INTRODUCTION + OBJECTIVE
Create an algorithm that can label when a Rolls-Royce AE3007 engine has reached its compressor wash interval. Show this on an interactive application where clients/operators can see when the aircraft engine requires a compressor wash.

What is a compressor wash interval?
- Aircraft engines occasionally operate in a corrosive environment (salinity, dust, industrial particulates, etc.)
- Material properties can degrade
- Cause engine failure
- Each engine has a “health bar” before it needs to go in for maintenance
- Remove the internal and external contamination by washing/cleaning the engine

What did we use?
- R in RStudio for Data Analysis, Cleaning, and Wrangling
- R Shiny Package to build an interactive app
- Show engine flight path and time spent in corrosive environment

What we started with:
- Public flight data from Department of Transportation Statistics
- Rolls-Royce engine data
- Map of the corrosive environments in the US

DATA CLEANING
- Input clean public flights data
- Merge the Rolls-Royce dataset and the public commercial flight dataset so we can map each engine to a plane. This adds on locational data to each engine, allowing us to track where each engine travels and see whether it operates in a corrosive region.

DATA MERGING
- Merge the Rolls-Royce dataset and the public commercial flight dataset so we can map each engine to a plane. This adds on locational data to each engine, allowing us to track where each engine travels and see whether it operates in a corrosive region.

STAGE 1: DATA WRANGLING
Merge the Rolls-Royce dataset and the public commercial flight dataset so we can map each engine to a plane. This adds on locational data to each engine, allowing us to track where each engine travels and see whether it operates in a corrosive region.

Two step process:
1. Clean the data so it is in a readable useful format
2. Match datapoints from each dataset based on three criteria

In the end, we have 500,000 points of Rolls-Royce engines mapped to public commercial flights over 5 years (refer to Figure 4)

REFERENCES
Special Thanks to Brian Woods and Rolls-Royce
U.S. Department of Transportation
Research and Innovative Technology Administration
Bureau of Transportation Statistics
PROJECT BACKGROUND

**Goal:** Map interactively in Shiny in order to see if flight’s history and travel locations warrant compressor wash

Compare Rolls Royce Flight information against public flight data provided by the Federal Aviation Administration.

WHY SHINY?

Powerful web app development tool
For this project we are using Shiny to:
- Map flight paths,
- Create a visualization of corrosive regions
- Display which engines most warrant compressor cleaning

PROBLEMS AND CHALLENGES

- Credit/debit system to gauge how much corrosion a flight would be susceptible too was not feasible
- Encountered many confounding variables (i.e. where a plane stays overnight)
- Overlaying maps led to some guesswork in determining what latitude and longitude to use based on the scale of initial map

ACKNOWLEDGEMENTS + REFERENCES

- We would like to recognize Brian Woods and the Rolls Royce
- In addition we want to thank The Data Mine, Purdue University, Dr. Mark D. Ward and Margaret Betz

APP METHODOLOGY

- Transposed Rolls Royce map onto a map of respective U.S. counties (see Figure 1)
- Recorded where 3000 counties lay based on zone (mild, moderate, severe) using Excel and R (see Figure 2)
- Extract the ZIP code database from Zip-Codes.com (~40,000 ZIP codes with both city and county information)
- Utilized public ZIP code database from the US Census Bureau and latitude/longitude information to ensure accuracy (worked with 43,000 codes with city and latitude/longitude information)
- Using R, the data were cleaned and merged, using county information and latitude/longitude to map the counties based on corrosion zones across the continental United States
- Formatted mapping into an app through Shiny (see Figure 3)
INPUT FIELDS

The main goal of this application was to highlight a specific engine’s flight patterns and its impact with corrosive regions. This is done through the following:

1. INPUT FIELDS (FIGURE 1)
2. DATA DISPLAYS (FIGURE 3)
3. AN INTERACTIVE MAP (FIGURE 2)

The app is configurable using the following inputs:

- **Date Range:** The user may select any range of dates for which there is data available.
- **Engine Numbers:** The list of all matched engines can be searched, and the user may select as many as they like to display.
- **Mild Region:** Checking the ‘Show Mild Region’ check box will display the mild region as a polygon on the map.

DATA DISPLAY

When engines and dates have been selected, the app will display some simple statistics including:

- The total number of days considered
- The total flights per engine that go through the corrosive region (marked red on the map)
- The total number of hours spent in the corrosive region.
- Both statistics are dependent on the time frame selected.

This information can give the engineers a good idea of which engines need to be washed.

INTERACTIVE MAP

The map provides visualization of the selected data including:

- Lines connecting origin to destination for all flights
- Coloring based on whether the flight passed out of the mild region
- The ability to pan, zoom in and out, and view arbitrary landmarks such as cities and state borders
- Clicking on a flight will display a pop-up revealing the date, engine number, tail number, origin and destination airport codes of that flight.

Date Range

2015-03-15 to 2015-06-02

Engines

Search

- 8688
- 293
- 3153
- 3725
- 8671
- Show Mild Region

DATA DISPLAY

Total number of days: 79

<table>
<thead>
<tr>
<th>Engine Number</th>
<th>Total Flights</th>
<th>Time flown in Corrosive Region (hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>293</td>
<td>244.00</td>
<td>363.55</td>
</tr>
<tr>
<td>8688</td>
<td>280.00</td>
<td>421.75</td>
</tr>
</tbody>
</table>

Figure 1: Input Fields

Figure 2: Interactive Map

Figure 3: Data Display