



Robust Route Planner

Final Project

Team SAFJ

Serif Soner Serbest

Asli Yorusun

Fatine Benhsain

Jelena Banjac

Lab in Data Science

June 2019

Content

- Data Cleaning and Analysis
- Connection Graph
- Timetable
- Confidence Calculation
- Robust Route Calculation
- Visualization



Data Analysis

Variables

TRIP_DATE

Categorical

Distinct count 505
Unique (%) 0.0%
Missing (%) 0.0%
Missing (n) 0
Infinite (%) 0.0%
Infinite (n) 0

31.12.2018 306802
14.12.2018 297016
18.01.2019 296771
Other values (502) 126165637

[Toggle details](#)

TRIP_ID

Categorical

Distinct count 506250
Unique (%) 0.4%
Missing (%) 0.0%
Missing (n) 0
Infinite (%) 0.0%
Infinite (n) 0

85:849:104711-01912-1 7632
85:849:104705-01912-1 7632
85:849:104713-01912-1 7632
Other values (506247) 127043330

[Toggle details](#)

OPERATOR_ID

Categorical

Distinct count 13
Unique (%) 0.0%
Missing (%) 0.0%
Missing (n) 0
Infinite (%) 0.0%

85:849 55180840
85:3849 47377715
85:773 13034041
Other values (10) 1147363

Can be seen in the notebook:
SBB_Data_Analysis.ipynb

Data Cleaning

- **ADDITIONAL_TRIPS** where trip is additional
- **FAILED** if trip failed
- **PASSES_BY** if trip passes by the station
- **SCHEDULE/ACTIVE ARRIVE/DEPART TIMES** if any of the times is null
- Etc.



Connection Graph

- Distances between stations and Zurich (using coordinates)
- Stations around Zurich (≤ 10 km)
- Distances between stations
- Form a matrix with the connections
- Walking matrix: with the connections feasible during the *max_walking_time*
- Then: merge connection and walk matrices into a one single adjacency matrix of all possible paths



Timetable

- Using the available data, we can now form a timetable that links between two stations at specific times when connections exist
- Using a self join



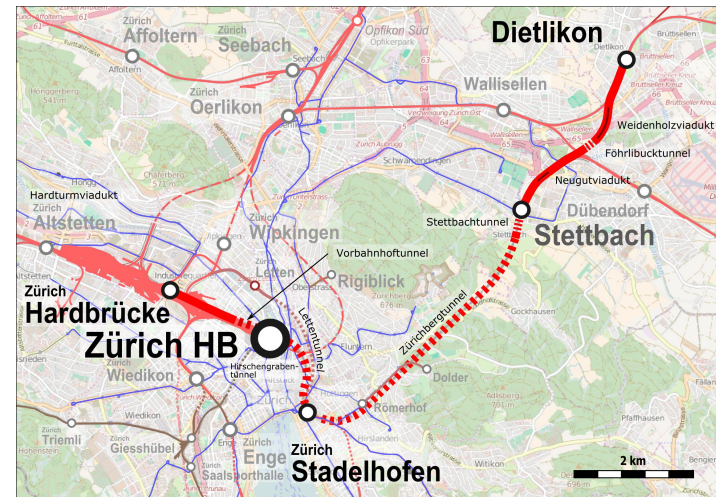
Confidence Calculation

- We created clusters using the data and several different attributes each time to find relationships between the attributes and the delays:
 - ACTUAL_ARRIVE_TIME
 - ACTUAL_DEPART_TIME
 - OPERATOR_ID
 - SERVICE_TYPE
 - STATION_ID
- Probability of delay computed using number of times delayed
- $P(\text{successful trip}) = \prod_i P(\text{segment}_i \text{ delayed})$



Robust Route Calculation

- Find best routes for given start and end destinations (reuse all previous methods)
 - by eliminating duplicate paths that involve only walking
 - by considering the probabilities of the arrival of each transportation on time
- Provide to user the best 3 options



route: 2

```
-----  
departure: 8503000 arrival: 8503307 departure_time: 2018-06-26 21:49:00 arrival_time: 2018-06-26 22:09:00 transport_type: Zug  
departure: 8503307 arrival: 8503016 departure_time: 2018-06-26 22:26:00 arrival_time: 2018-06-26 22:31:00 transport_type: Zug  
confidence = 0.9998768818457573  
-----
```


Visualization

DEMO

