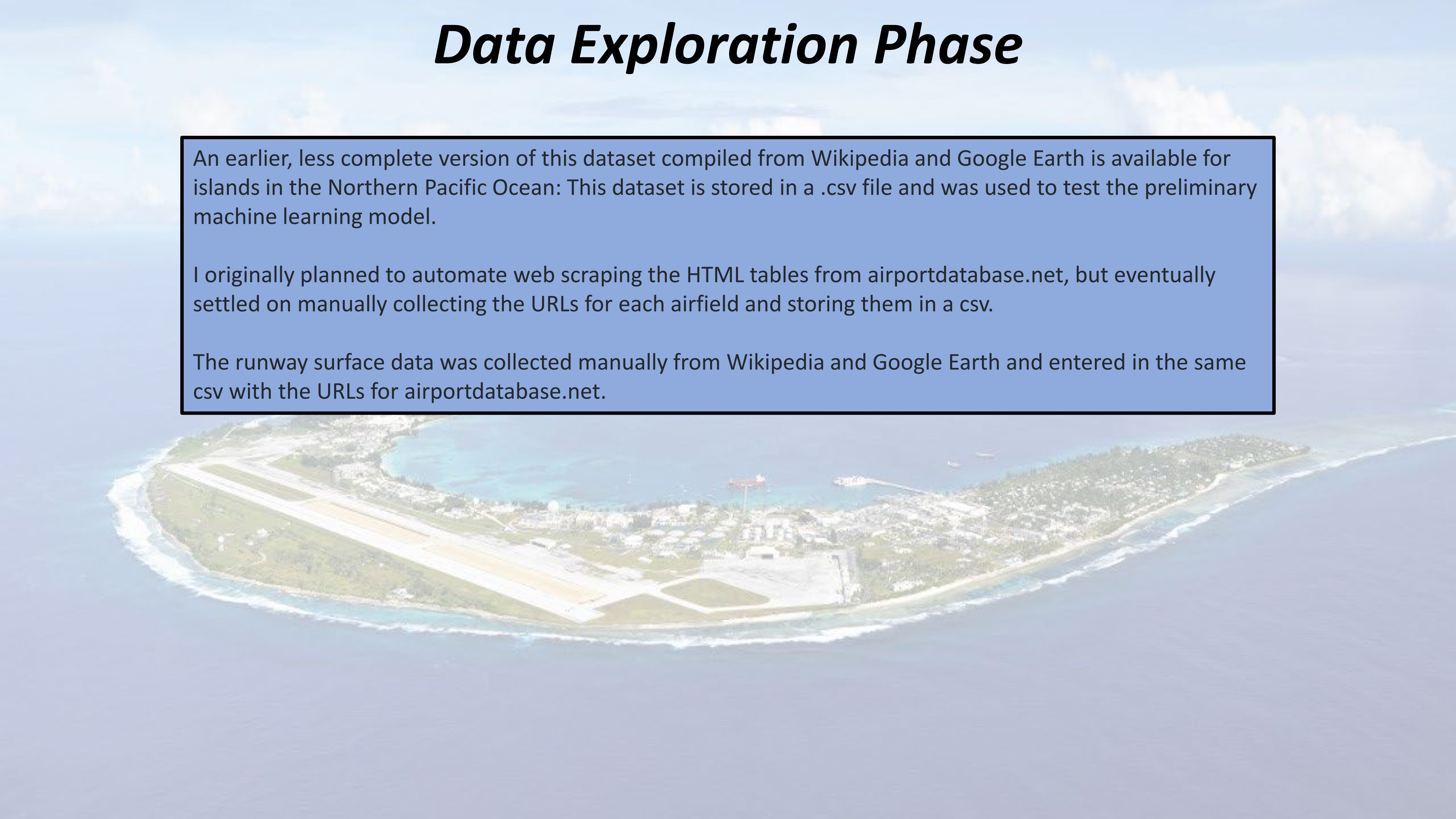
Google Earth allows us to visually assess runway surface and mensurate runway dimensions when no data is available from other sources.

Data Exploration Phase

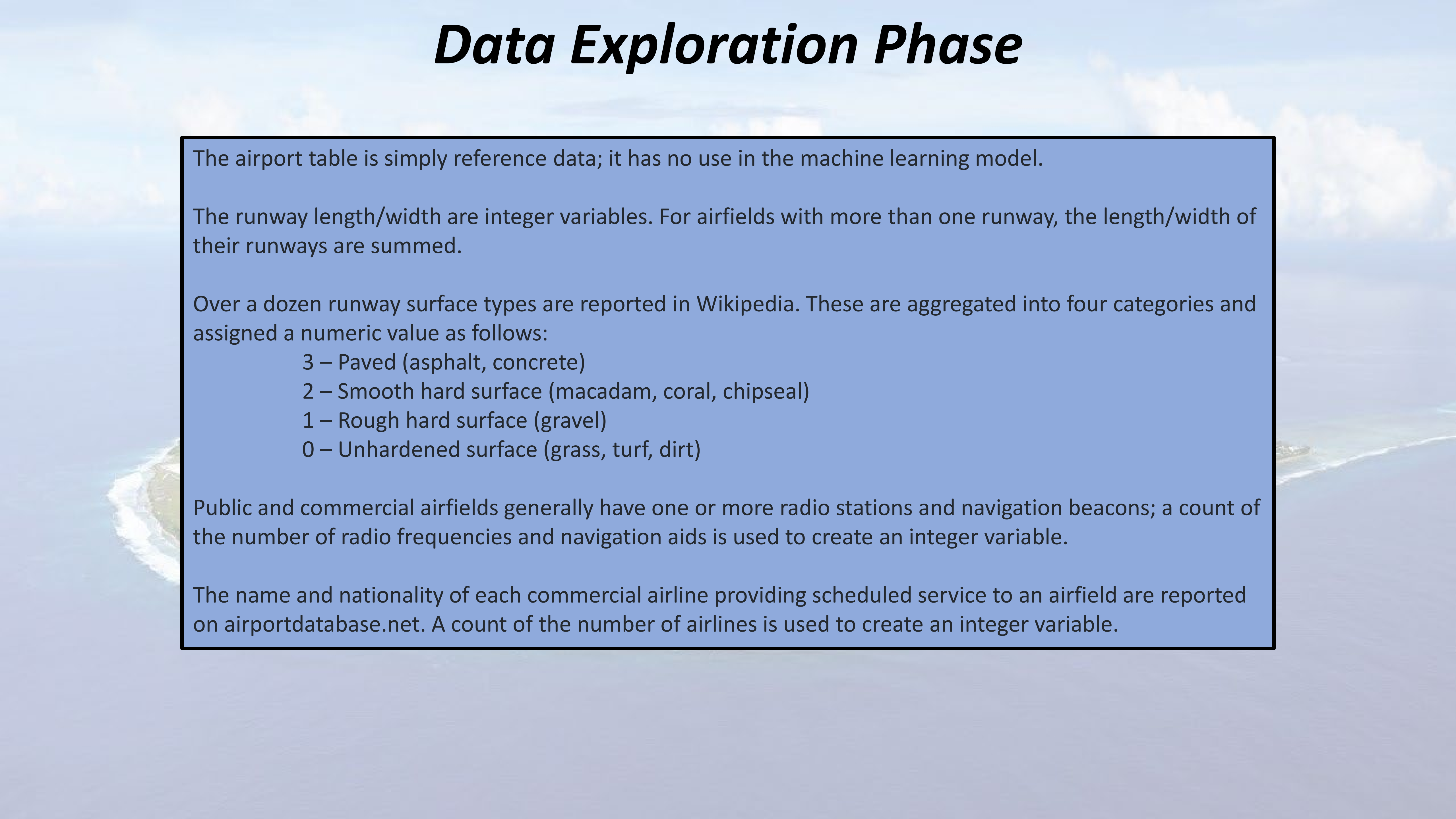
An earlier, less complete version of this dataset compiled from Wikipedia and Google Earth is available for islands in the Northern Pacific Ocean: This dataset is stored in a .csv file and was used to test the preliminary machine learning model.

I originally planned to automate web scraping the HTML tables from airportdatabase.net, but eventually settled on manually collecting the URLs for each airfield and storing them in a csv.

The runway surface data was collected manually from Wikipedia and Google Earth and entered in the same csv with the URLs for airportdatabase.net.



Data Exploration Phase



The airport table is simply reference data; it has no use in the machine learning model.

The runway length/width are integer variables. For airfields with more than one runway, the length/width of their runways are summed.

Over a dozen runway surface types are reported in Wikipedia. These are aggregated into four categories and assigned a numeric value as follows:

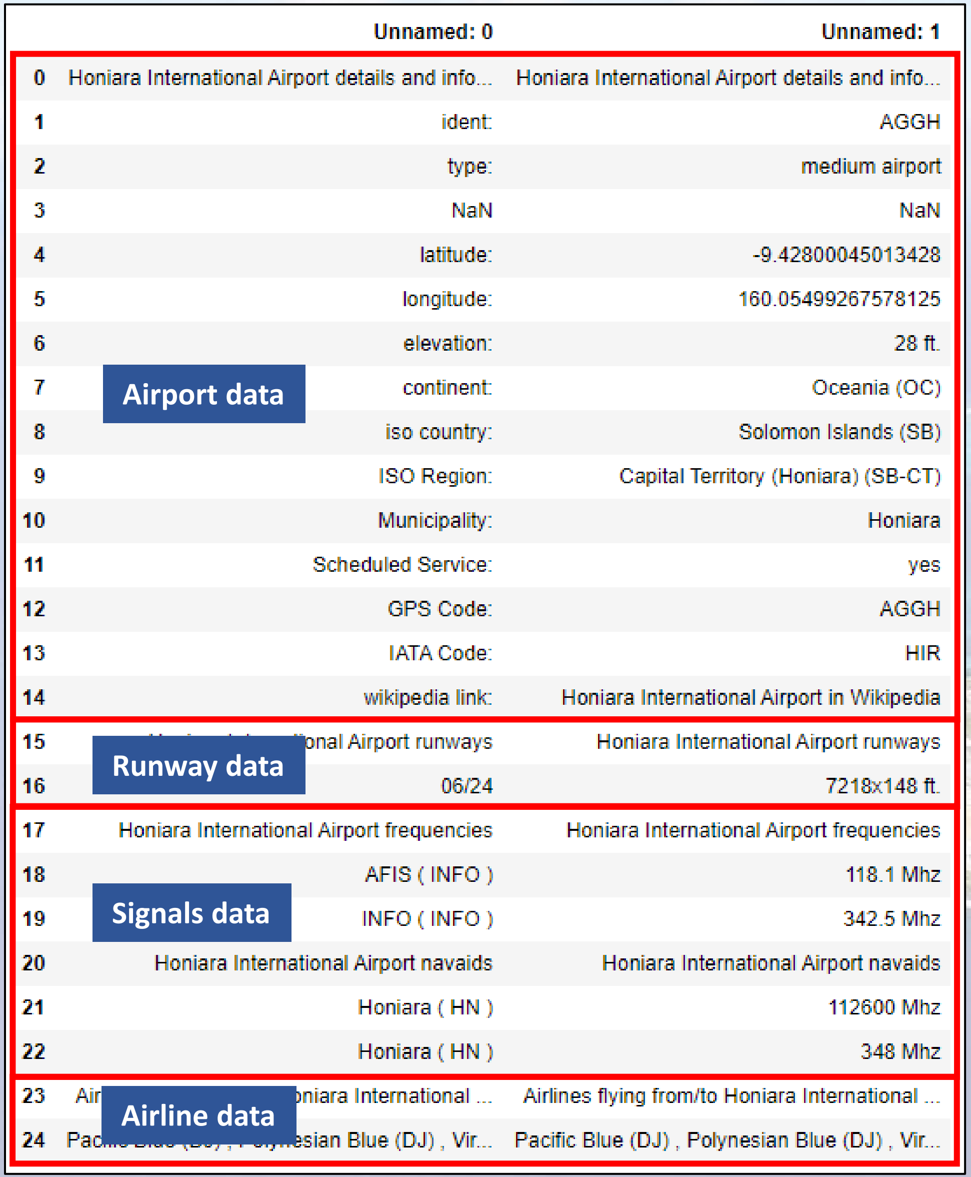
3 – Paved (asphalt, concrete)
2 – Smooth hard surface (macadam, coral, chipseal)
1 – Rough hard surface (gravel)
0 – Unhardened surface (grass, turf, dirt)

Public and commercial airfields generally have one or more radio stations and navigation beacons; a count of the number of radio frequencies and navigation aids is used to create an integer variable.

The name and nationality of each commercial airline providing scheduled service to an airfield are reported on airportdatabase.net. A count of the number of airlines is used to create an integer variable.

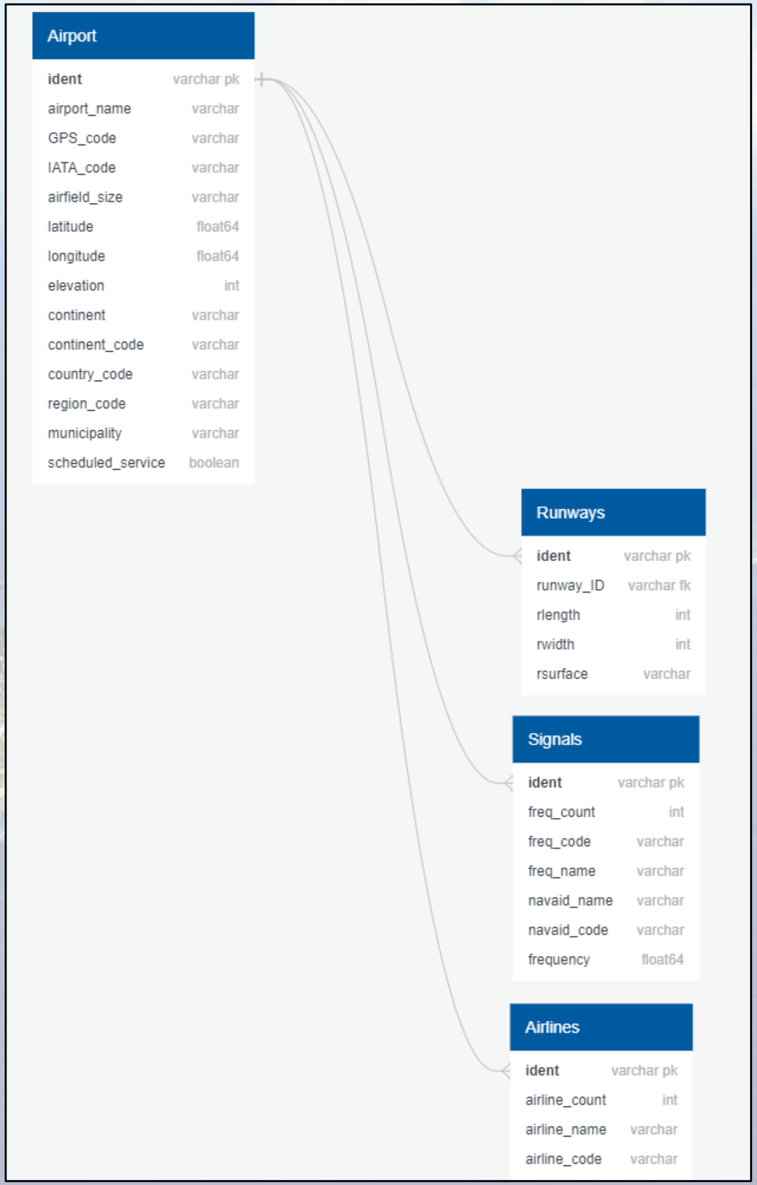
Raw Data -> SQL Database

HTML table from Airportdatabase.net



	Unnamed: 0	Unnamed: 1
0	Honiara International Airport details and info...	Honiara International Airport details and info...
1	ident:	AGGH
2	type:	medium airport
3	NaN	NaN
4	latitude:	-9.42800045013428
5	longitude:	160.05499267578125
6	elevation:	28 ft.
7	continent:	Oceania (OC)
8	iso country:	Solomon Islands (SB)
9	ISO Region:	Capital Territory (Honiara) (SB-CT)
10	Municipality:	Honiara
11	Scheduled Service:	yes
12	GPS Code:	AGGH
13	IATA Code:	HIR
14	wikipedia link:	Honiara International Airport in Wikipedia
15	onal Airport runways	Honiara International Airport runways
16	06/24	7218x148 ft.
17	Honiara International Airport frequencies	Honiara International Airport frequencies
18	AFIS (INFO)	118.1 Mhz
19	INFO (INFO)	342.5 Mhz
20	Honiara International Airport navaids	Honiara International Airport navaids
21	Honiara (HN)	112600 Mhz
22	Honiara (HN)	348 Mhz
23	Air oniara International ...	Airlines flying from/to Honiara International ...
24	Pac esian Blue (DJ) , Vir...	Pacific Blue (DJ) , Polynesian Blue (DJ) , Vir...

ERD Diagram for SQL database



Airport

ident	varchar pk
airport_name	varchar
GPS_code	varchar
IATA_code	varchar
airfield_size	varchar
latitude	float64
longitude	float64
elevation	int
continent	varchar
continent_code	varchar
country_code	varchar
region_code	varchar
municipality	varchar
scheduled_service	boolean

Runways

ident	varchar pk
runway_ID	varchar fk
rlength	int
rwidth	int
rsurface	varchar

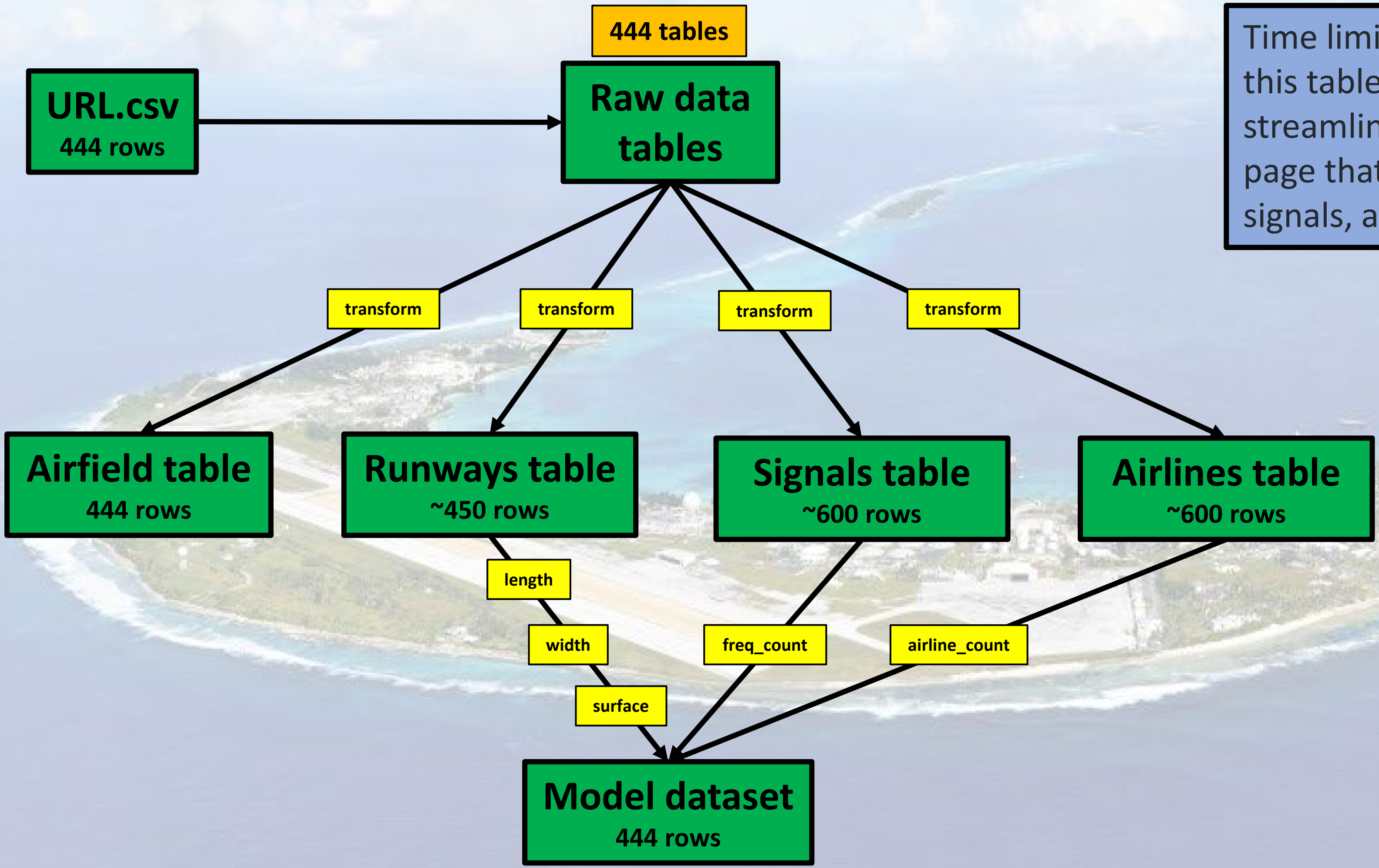
Signals

ident	varchar pk
freq_count	int
freq_code	varchar
freq_name	varchar
navaid_name	varchar
navaid_code	varchar
frequency	float64

Airlines

ident	varchar pk
airline_count	int
airline_name	varchar
airline_code	varchar

ETL Process



444 tables

Raw data tables

URL.csv
444 rows

transform

transform

transform

transform

Airfield table
444 rows

Runways table
~450 rows

Signals table
~600 rows

Airlines table
~600 rows

length

width

surface

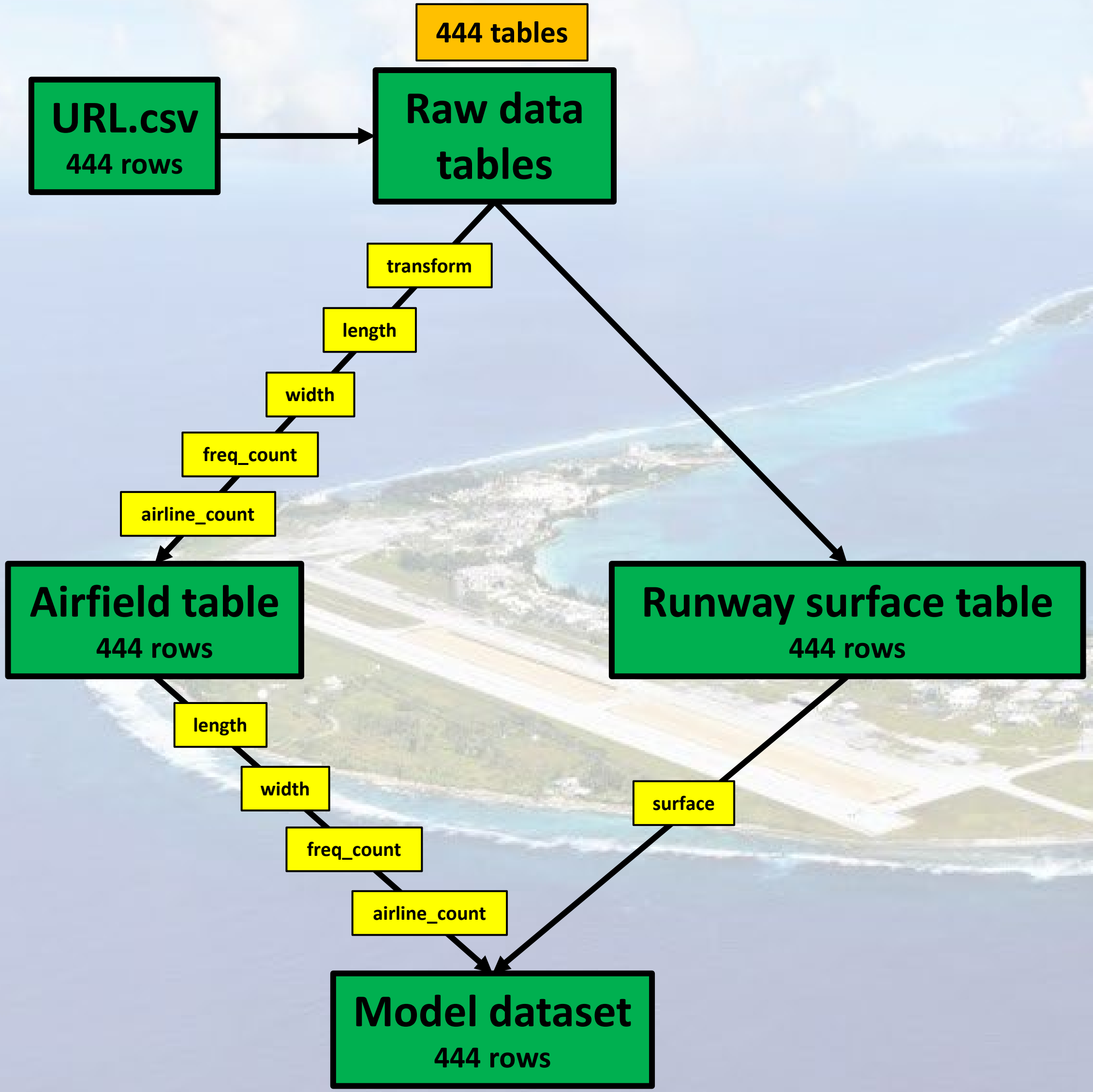
freq_count

airline_count

Model dataset
444 rows

Time limits prevented me from implementing this table structure; I had to go with a streamlined ETL process shown on the next page that dispensed with the runways, signals, and airlines tables.

Data Processing



444 tables

URL.csv
444 rows

Raw data tables

transform

length

width

freq_count

airline_count

Airfield table
444 rows

Runway surface table
444 rows

length

width

freq_count

airline_count

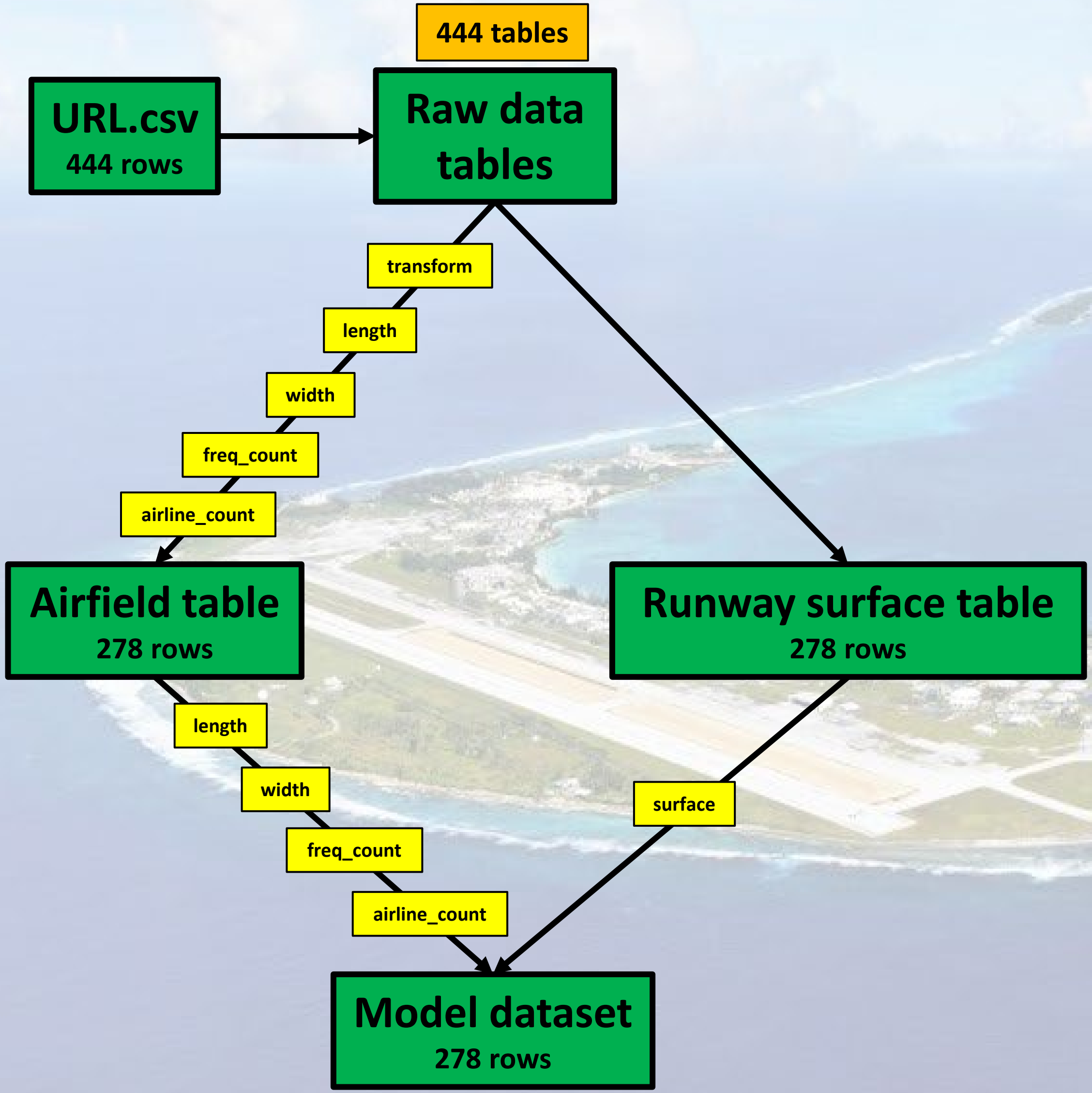
surface

Model dataset
444 rows

Instead, I created four of the model variables (runway width, runway length, signals count, and airline count) during the transform process and stored them in the airfield table.

The runway surface variable was stored in a separate SQL table, then joined with the other model variables to create the input dataset for the model.

Data Challenges



444 tables

URL.csv
444 rows

Raw data tables

transform

length

width

freq_count

airline_count

Airfield table
278 rows

Runway surface table
278 rows

length

width

freq_count

airline_count

surface

Model dataset
278 rows

It required approximately 70 lines of code to transform an HTML table from the airportdatabase.net website into a single row of output data. While I was able to loop this process with the pandas iterrows() function, I was unable to debug all the errors due to missing values and unseen deviations in the HTML source.

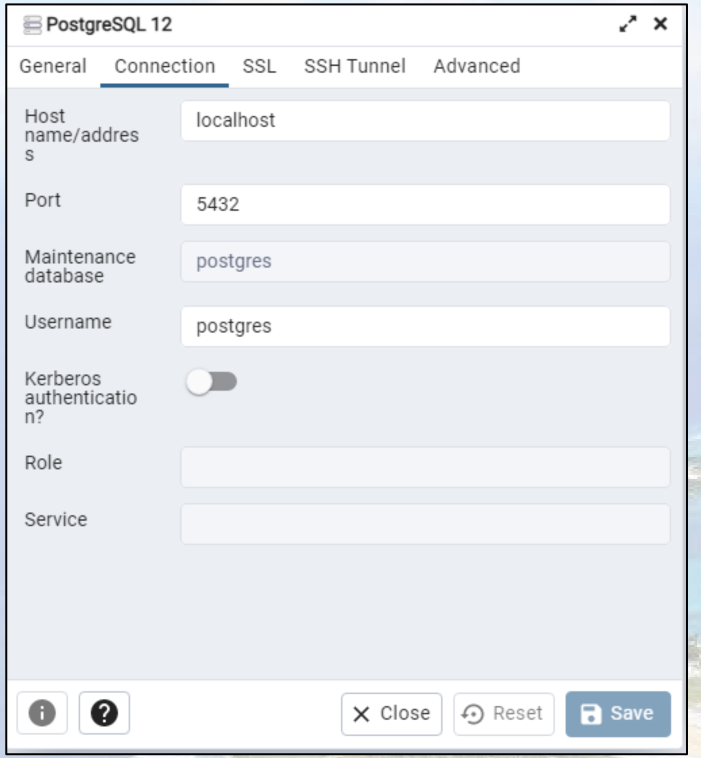
This left me with a semi-automated process; reading one HTML table at a time by cutting/pasting each airfields URL into the data transform code which automatically appended the output data for that airfield into a .csv.

This also cost a lot data; 166 data tables were missing runway length data and there was not sufficient time to collect it manually. These missing observations are overwhelmingly skewed towards small airfields with unpaved runways and little or no ground facilities.

This missing data is clearly visible (by its absence) in the model results described below.

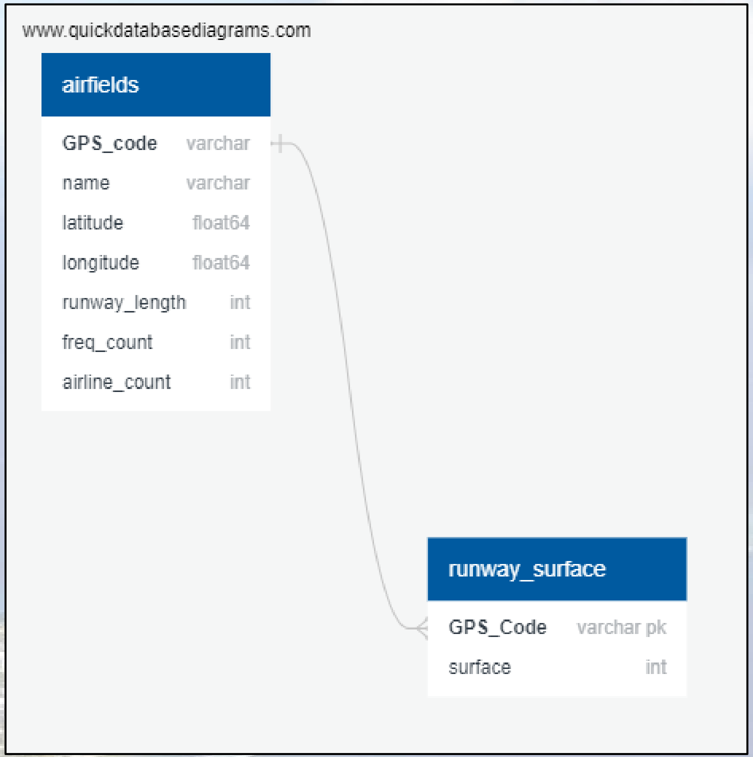
SQL Database

SQL connection information



Host name/address	localhost
Port	5432
Maintenance database	postgres
Username	postgres
Kerberos authentication?	○
Role	
Service	

ERD



airfields	
GPS_code	varchar
name	varchar
latitude	float64
longitude	float64
runway_length	int
freq_count	int
airline_count	int

runway_surface	
GPS_Code	varchar pk
surface	int

The “Airfields” SQL database contains two tables:

“airfields” contains the data scraped from airportdatabase.net

“runway_surface” contains the data collected manually from Wikipedia and Google Earth.

Required database functionality is documented in the *.sql files and *.ipynb model files.

Feature Engineering

The available data from airportdatabase.net breaks airfields into three categories; small, medium, and large. These categories are not sufficient to distinguish between the functional categories of airfields that are important for operational and strategic analysis.

To support this analysis requires segregating airfields into at least six groups based on their operational attributes. To achieve this, an unsupervised machine learning model - K-means clustering - will be used to group the airfields into their functional categories.

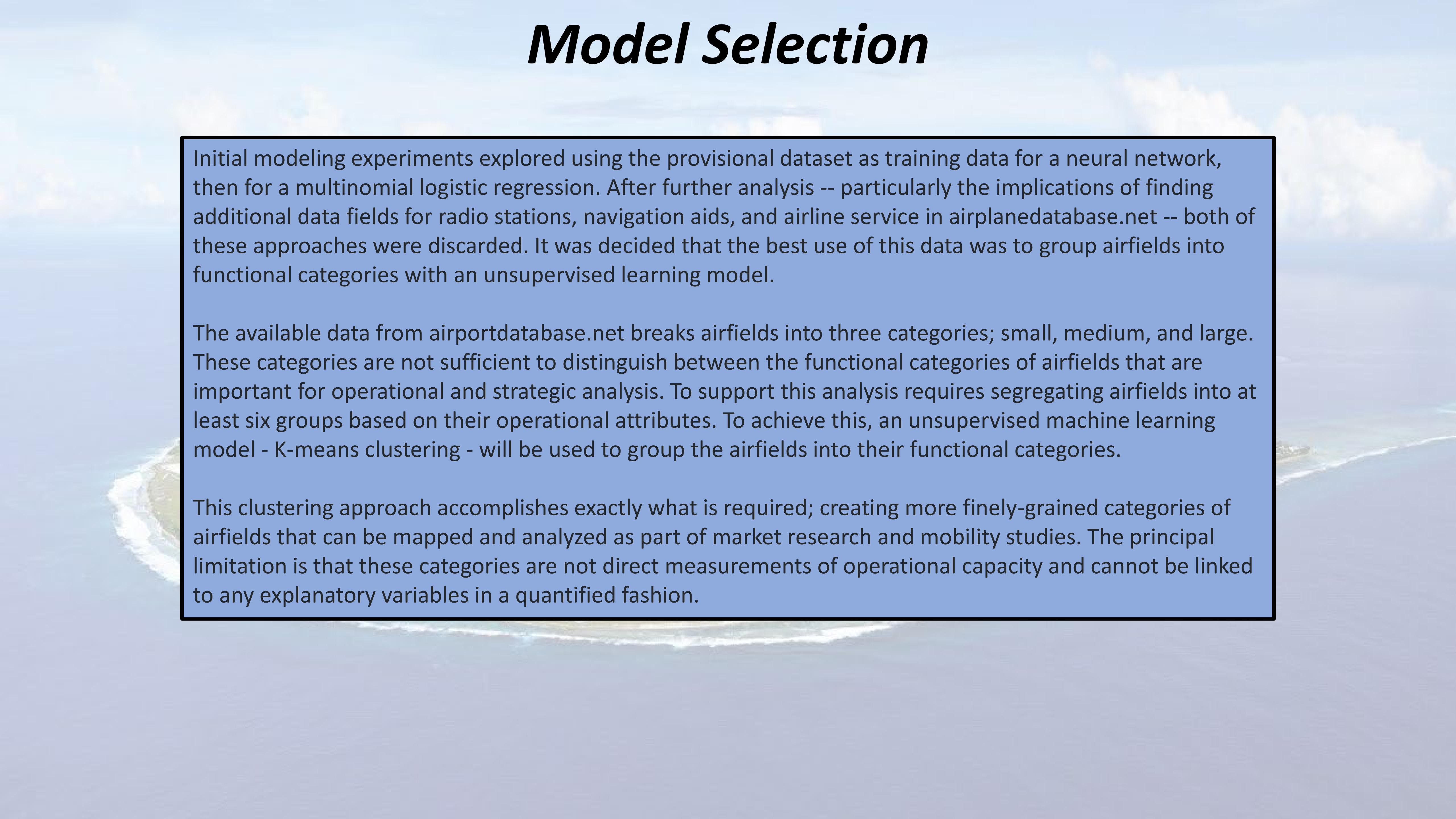
Runway length/width are physical measurements that directly correlate with the operational capacity of an airfield.

Runway surface type is also correlated with operational capacity but is much less precisely measured. I chose to aggregate the numerous surface types into a numeric categorical variable.

The number of radio frequencies and navigation aids used by the airfield is also a direct measure of its capacity to handle air traffic.

Ideally, the final variable(s) would measure the airfield's refueling, servicing, repair, passenger and cargo handling capacities. This data is not freely available for most airfields and is not consistent when it is collected. Therefore, the number of airlines serving each airfield is used as a proxy variable for ground facilities.

Model Selection



Initial modeling experiments explored using the provisional dataset as training data for a neural network, then for a multinomial logistic regression. After further analysis -- particularly the implications of finding additional data fields for radio stations, navigation aids, and airline service in airplanedatabase.net -- both of these approaches were discarded. It was decided that the best use of this data was to group airfields into functional categories with an unsupervised learning model.

The available data from airportdatabase.net breaks airfields into three categories; small, medium, and large. These categories are not sufficient to distinguish between the functional categories of airfields that are important for operational and strategic analysis. To support this analysis requires segregating airfields into at least six groups based on their operational attributes. To achieve this, an unsupervised machine learning model - K-means clustering - will be used to group the airfields into their functional categories.

This clustering approach accomplishes exactly what is required; creating more finely-grained categories of airfields that can be mapped and analyzed as part of market research and mobility studies. The principal limitation is that these categories are not direct measurements of operational capacity and cannot be linked to any explanatory variables in a quantified fashion.

Model Results

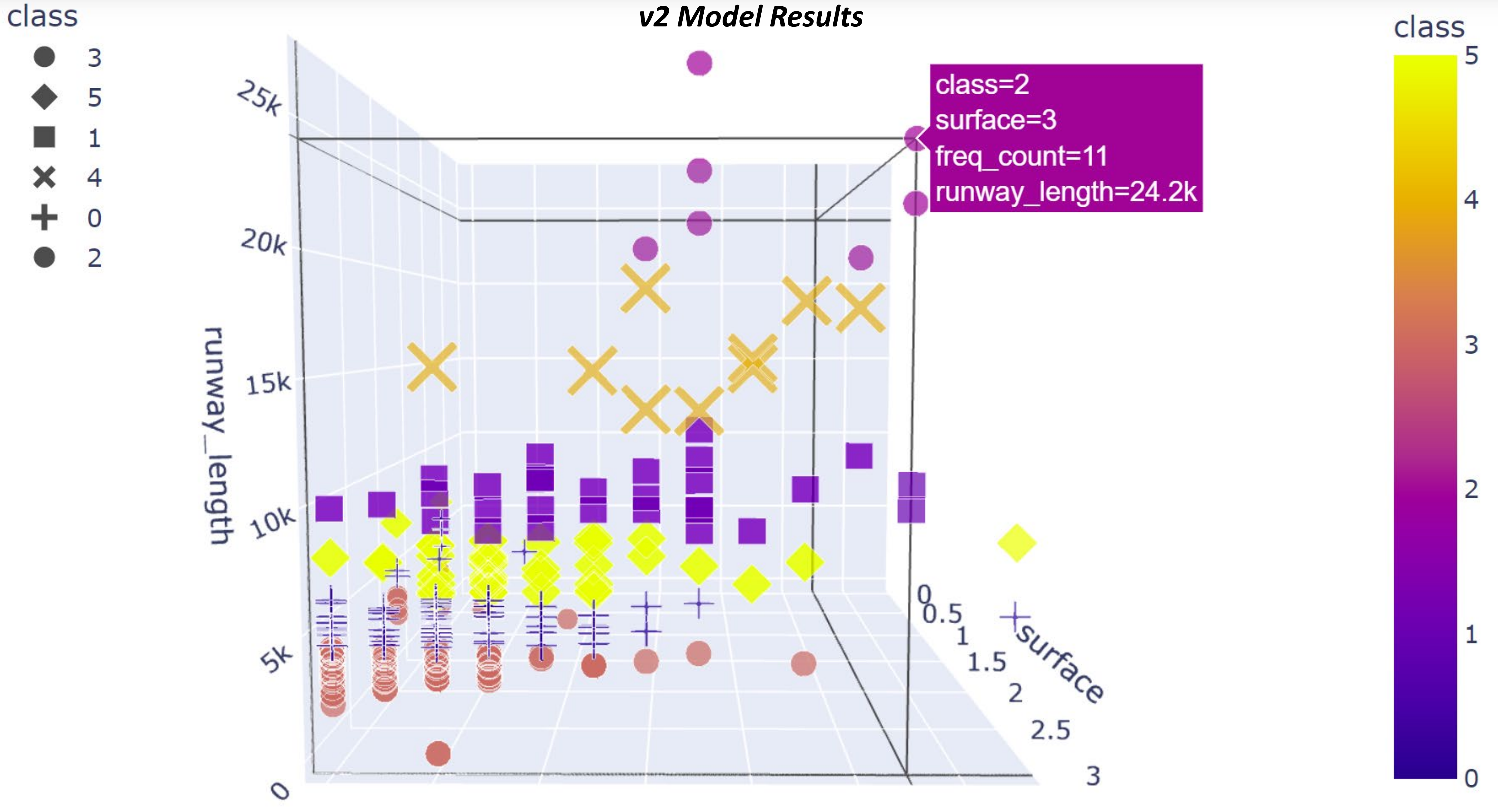
The K-means algorithm does not generate predictions, so it is not generally useful to generate accuracy scores for this model. What it does accomplish is to group observations based on their proximity to centroids in the n-dimensional model space (where n equals the number of model features; n=4 in this analysis).

The value of the model's output comes from visualization of the model output. In this analysis the output is placing airfields into one of six classes, based on they cluster in the 4-dimensional space defined by total runway length, runway surface, # of EM signals, and # of airlines providing scheduled service.

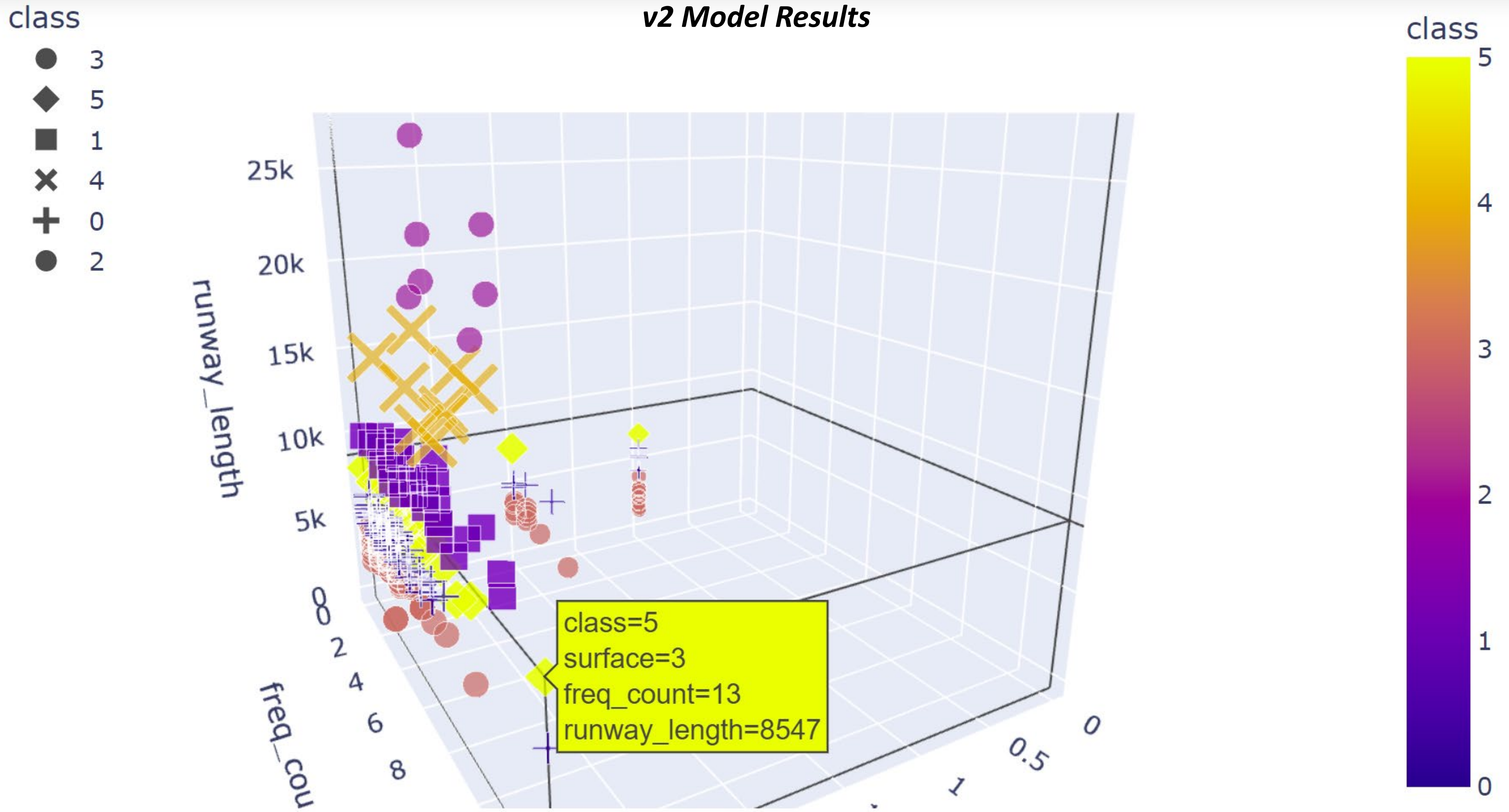
Initial results are displayed on the following three pages. They show that airfields are clustered in six clear bands generally defined by runway size. The missing data for small/unimproved airfields will provide a richer picture, but it does appear that runway length is the dominant feature in the dataset.

The following three pages show results from the preliminary model from segment one, which used a smaller dataset for the Northern Pacific Ocean, but with more observations for small/unimproved airfields. This type of 3D visualization is the preferred method for presenting results to users, however it is well beyond the capabilities of Tableau.

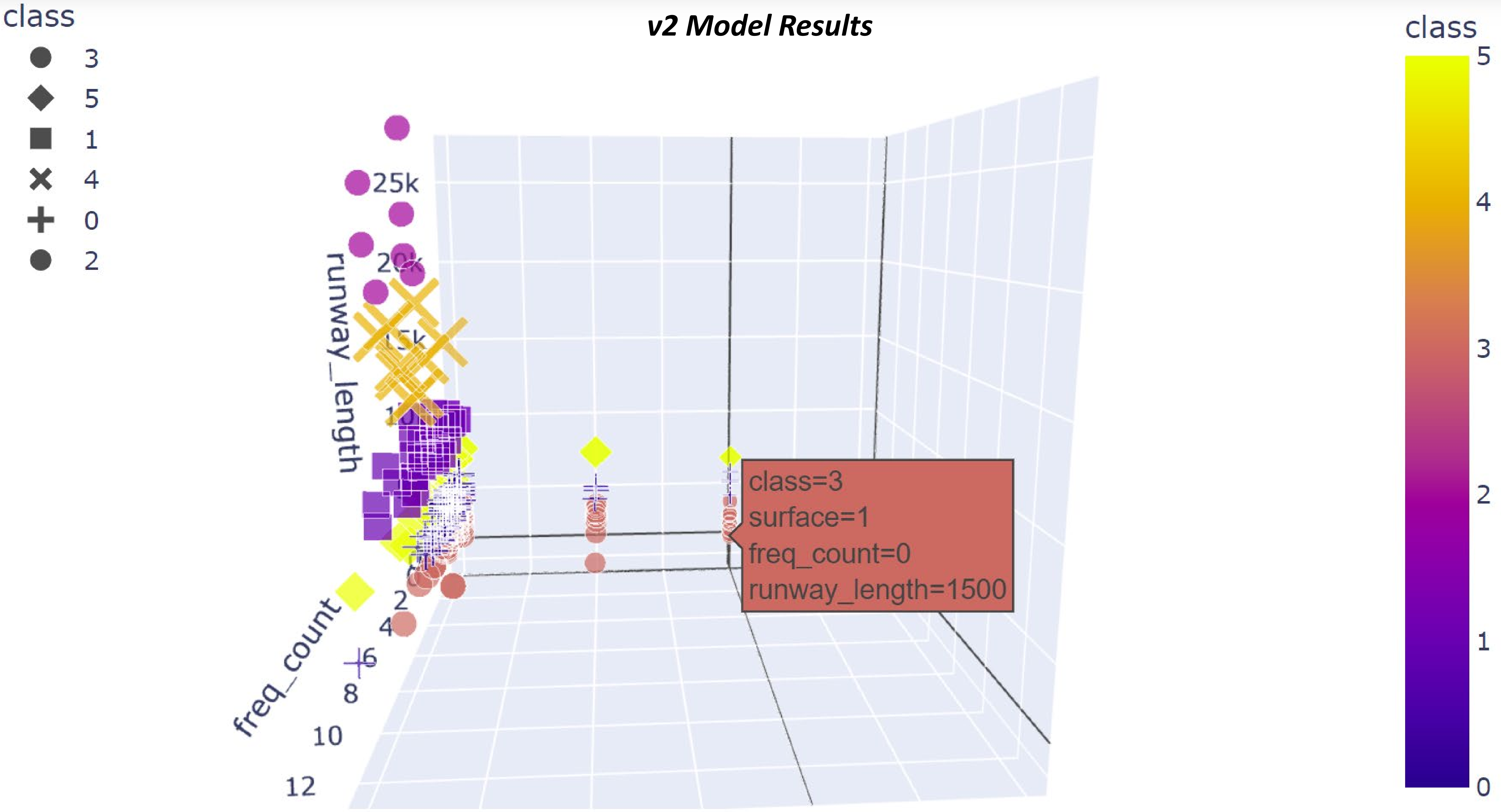
v2 Model Results



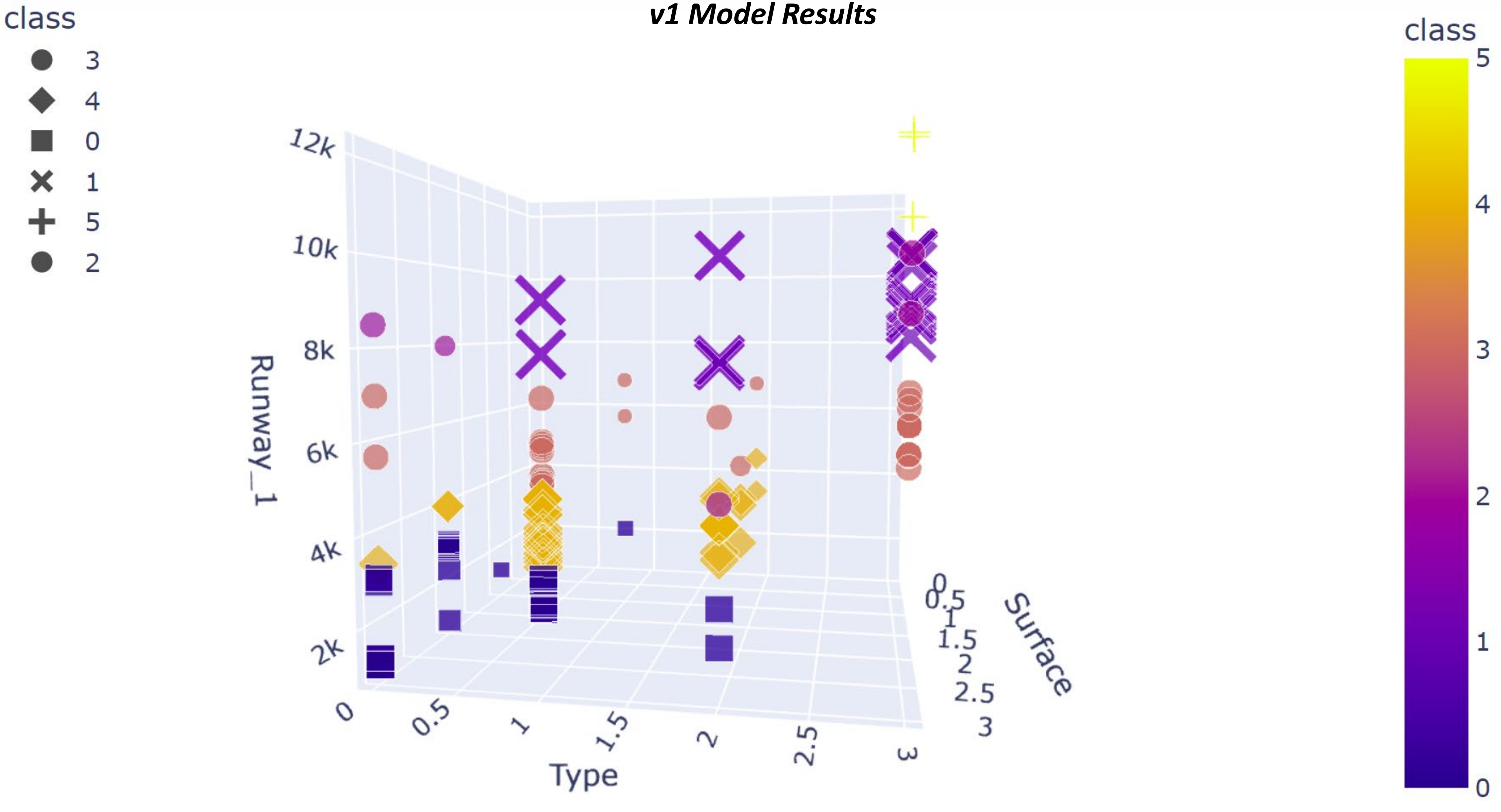
v2 Model Results



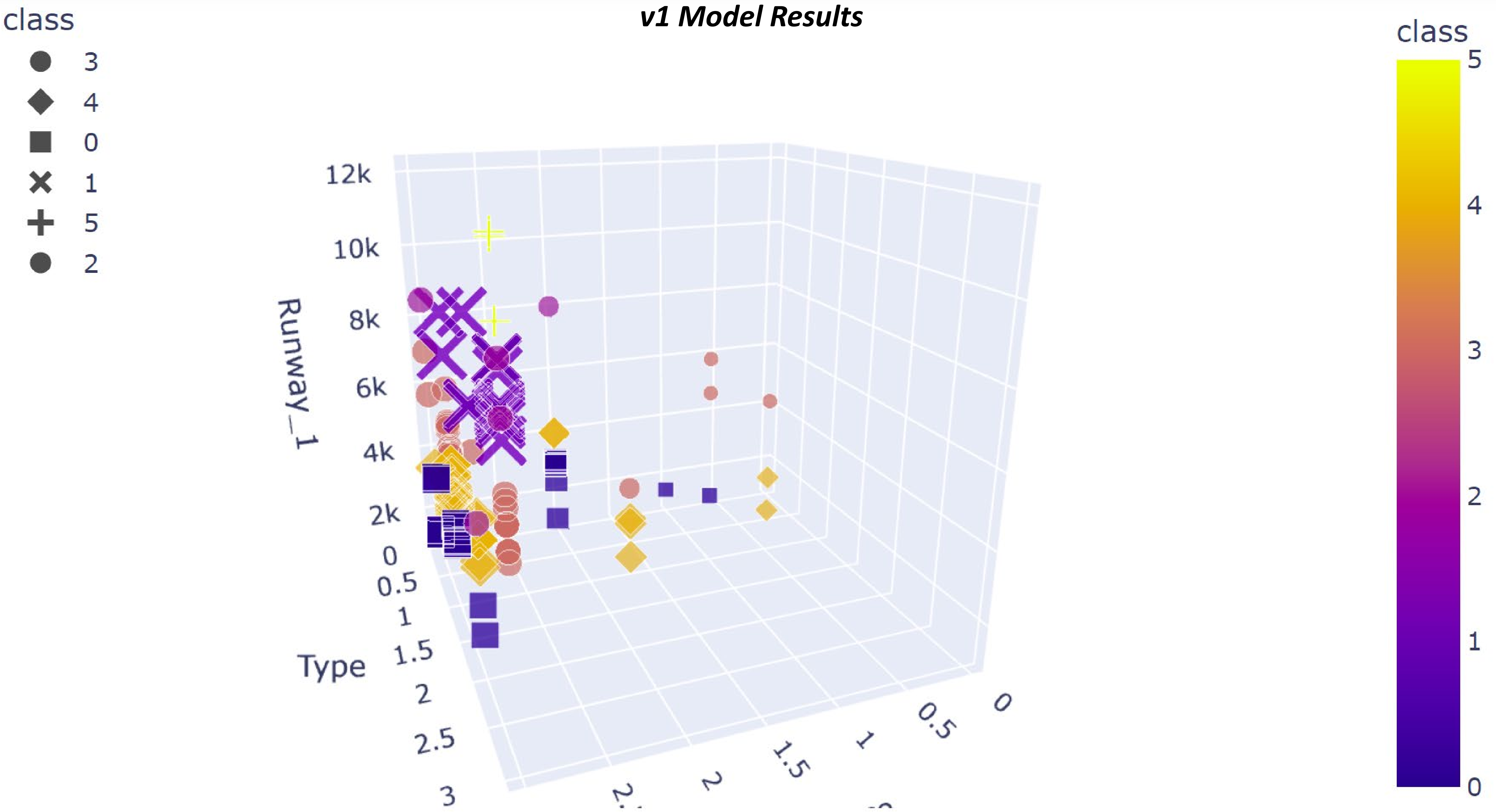
v2 Model Results



v1 Model Results



v1 Model Results



v1 Model Results

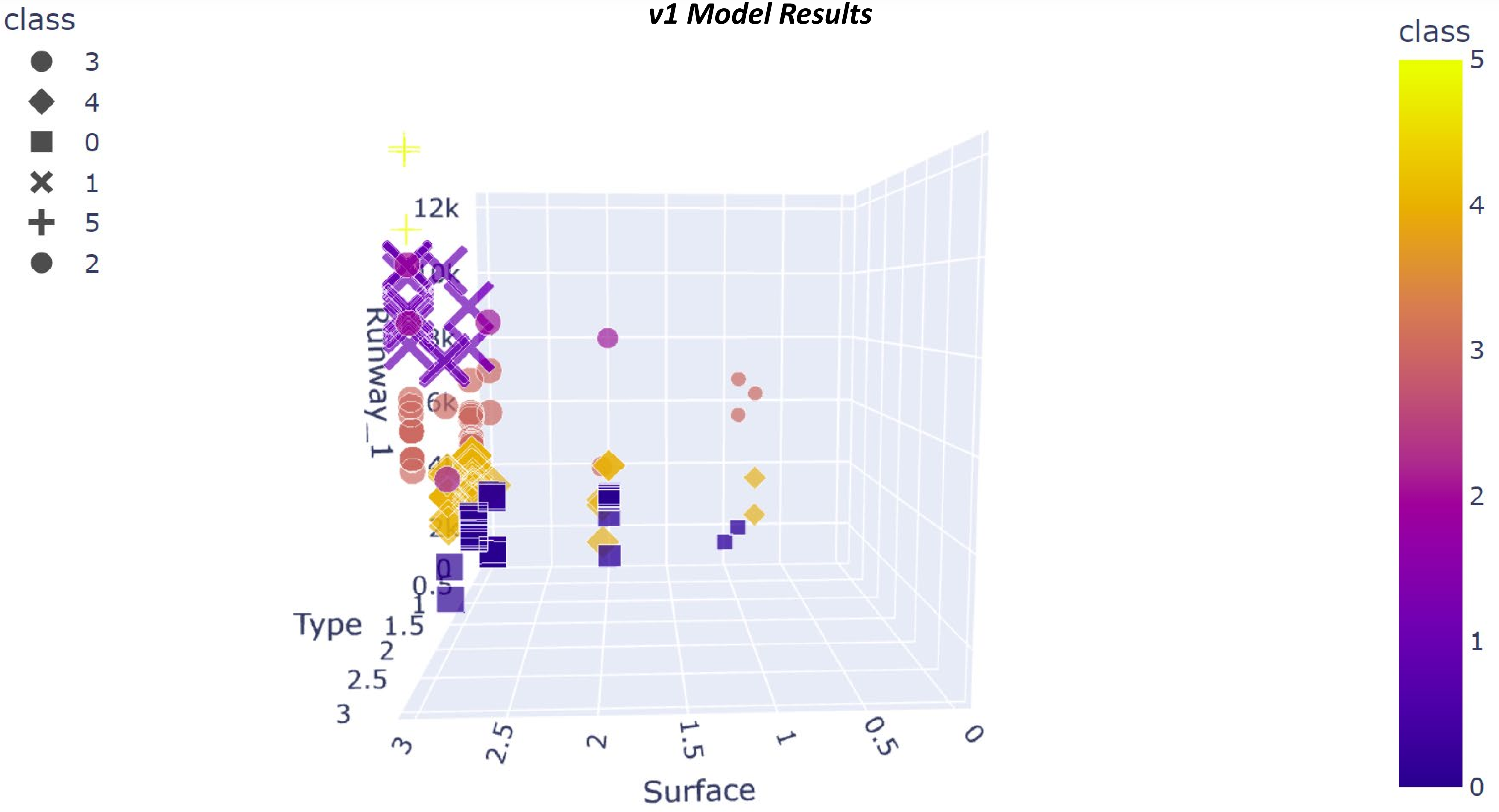
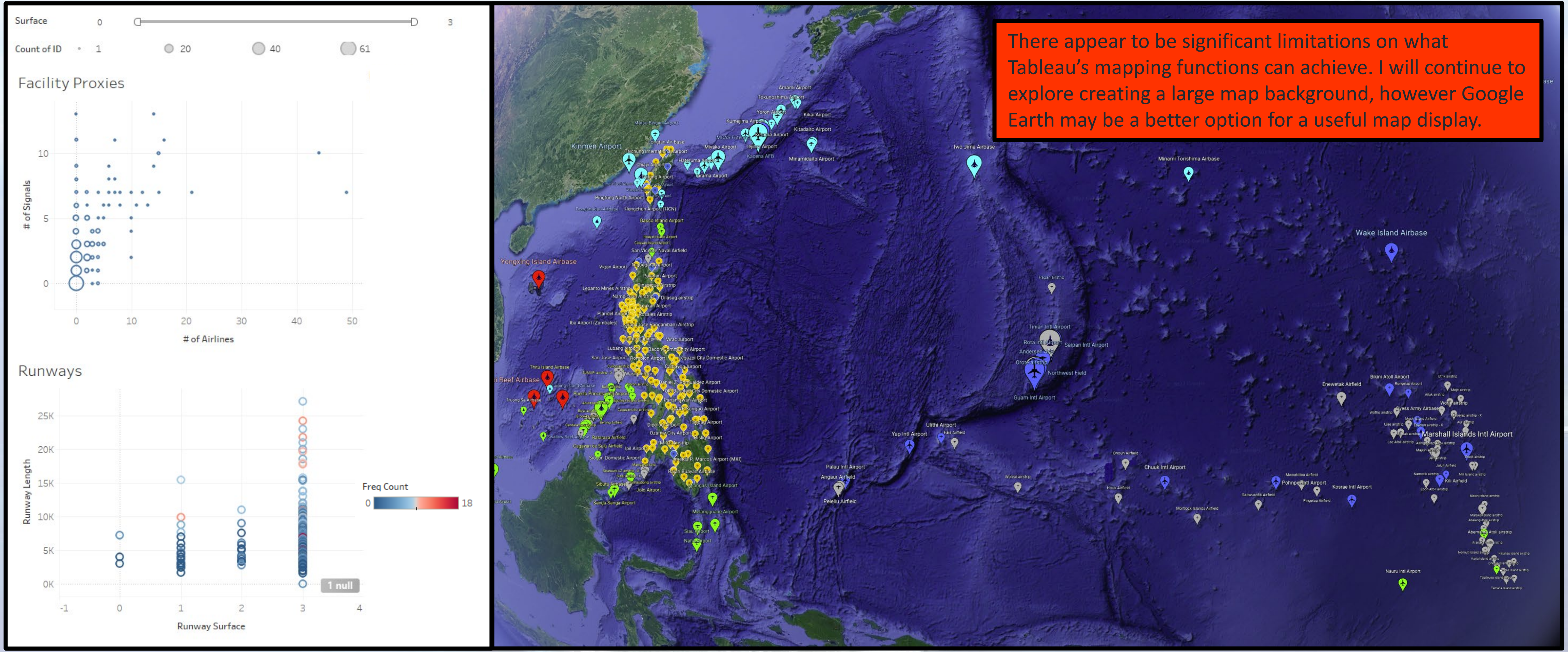


Tableau Dashboard

Slider bar filters airfields on surface type



There appear to be significant limitations on what Tableau's mapping functions can achieve. I will continue to explore creating a large map background, however Google Earth may be a better option for a useful map display.

The interactive elements for this dashboard will include:

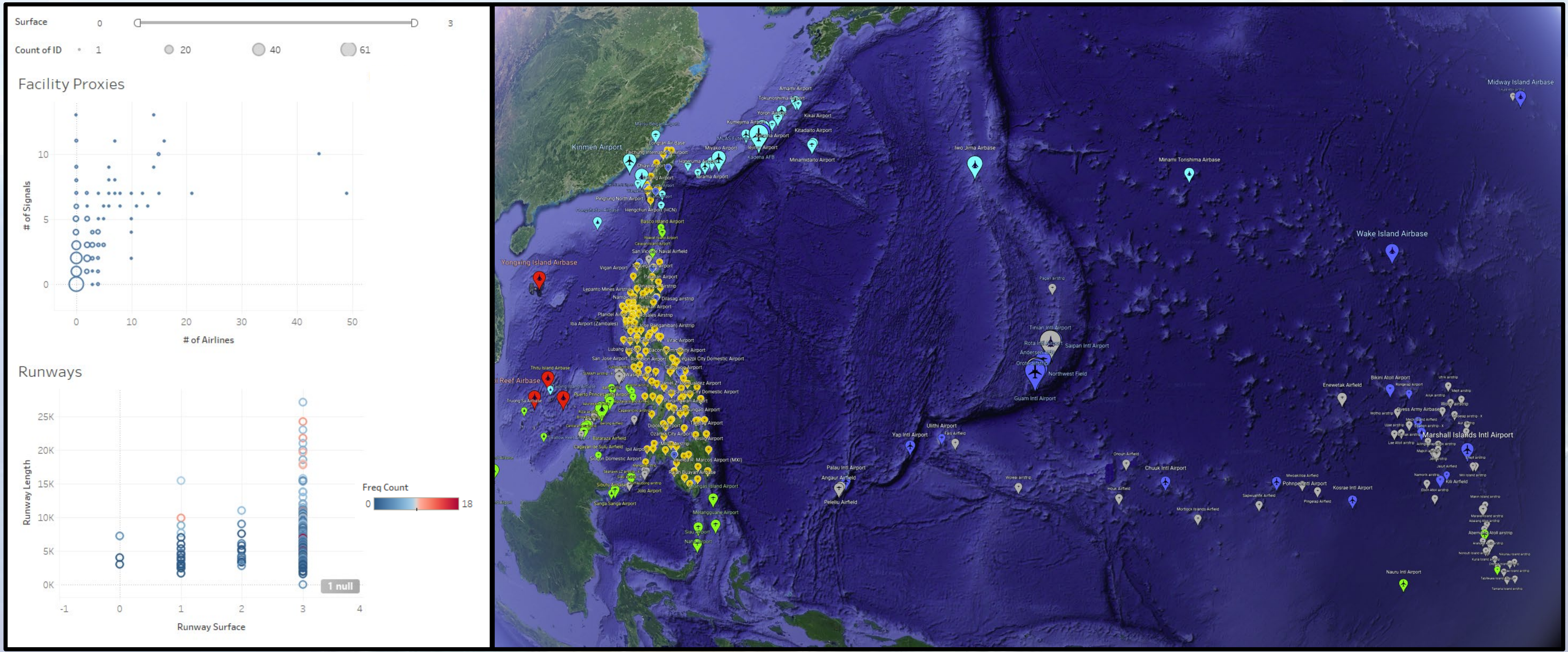
1) The map will allow filtering the displayed airfields based on any of their model attributes (runway length, runway surface, # of signals, # of airlines, airfield class) as well as countries.
2) The analytic graphs will allow filtering by countries and airfield class.

The dashboard for this project will consist of two main sections:

1) A geographic view that displays airfields by location. This view will display airfield icons with their sizes determined by the output of the machine learning model.
2) An analytic view comprised of 2-4 graphs that show key empirical relationships. For example, the "Runways" graph shows runway surfaces vs runway sizes, while the "Facility Proxies" graph shows the # of signal stations vs # of airlines at each airfield (the size of the bubbles represents the number of airlines at each ordered pair).

Visualization

Slider bar filters airfields on surface type



The interactive elements for this dashboard will include:

1) The map will allow filtering the displayed airfields based on any of their model attributes (runway length, runway surface, # of signals, # of airlines, airfield class) as well as countries.
2) The analytic graphs will allow filtering by countries and airfield class.

Their appear to be significant limitations on what Tableau's mapping functions can achieve. I will continue to explore creating large map backgrounds, however Google Earth or Mapbox may be a better options for an interactive map display.