

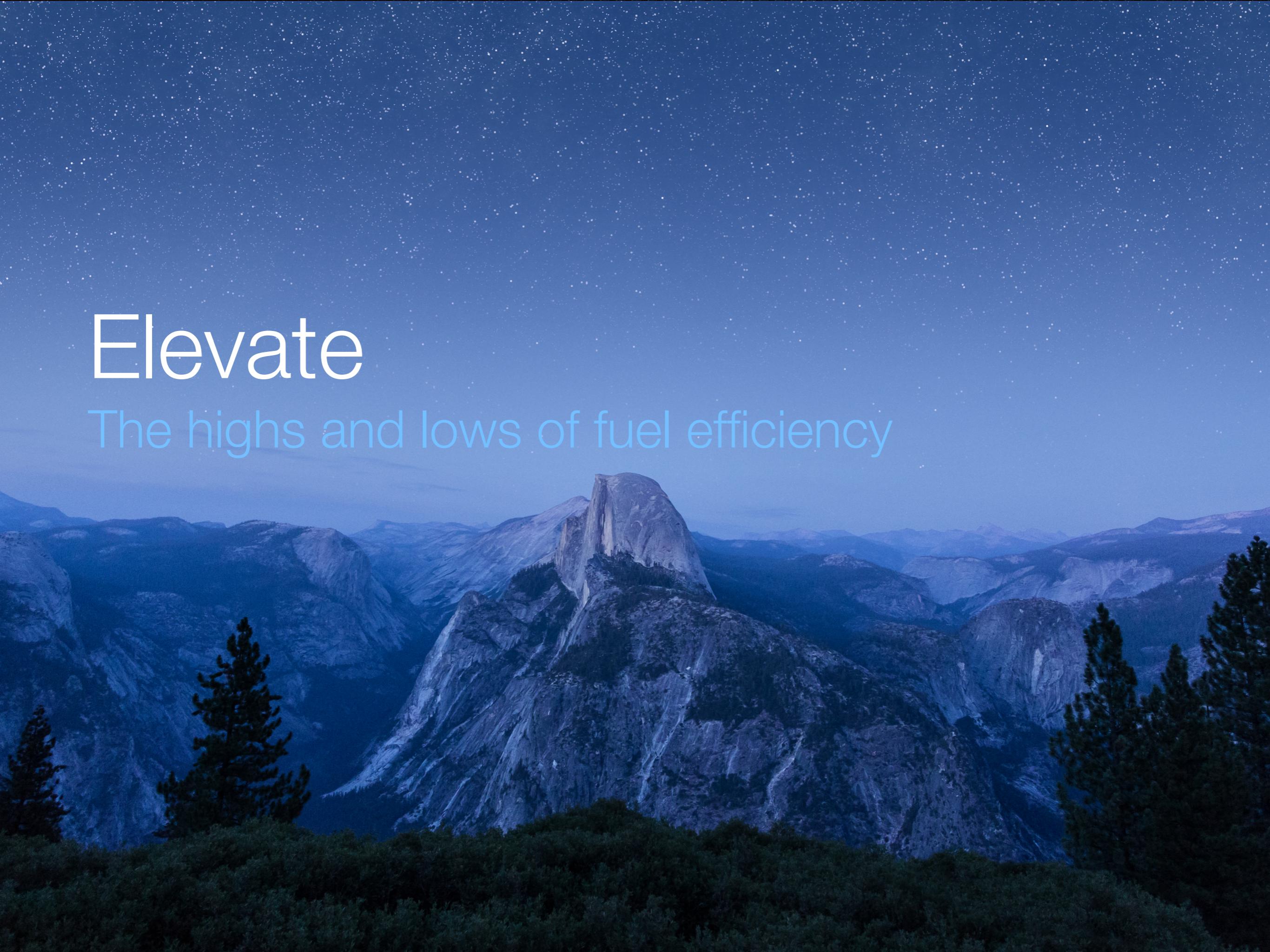
An aerial photograph of a vast, icy landscape, likely the Arctic or Antarctic. The foreground and middle ground are filled with large, white icebergs of various sizes, some with dark blue meltwater pools. In the distance, a massive, textured ice cap stretches towards the horizon under a clear, pale sky.

Like the icecaps,
Time efficiency isn't
COOL
anymore

Fuel efficiency is!



What if your
journey used less
fuel?

The background image shows a panoramic view of a mountain range, likely Half Dome in Yosemite. The sky is a deep, dark blue, filled with numerous small white stars, suggesting a night sky or a heavily overexposed daytime shot. The mountains in the foreground are dark and silhouetted against the sky, while the peaks in the distance are visible.

Elevate

The highs and lows of fuel efficiency

Elevate is the latest project in changing our impact on the environment.

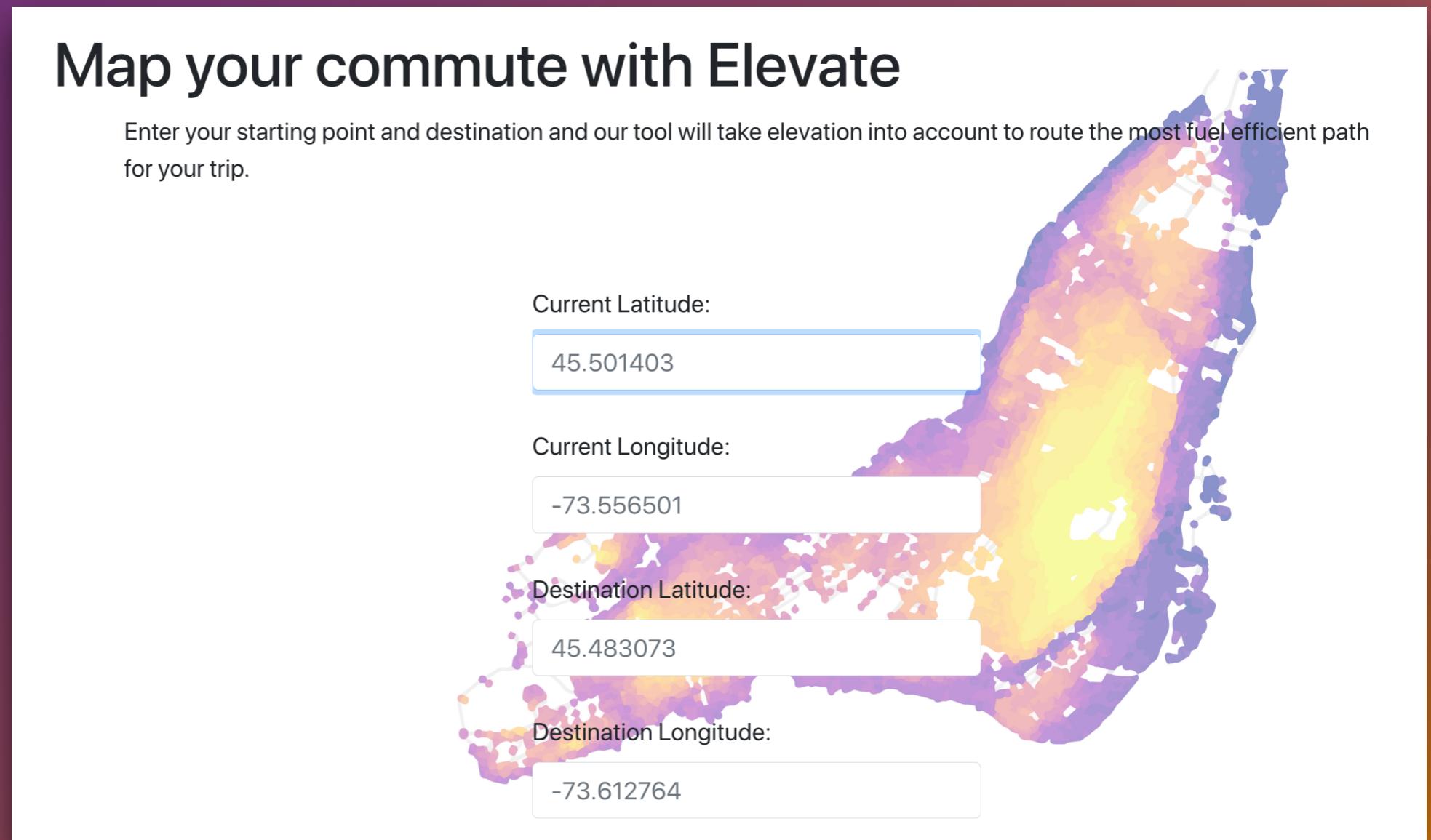
Through simple calculations of automobile fuel consumption in a journey including:

- **Change in elevation,**
- Car type (mass, fuel consumption),
- Stop signs encountered,
- Traffic lights encountered,
- Traffic, and
- Speed limit

Elevate optimises a route of least impedance, saving energy, money and often saving time.

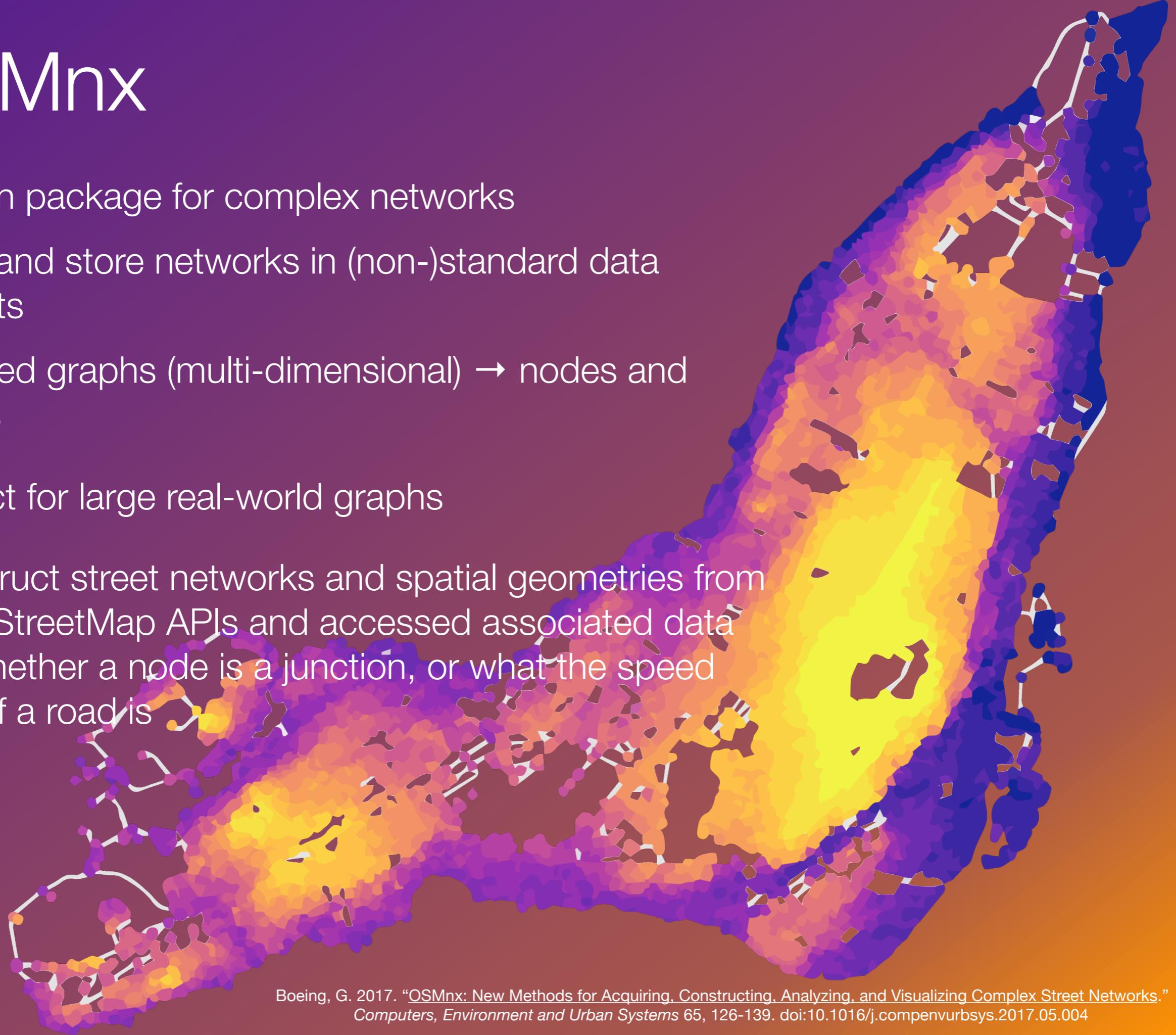
Web based app

- Calculate the most fuel efficient route between two coordinates!
- OpenStreetMap & Google Map API integration
- Python-based
- OSMnx



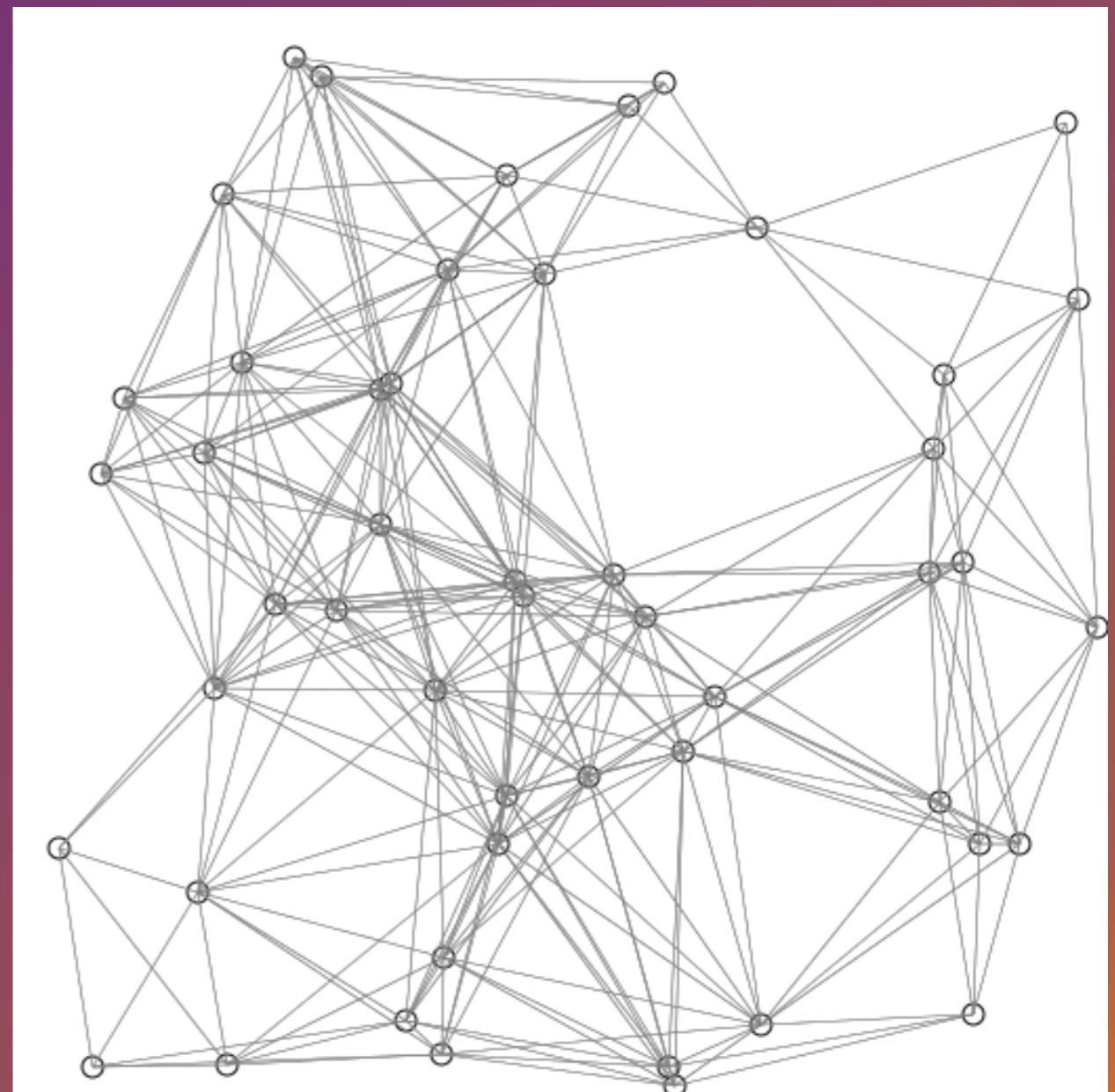
OSMnx

- Python package for complex networks
- Load and store networks in (non-)standard data formats
- Directed graphs (multi-dimensional) → nodes and edges
- Perfect for large real-world graphs
- Construct street networks and spatial geometries from OpenStreetMap APIs and accessed associated data i.e. whether a node is a junction, or what the speed limit of a road is



Dijkstra's Algorithm

- Find the shortest paths between nodes in a graph
- Pick the unvisited vertex with the lowest distance, calculate the distance from it to each unvisited neighbour, and updates the neighbour's distance if smaller



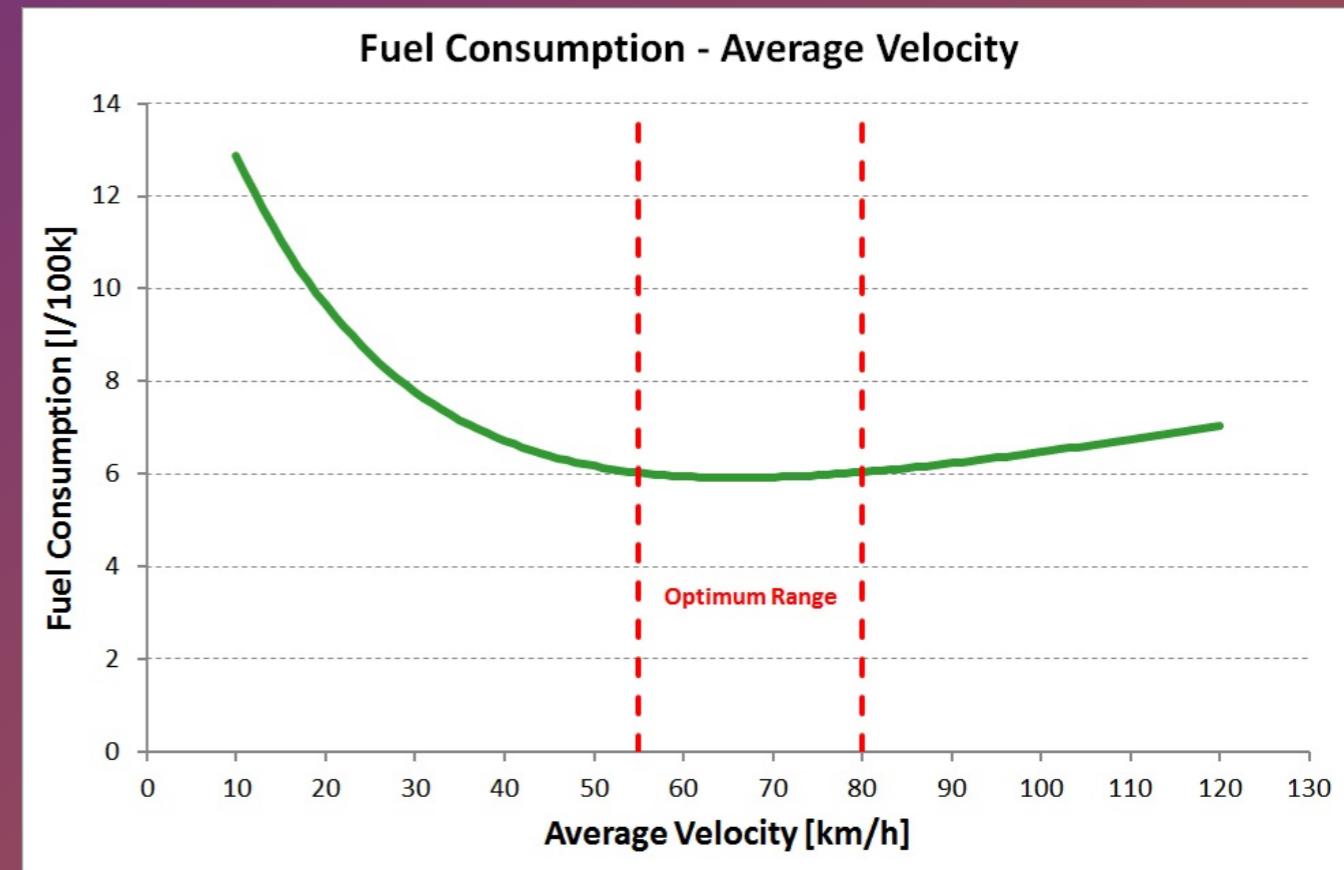
Dijkstra's Algorithm

- Find the shortest paths between nodes in a graph
- Pick the unvisited vertex with the lowest distance, calculate the distance from it to each unvisited neighbour, and updates the neighbour's distance if smaller
- We alter Dijkstra's algorithm to weight each edge with an impedance due to **extra fuel consumption**, not length



Fuel Consumption Efficiency

- Use average engine efficiency at road speed limit
- Calculate fuel consumption in change in elevation
- Montreal has average speed limit of 50 km/h
- Average fuel consumption = 7L/100km
- Example: Lowest 6m, Highest 233m.



Car	Distance [km]	Average Speed [km/h]	Travel Time	Fuel Consumed [l]
A	100	55	1h 49m	6.02
B	100	80	1h 15m	6.04
C	100	120	0h 50 m	7.03

Google Maps

Phi Centre 45.501403, -73.556501

to

Westmount: 45.483073, -73.61276

16 min

10.4km

=

0.728 L

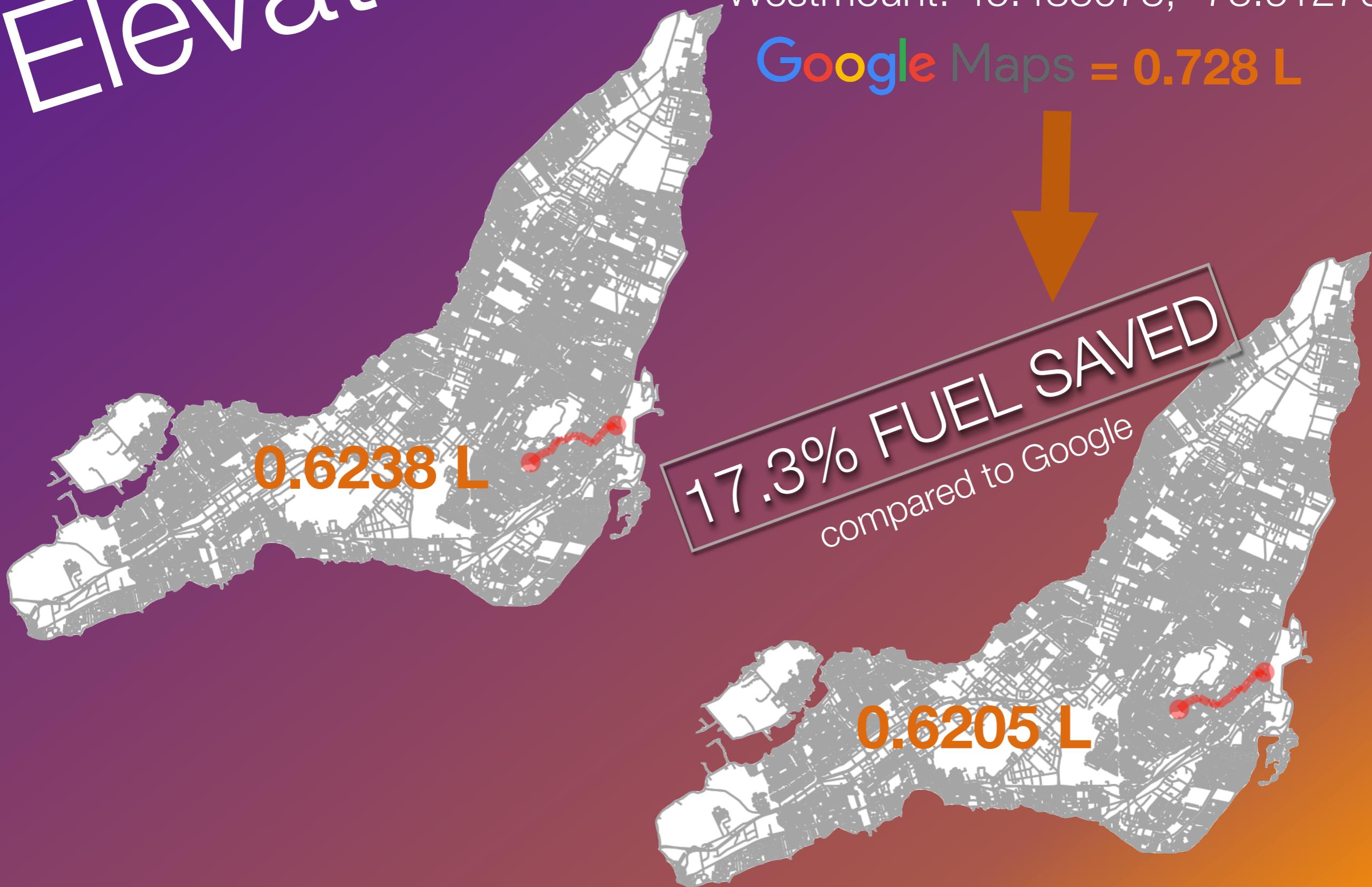


*This calculation is a simple fuel used per km calculation and does not take elevation into account. The journey selected was the first recommended by Google Maps at a specific time with specific conditions.

Elevate!

Phi Centre 45.501403, -73.556501
to
Westmount: 45.483073, -73.61276

Google Maps = 0.728 L



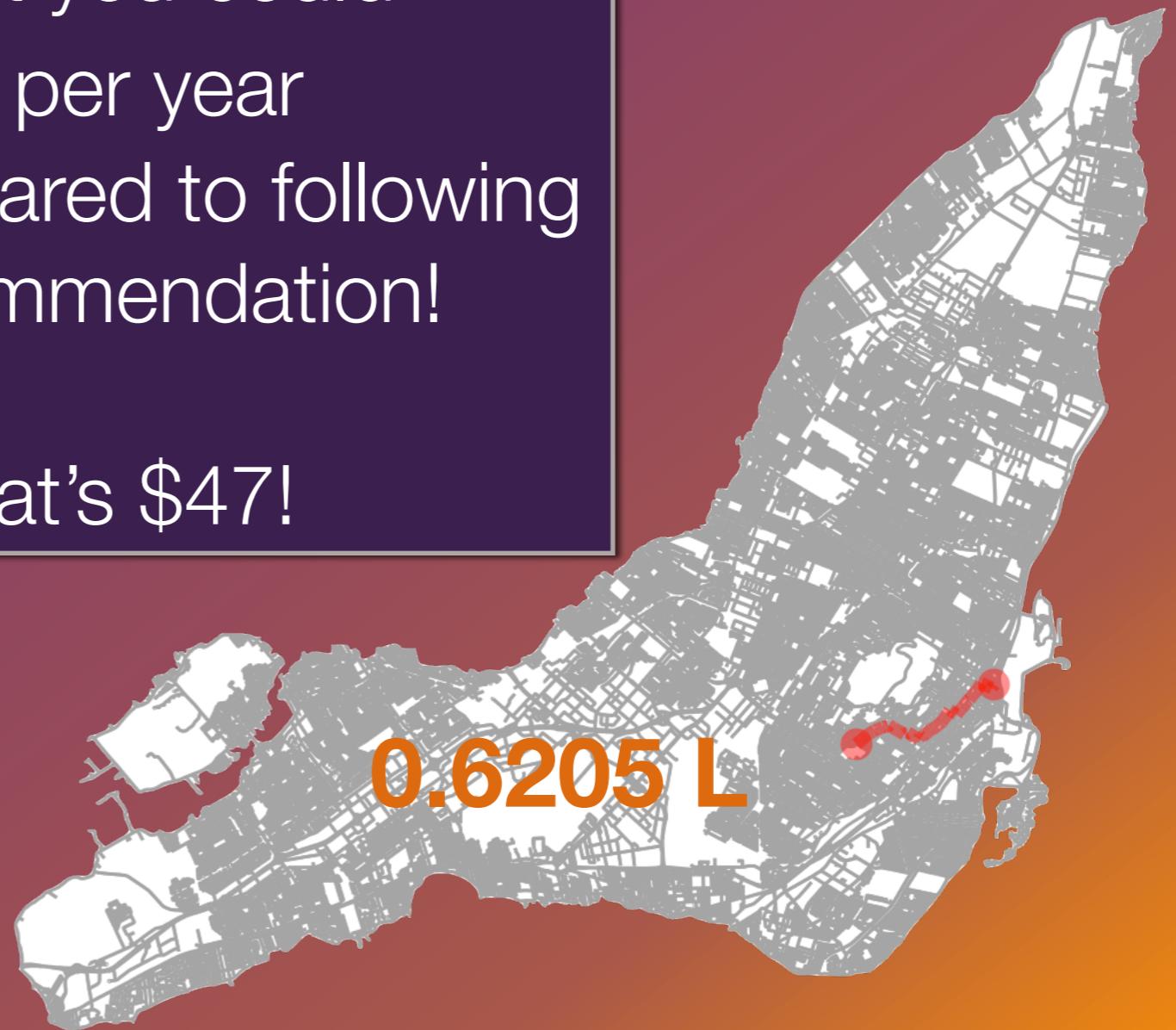
Elevate!

Phi Centre 45.501403, -73.556501
to
Westmount: 45.483073, -73.61276

Google Maps = 0.728 L

Elevate calculates that you could
SAVE 39 L of fuel per year
for this **ONE** journey compared to following
google maps first recommendation!

At today's price, that's \$47!



Elevate!

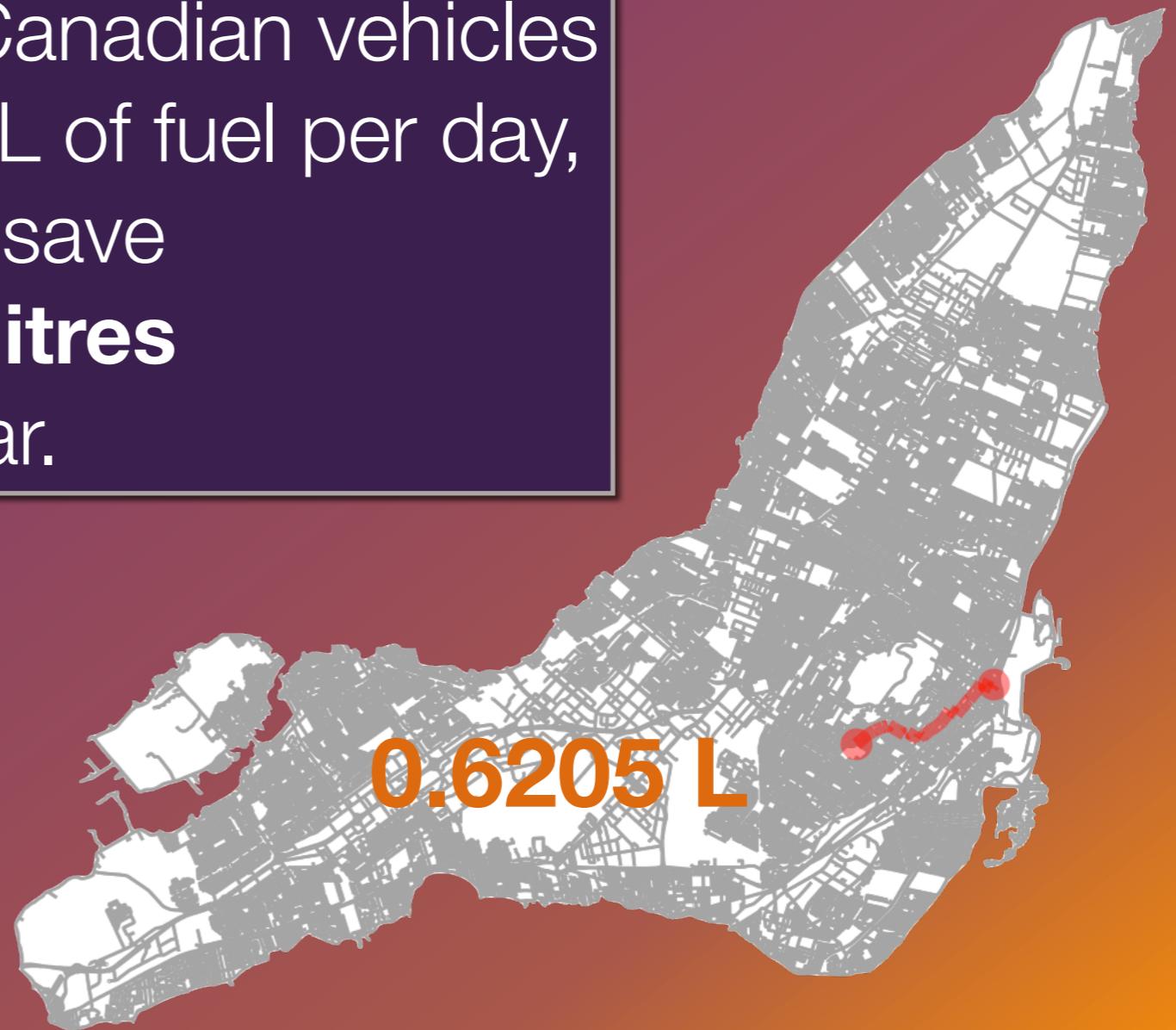
Phi Centre 45.501403, -73.556501
to
Westmount: 45.483073, -73.61276

Google Maps = 0.728 L

If the 22 million registered Canadian vehicles (under 4.5 tons), saved 0.1L of fuel per day, Canada would save **803 MILLION litres** of fuel a year.

PLUS this example was for a SINGLE ~7km journey!

Imagine the difference we could make!

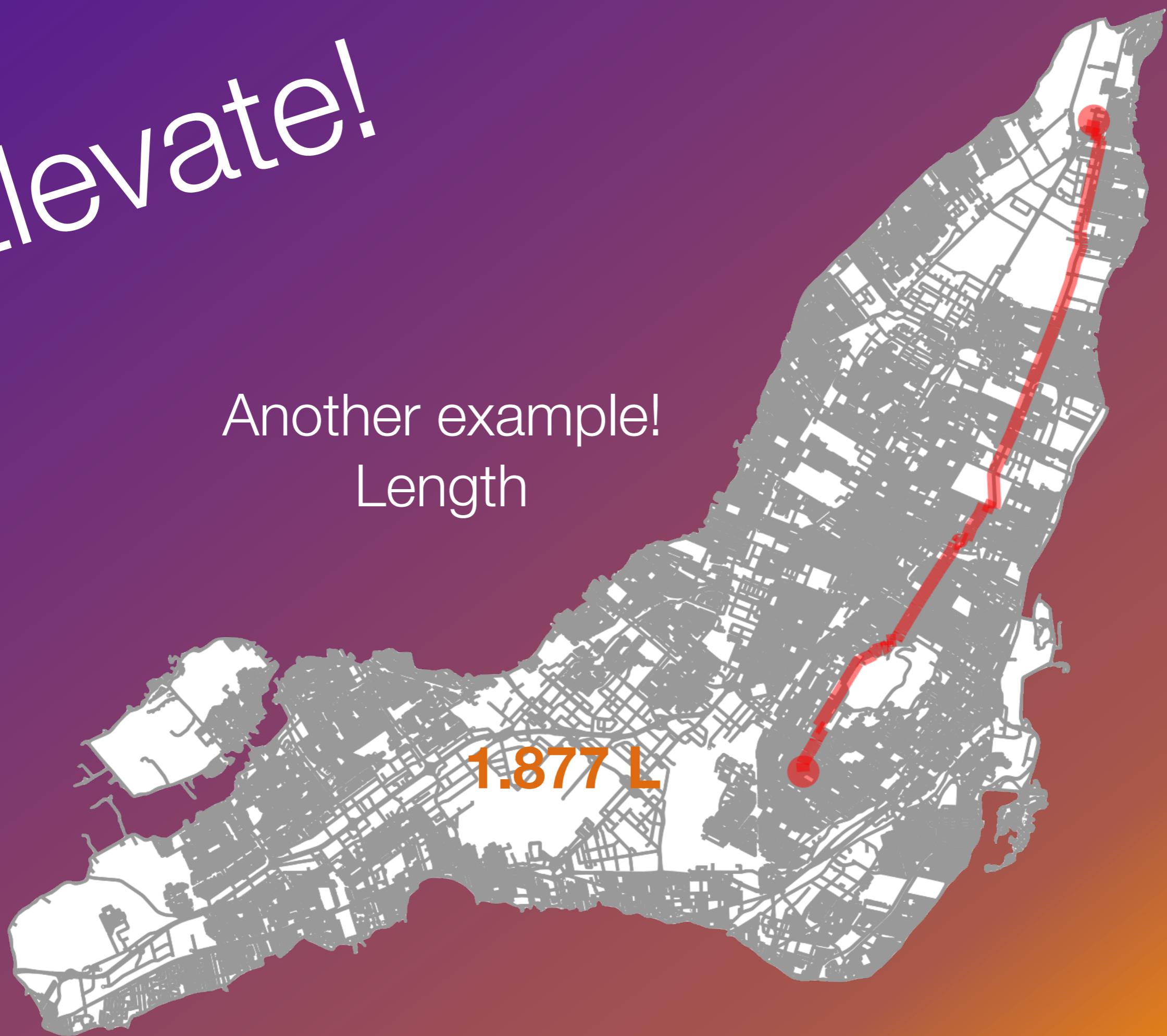


Elevate!

Another example!

Length

1.877 L



Elevate!

Another example!
Impedance



Future Development

- Optimise the speed of our app - reduce runtime to be able to compare data between two journeys
- Walkable, drivable, or bikable urban network data available
- Through each available mode of transport we calculate your fuel consumption
- Compare how much fuel/money/time you save for each possible journey
- Implement:
 - Fuel or car type i.e. hybrid vs. diesel!
 - Actual traffic lights encountered
 - Live traffic
 - Elevation about sea level



Elevate your efficiency