Project 442-7: Hacking the Paris metro

Big data analysis of real-time RATP data

Sonia Vanier

Roberto Blanco

INF442 2019-2020 (X2018)

1 Introduction

In early 2017, RATP made available to the public (through a web service) its real-time data of the public transportation system in the Paris metropolitan area [1]. The consequence is that vast amounts of raw data (indeed, **big data**) are being produced all the time. Our goal here is to use the tools you learn in INF442 to begin to make sense out of this data.

The question has academic as well as a very real practical interest. One one hand, it is a natural target for the application of data science principles. On the other, it presents an immediate chance to improve the quality of life of millions of people around you every day, in small but meaningful ways. Here are some of the questions you may ask yourself:

- Can you predict the actual schedule of a line better than the RATP website and application? And can you use real-time data to adjust and get better, faster itineraries? (Spoiler: the answer is yes!)
- Do trains break down more often in the morning, afternoon or evening? On weekdays or on weekends?
- Do transport lines honor their nominal frequencies? Are passages evenly spaced or irregular? And can we work around it to make our travel faster?
- Can bus lines and their delays be used to estimate city traffic?
- What are the odds that the RER that is supposed to take you back to the École will leave you waiting at Massy-Palaiseau?

The good part is that now you have the tools and the data to answer them.

There is an additional benefit to this exercise. There is all too often an inevitable pedagogical divide between the conceptually clean, academic applications studied in class and the less ideal, more complex applications we, as engineers, encounter in our professional lives. By working with real data and solving real problems, you begin to close that gap.

Figure 1: Real-time estimates *in situ* (CC-BY-3.0, Poudou99)

2 Data set

For this project, we collect measurements taken from the real-time API exposed by RATP [5]. For a given line, direction and stop, we periodically poll a web service that reports estimated times of passage, like those seen in information screens (cf. Figure 1). These data, along with extensive information about the network is published under a permissive, share-alike copyleft license [3].

We will make available a set of historical data consisting of fine-grain, periodic measurements for rail and metro lines. You will analyze, data mine, and present results derived from these data. If you are interested in exploring online algorithms [7], direct access to the API can be requested. The study can be extended or shifted to other areas of the public transportation system, such as bus lines, if desired.

3 Methods and results

The scope of this proposal is chiefly concerned with the contents of Part 2 of the course: classification, detection of patterns and anomalies, etc. The computational treatment may include components of HPC as seen in Part 1, although this is not strictly necessary.

First, you will come up with a design that transforms raw data into tangible answers

to your questions about the data. To this end, the raw data produced by the API may need to be preprocessed to suit your needs. **Example:** to study the average frequency of passage in a station, we need to start from the actual times of arrival, not arrival estimates. In Figure 1, for Line 4 to Porte de Clignancourt, at 19:02 passages are estimated in 4 and 6 minutes' time. If everything goes according to plan, we will see arrivals at 19:06 and 19:08, and a wait of 2 minutes between metros (but the previous arrival happened before 19:02, so the wait will be at least 5 minutes...).

Second, you will implement or reuse (from your own TDs, class materials, or readily available data mining libraries like [6]) the necessary classification algorithms on the sanitized data, and analyze and present your results. You may concentrate on performing sophisticated analyses, or in summarizing and presenting the data in a way that is useful and easy to understand. An elementary but very visual example is this treatment of real-time data for the London tube, which plots the current location of all trains on a map of the city [2]. **Example:** to find out the average frequency of passage, we simply take times of arrival and average the waits. For large amounts of data, we could use MPI or Hadoop reduce operations on a cluster of computers, among other techniques you will learn in class.

The scope of the project is intentionally open-ended. While we will suggest projects of varying complexity (the list in Section 1 presents a few realistic questions, most relatively simple), you are free to propose problems that matter to you. Emphasis on the various components of data science will change depending on your interests. The only limit to what you can do with data is your imagination (and professional ethics! [4]).

References

[1] La RATP ouvre (enfin) ses données « temps réel ».
`http://www.lemonde.fr/economie/article/2017/01/05/la-ratp-ouvre-enfin-ses-donnees-temps-reel_5057926_3234.html`

[2] Live map of London Underground trains.
`http://traintimes.org.uk/map/tube/`

[3] Open Database License.
`https://opendatacommons.org/licenses/odbl/`

[4] What's up with big data ethics?
`http://radar.oreilly.com/2014/03/whats-up-with-big-data-ethics.html`

[5] Horaires Temps réel RATP.
`https://data.ratp.fr/explore/dataset/horaires-temps-reel/`

[6] `scikit-learn`: machine learning in Python
`http://scikit-learn.org/`

[7] Online algorithm.
`https://en.wikipedia.org/wiki/Online_algorithm`