Uber Nashville Project

Adam Sullivan

**Project Description**

This project was a pre-interview project designed by Uber for their machine learning specialist position in Pittsburgh. The dataset is a ~22,000 entries from trips originating in/around Nashville, TN for a few weeks in February, 2015. The objective was open ended: design a machine learning application from the given dataset.

The application that I designed was a way to predict the time of day we would expect a ride given a day of the week, user, expected starting position, and number of unique clusters that the rider has been to in the training data set. This application would allow Uber to assist in proper scheduling of drivers for different periods of time.

**Machine Learning**

There are two main elements of machine learning that I exercised in this API. The first is that there is an unsupervised clustering of all locations using k-means. The default number of clusters is 32, chosen by intuition. This clustering of the locations allows for elimination of GPS noise, people walking, etc. The features set also consisted of a factor day of the week, a factor start cluster, a factor end cluster, and a factor “time of day”, which separates the hours into 5 categories. The objective is to take the user ID, start cluster, (presuming we have a decent guess of where the individual might be) day of the week, and predict the time of day that the user would make a call.

The actual prediction uses a naïve Bayes predictor. I chose this predictor for a number of reasons, mostly because of the multinomial prediction I was attempting to make. This seemed like the most likely way to take into account the individual user in my model in the few hours allotted for the problem. I set out that I wanted the model to be user specific instead of a global prediction model. For example, if a new user and old user both hailed an Uber at 1:30AM on a Saturday night, I wanted the predicted time to have the flexibility to vary by user.

The API in run from a main.R script. One can set the training, testing, and new testing datasets, as well as option for changing output directories. The entire project is bundled as an R package, similar to a python module. Each of the functions has roxygen documentation, which is compiled into a native R documentation project format. This can be viewed in Rstudio, and it can also be exported to a PDF.

**Other ideas**

When I first started, my initial inclination was to create a hierarchical multinomial logistic model to predict the end cluster, given a starting cluster (there is a lot of evidence of this throughout my code). There were two flavors of models that I created: a bayesian MCMC hierarchial logit model and a frequentist multinomial logit model. This would’ve handled my main objective: predict on both a global level and the individual level. I wanted to be able to make good predictions for the guy who takes Uber every morning to work that is more individualist, while making a more global model for picking up somebody who is using Uber for the first time to get home from Second Ave/Broadway. Unfortunately, for multinomial set of classes (the start clusters, which we had 32), I wasn’t comfortable with my design set up. I could achieve a model, but I really wanted to be able to capture the individual/global effects, and understanding that wasn’t within the scope of this project. Yet!

**Future Extensions**

Obviously, I’d love to have more data and more time. The dataset wasn’t too large to crush any of my computing resources. I would love to continue to exercise the prediction of the end clusters based on the other available data. My project goal would have a function similar to:

endCluster[i,j] ~ startCluster[i,j] + avgDistance[i] + time[I,j] + dayOfWeek[I,j] + likelyHomeLocation[i] + likelyWorkLocation[i] + downtownOrigination[I,j] + barTrip[I,j] + businessTrip[I,j] + businessCustomer[i] + barCustomer[i]

where I is the individual ID and j is the trip ID. This model would allow for individual and global effects. This would turn it into a Choice Model, similar to marketing analysis for product research. If a customer is subjected to a few effects (day of week, time of day, starting location, type of trip, etc.) what would be their predicted final destination? A Bayesian hierarchical multinomial logistic model would allow us to make these predictions. I would develop this using the R packages bayesM, or in stan, using an MCMC sampling method.

Hopefully the API is easy to understand. The main.R is the example for the how the API is handled.