Exercise: Predicting Rideshare Demand

Objectives

In this exercise, you will use an LSTM model to make to make predictions of rideshare demand.

After completing this exercise, you will:

1. understand modeling urban spatio-temporal prediction problems
2. use an LSTM architecture to make predictions using mobility data
3. understand the influence of various parameter and design choices

Code:
The code we will use is available on github:
https://github.com/annieyan/RideAustin_demand_prediction

We will use the jupyter notebook lstm_rideaustin.ipynb for this exercise.

Setting up:

Make sure you have the required libraries installed. I recommend using conda to install them.

```
$ conda install tensorflow numpy pandas matplotlib seaborn

$ git clone git@github.com:billhowe/urbanalyticssummerschool.git
$ cd urbanalyticssummerschool/part2/exercise2
$ git clone https://github.com/annieyan/RideAustin_demand_prediction.git
$ cd RideAustin_demand_prediction/notebooks
$ jupyter notebook lstm_rideaustin.ipynb
```

Datasets:
We will use the RideAustin Rideshare dataset. RideAustin is a non-profit rideshare service deployed in Austin, Texas, USA to fill the gap left when uber was not permitted to operate in the city.

The data has been formatted as a grid of locations covering the city, where each grid cell has a hourly timeseries of rides taken from August 1, 2016 to April 13, 2017.

Question 1

Run the cells in the jupyter notebook and seek help from the instructor or other students to resolve any errors. Read the comments.

Write a few lines of code to find the top 5 regions with the highest total rides across the whole timeseries. We want to find the busiest regions to work with to train our model.

Use the first plot in the notebook (the timeseries plot) to double check that your results seem correct, by changing the region being plotted in the cell above.

Question 2

Find the two busiest hours of the day across the whole city.

You can use `pd.DatetimeIndex(hourly_grid_timeseries.index).hour` to extract the hours. Then compute the average ride count by hour (i.e., group by hour), and report the top two.

Question 3

Train the model for each of the 3 regions you found in Question 1. Report the MAE for each of the three regions.

Question 4

For your best performing model associated with region X, evaluate the model trained on X on the test data from the other two regions. You can construct a generateData object for each region, but use the test method of the SeriesPredictor class for your *trained* region X.

Report the MAE of the model associated with region X on all three regions.

Comment on the performance. Is it feasible to train a model on one region and use it on another, as a simple instance of transfer learning?

Question 5

Evaluate your best performing model associated with region X on the 8 regions neighboring X. For example, if X is 8_17, evaluate on 7_16, 7_17, 7_19, 8_16, 8_19, 9_16, 9_17, 9_18. Report the MAE for these neighboring regions.

Does the model perform well on nearby regions, relative to your answers in question 4? Why or why not?

What to turn in:

Report your answers to the questions on this google form:
https://forms.gle/RJUPbA1w6itaqA2R8