Seems to be focusing on prediction of the next location of a vehicle (transport context). Network conditions before they occur (?really?)

Trip purpose: difficult to acquire accurately? They are using rule based methods at the moment.

Tiers: should them all have some kind of impacts on the destination?

First 5/15/30 days, drifting effects?

LTDS and NTS work at bigger scale - define the rules

Mobile data: disaggregate scale and providing updates.

The Netlogo model

Target: to explore the relationship between prediction accuracy with data availability and uncertainty in people’s travel behaviour.

Currently, the type of factors considered are:

* Land use considered: residential, shopping, work and school
* Day of week – weekdays, weekends
* Type of employment – full time employee; part-time; students; unemployed (basically decides their schedule)

What can be set:

1. Speed of travel (although this is universal, can’t define separate speed for different journeys yet, but should not be too difficult to do)
2. Type of data available: mobile phone data;
3. Setting how complicated the road network to be
4. The number of agents to be simulated
5. Data availability: Mobile phone: number of phone calls a day (assuming uniform distribution)
6. Variability

How the model works:

1. Generate land use based on the defined coordinates
2. Generate road networks and assign random link length based on the parameters set on the network
3. Generate the population and assign parameters, including type of employment, day schedule (once it’s set it’s fixed at the moment)
4. Starts the simulation, each agents move based on their schedules. Detailed routes are selected based on the shortest path problem.
5. Data recorder writes data into a text file for machine learning

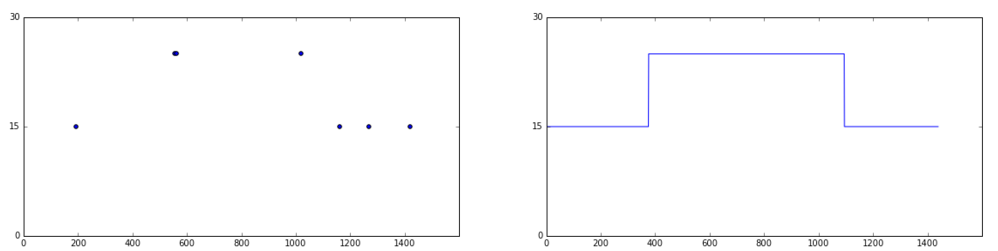
What can be predicted:

1. When will people travel, to which land use area

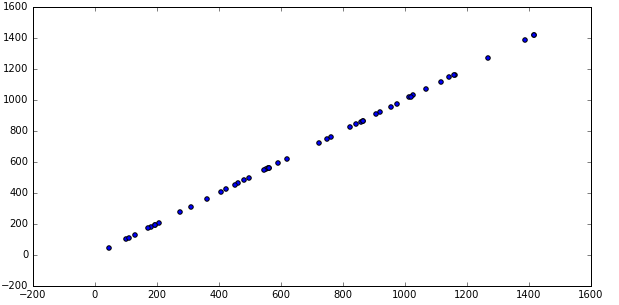
Things not so smooth:

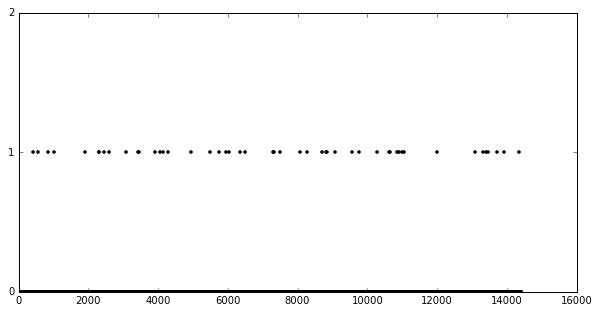
1. The “enroute” status not available in the schedule, but available in the simulation
2. At the moment only forecast “move”, “stationary”
3. Connection with python and netlogo is a bit slow – have to do frequent file writing and reading through hard disk which slows down the performance
4. Use pcolor to represent zone (maximum at the moment 140)

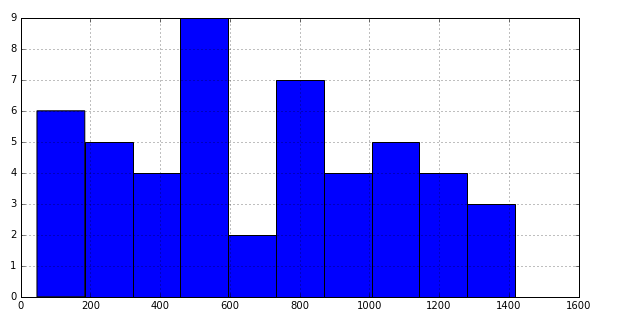
Testing of KNN imputation for the missing data:

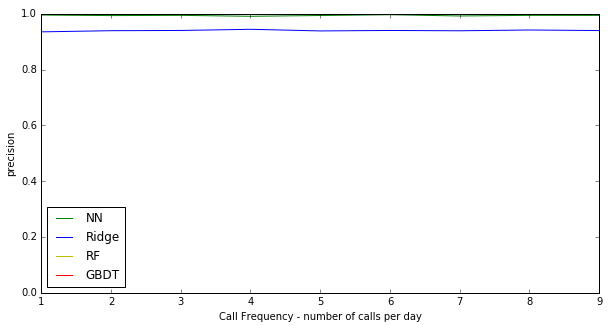


Checking the distribution of mobile phone calls:









Evaluation metrics (what to choose, depend on application)

* Accuracy
* Precision
* Etc.

precision is the ability of the classifier not to label as positive a sample that is negative, and recall is the ability of the classifier to find all the positive samples

Flow chart for the simulation environment:

Setting key simulation parameters through Python scripts

Generate schedule for each of the agents (people)

Through a Java bridge, run Netlogo from Python and read the schedules

Netlogo writes simulation results to hard disk

Python read simulation results and carry out prediction

How Netlogo simulation works:

Netlogo initiation

Generating the required number of agents and assign attributes, inc home,work, school shopping locations

Setting up road network based on input para with attributes

Setting up predefined zones

Setting up data collection structure (e.g. mobile towers)

At start of the day read schedules for each type of the agents

Using shortest path algorithm, according to their schedules, agents move along the road network for interval i

Does a data trigger event occurs?

Yes

Data collection event triggered. Data collected based on its own systems

No

Write to files – true trajectories / zones history recorded for the agents

Write to files – data collector’s records about partial trajectories/zones history for the agents

End of the day?

No

End of the simulation?

Yes

No

Simulation ends

Yes

Python initiation (parameter settings)

Generating schedules for (the next) x days of y agents

Run Netlogo for x days

Simulation data for the True trajectories/ zones visited

Simulation data from the data collector (data simulator)

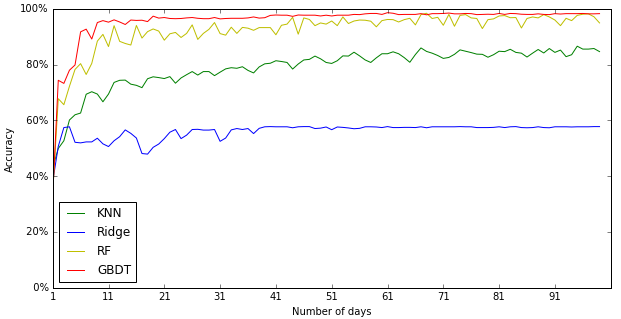
Python ML training and cross validating based on x-1 days

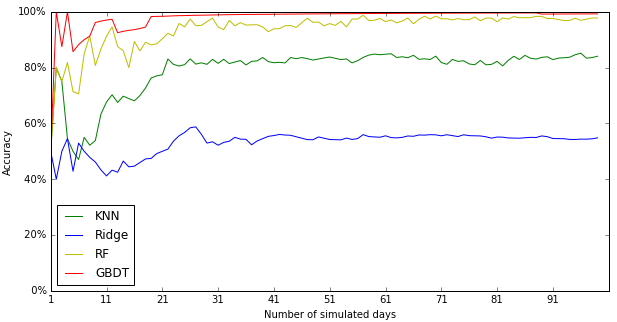
Python prediction for the xth day

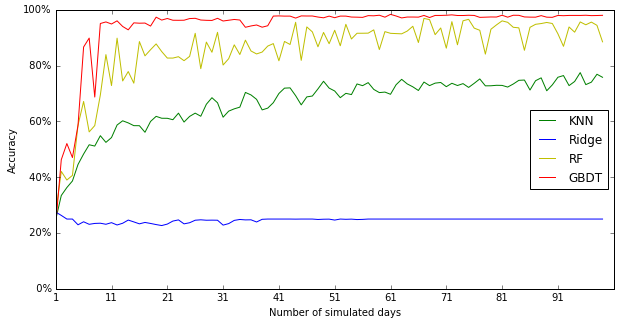
Comparing with true data for the x day. Benchmarking the performance

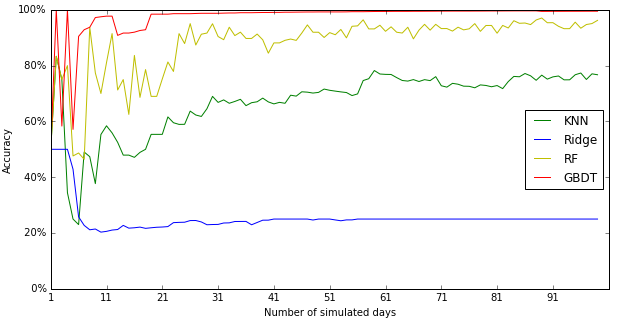
Plotting, visualisation

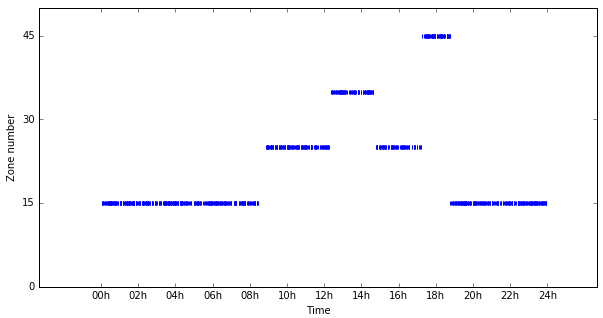
Assumptions with regards to data availability, variability and quality

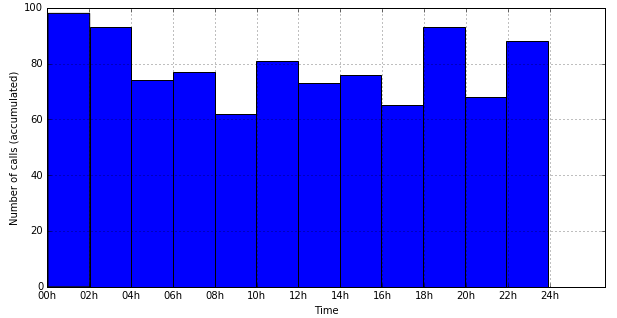
Prediction accuracy:

Testing results:

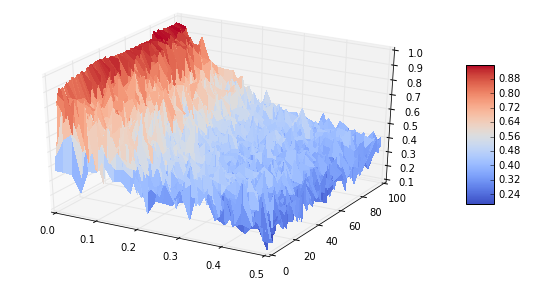
Recall results (prediction):

Validation - recall

Accumulated pattern (mobile data)

Call distribution (accumulated)

* Possibility of feature selection, and its impact on prediction (especially knn)



How to formulate a model by combing different data sources?

Market prediction (stock prediction),