Machine Learning 498_PS 1 Write-Up

Nick Rucker

April 23, 2019

1 Preface

For this problem set, I decided to investigate the practicality of using machine learning to predict airline routes. The problem of efficiently scheduling airlines routes is a daunting task that has yet to be perfected. My goal is to use current data to be able to predict which routes airlines will take from any two points in the US-based off their current scheduling methodology.

2 About the Data

In order to predict airline routes, a few pieces of data are needed. I knew I was going to need data from a traditional airfare booking site, but I was also going to need the coordinates for all the locations on the trip. To collect the data I decided to scrape Expedia.com and gps-coordinates.org. The collection of the data was a little tricky since the results on Expedia were processed with JavaScript and the coordinates required information to be entered in text-boxes. To overcome this hurdle I used a developer package called 'Selenium' which makes it possible to automate the collection data efficiently. On Expedia, I scraped the Airline, ticket price, departure, destination, layover city (if applicable), and the total travel time. The purpose of this data is to discover which airlines cluster at which airports. The GPS coordinates are used in order to apply the clustering to any airport within the US. Since strings are essentially meaningless in the machine learning process

I randomly assigned a value to each airline. I was able to use a random number since the purpose of the assignment is to match the number to a particular string. The three character airport code was used as a 'key' and the value was the GPS coordinate location of that airport.

**Data was collected departing from CHS to the 20 most busy airports within the USA*

3 Machine Learning Process

As previously stated, the goal of the machine is to predict a route that is similar to that of a particular airline. This means that given a departure airport, destination airport, and airline, the result would be the layover airport. Since the target will consist of two data points, a machine that can handle a multi-target output is required. Since clustering seemed like an option to predict routes, I ended up choosing a KNN Regression over a KNN Classifier to handle the multi-target output. Intuitively, the results from a test of predicted values make sense. For the predicted routes the layover is deterministic based off the airports the airline has a hub at and position along the route. A route from ATL to BOS on Delta does not result in a layover in SEA. Although the routes make sense, it is hard to say how closely the machine mimics the diversity of real airline routes. While the route makes sense and is possible airline route, airlines often offer multiple

routes to get to a particular destination. The machine does not account for the variability of routes between two places and computes the same result each time. This brings up a little flaw in the internal validation process. Just because the machine did not predict each particular route correctly, does not mean the machine is not predicting accurate, realistic routes. Since there are a myriad of factors that go into scheduling airline routes the machine's accuracy score is not as relevant as if when personally judged on if a route makes sense.

4 Final Statement

For this particular machine the routes predicted, with some tolerance* on interpretation, make sense all things considered and accurately predicts plausible routes.

**Tolerance meaning a flight from CLT to LGA with a layover in CLT on American Airlines should be interpreted as a direct flight*

e.g. A predicted route from FLL to SEA on Delta results in a stop in ATL. Delta has a large hub in Atlanta which would be able to offer the direct flight to Seattle, so route seems more than plausible. The layover in ATL is the best answer given it is the largest 'hub' for Delta. Although there is still a possibility the flight is routed through ORD, that route is not predicted.