Efficiency of the Tokyo Metro System

A Comparative Analysis

By Arely Aguirre, George Anders, Andrew Pham, and Hoachen Yang

Introduction

Motivation

Tokyo's public transport system ranked the third best in the world

According to this survey, Tokyo's transport system is well-maintained and easy to use, even for non-Japanese speakers

Tokyo's Metro System is the Best in the World — Here's Why

Japan is famous for being at the forefront of technological innovations. Many of its biggest brands are tech companies, including Sony, Fujitsu, Hitachi, NEC, and Softbank, most of whom export to and invest in just about every part of the rest of the world.

Efficiency and Innovation: What Tokyo's Railway Network Can Teach Us

Tokyo's railway network is an impressive feat of efficiency and innovation.

INFRASTRUCTURE

Tokyo Has Built the World's Best Subway

Literature Review

- Core-Periphery Structure in Networks by Rombach et al
    - Introduces an equation for calculating core scores
- Efficiency in The Evolution of Metro Networks by Pei et al
    - Observes the "the spatial distribution of metro networks' most efficient nodes."
    - Develops a measurement of metro network efficiency "based on the shortest path calculated as path distances"
- Networks: An Introduction by Newman
    - Node betweenness
    - Node dynamical importance
    - Gaster-Newman Model

Measuring Efficiency Mathematically

Analysis of Important Stations

Similarities shared by important stations may provide clues as to what makes a network efficient

Robustness Analysis

Make small changes that negatively impact the network. Can the network still perform well in suboptimal conditions?

Analysis of Distance Traveled

Between any two stations, is the the distance traveled on the metro similar to the geographical distance?

Data Collection

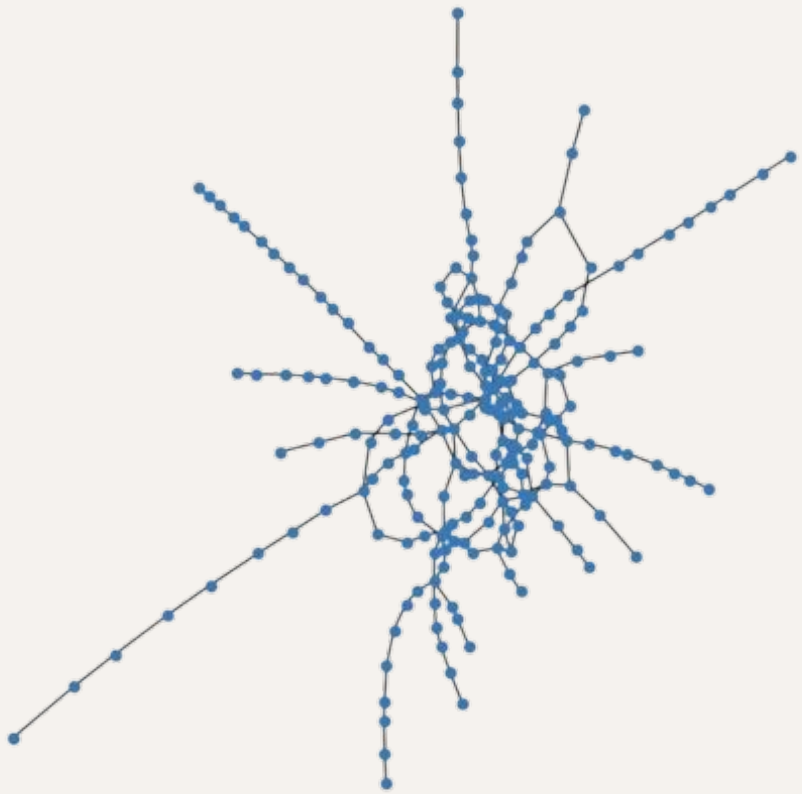
Tokyo Metro Data

- Data was in a usable, but suboptimal adjacency matrix
  - Combined information by station
  - Made adjustments to matrix values to easily create graph object

	G01	G02	G03	G04	G05	G06	G07	G08	G09
G01	-1.0	1.3	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
G02	1.3	-1.0	0.7	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
G03	-1.0	0.7	-1.0	0.7	-1.0	-1.0	-1.0	-1.0	-1.0
G04	-1.0	-1.0	0.7	-1.0	1.3	-1.0	-1.0	-1.0	-1.0
G05	-1.0	-1.0	-1.0	1.3	-1.0	0.9	-1.0	-1.0	-1.0
G06	-1.0	-1.0	-1.0	-1.0	0.9	-1.0	0.6	-1.0	-1.0
G07	-1.0	-1.0	-1.0	-1.0	-1.0	0.6	-1.0	0.8	-1.0
G08	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	0.8	-1.0	0.9
G09	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	0.9	-1.0

Tokyo Metro Graph

- Edges
  - Distances
  - Weights
- Nodes
  - Coordinates
  - Station Names



Madrid Metro Data

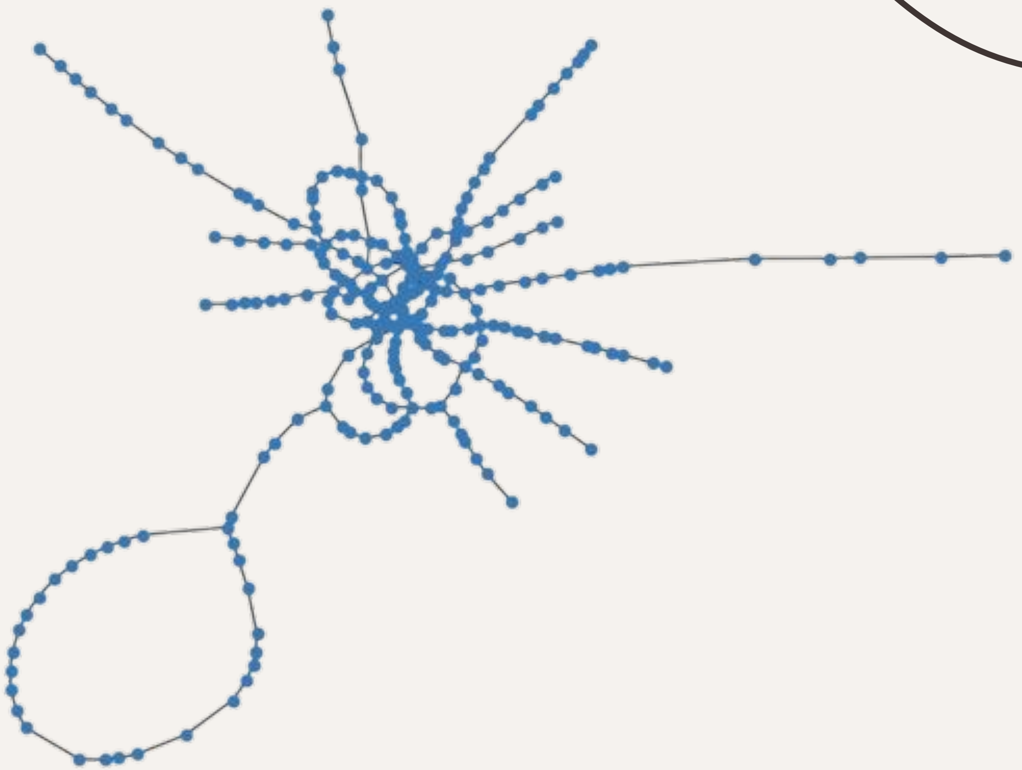
Source	Target	Weight	edge_name	edge_color	travel_seconds	longitude_Source
39	50	0.004048583	2	ED1C24	247.0	-3.70557
185	186	0.0018761726000000002	9	A60084	533.0	-3.47745
210	205	0.0031545741	12	A49800	317.0	-3.8212
213	214	0.0028571429	12	A49800	350.0	-3.87218
103	202	0.0033222590999999997	10	005AA9	301.0	-3.7610099999999997

latitude_Source	longitude_Target	latitude_Target	distance_meters	speed_ms
40.430009999999996	-3.7074199999999995	40.42484	595.1695167213638	2.4095931851067363
40.31902	-3.44752	40.303670000000004	3062.3757203955797	5.745545441642738
40.34969	-3.8121099999999997	40.345240000000004	916.8352682805719	2.8922248210743593
40.33512	-3.86354	40.3285	1038.959886229321	2.96845681779806
40.403240000000004	-3.7746199999999996	40.39698	1348.3860275285208	4.479687799098076

- Data was in table format with nearly all required data
  - Made slight adjustments to measurement units
  - Manually added nodes and edges to graph object

Madrid Metro Graph

- Edges
  - Distances
  - Weights
- Nodes
  - Coordinates
  - Station Names



Tokyo Network Statistics

```
Number of nodes: 223
Number of edges: 265
Average degree: 2.376681614349776
Average clustering coefficient: 0.0153462879920279
Diameter in km: 50.500000000000014
Average shortest path length in km: 11.927637054094443
Betweenness centrality (top 5 nodes):

('Otemachi', 0.3810619275320747)
('Kudanshita', 0.2412644633475394)
('Jimbocho', 0.2206488064875049)
('Shinjuku-sanchome', 0.19071899347937793)
('Ichigaya', 0.184768558086311)
```

Madrid Network Statistics

```
Number of nodes: 241
Number of edges: 275
Average degree: 2.2821576763485476
Density: 0.009508990318118948
Degree distribution: [0, 11, 195, 4, 22, 6, 2, 1]
Average clustering coefficient: 0.007804781663702826
Diameter in km: 58.449866424890416
Average shortest path length in km: 14.27420004421963
Betweenness centrality (top 5 nodes):

('GREGORIO MARAÑON', 0.32142145903259917)
('PRINCIPE PIO', 0.31128962932310195)
('NUEVOS MINISTERIOS', 0.30592224546722424)
('ALONSO MARTINEZ', 0.29607820729944173)
('AVENIDA DE AMERICA', 0.2818497482004279)
```

Important Stations
&
Robustness Analysis

Core-Periphery, Dynamical Importance Applied

- Network is meso-scale → Can measure its **core-periphery** structure
  - “Identifying densely connected core nodes” (Rombach, Porter et al)
- Core-periphery/dynamical importance insights
  - In this context, node dynamical importance tells us how the system is affected by the closure of a station

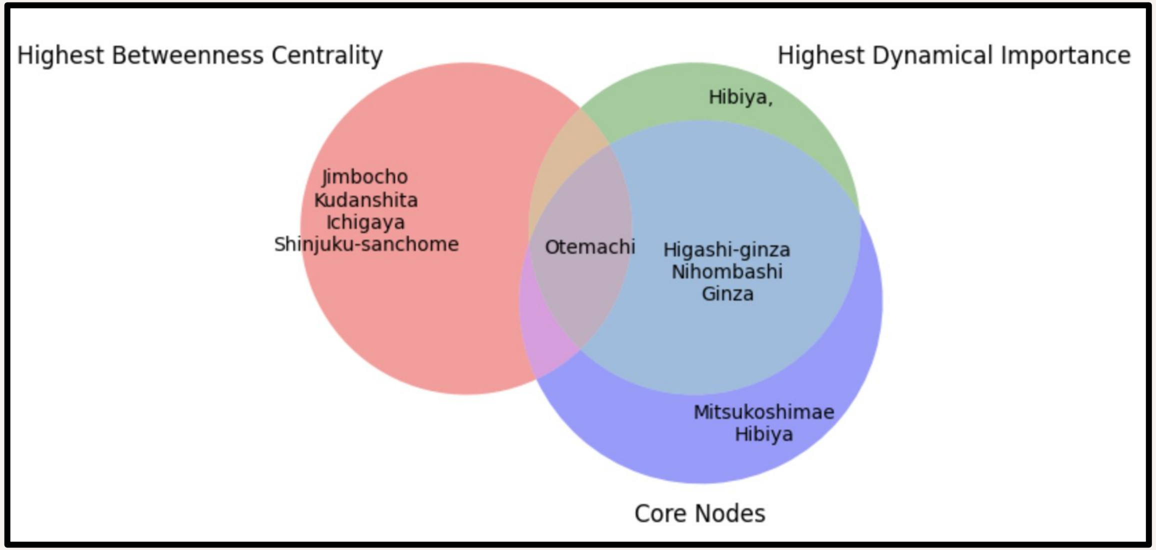
but is fully connected to the core. Their method aims to find a vector C of length N whose entries can be either 1 or 0. The ith entry C_i equals 1 if the corresponding node is assigned to the core, and it equals 0 if the corresponding node is assigned to the periphery. Let $C_{ij} = 1$ if $C_i = 1$ or $C_j = 1$, and let $C_{ij} = 0$ otherwise. Define

$$\rho_C = \sum_{i,j} A_{ij} C_{ij}, \qquad (4.1)$$

where the adjacency-matrix element A_{ij} represents the weight of the tie between nodes i and j and equals 0 if nodes i and j are not connected. This method of computing a discrete core-periphery structure seeks a value of ρ_C that is high compared to the expected value of ρ_C if C is shuffled such that the number of 1 and 0 entries is preserved but their order is randomized. The output is the vector C that gives the highest z-score for ρ_C.

Borgatti and Everet discrete core-periphery algorithm, taken from Rombach et al (*)

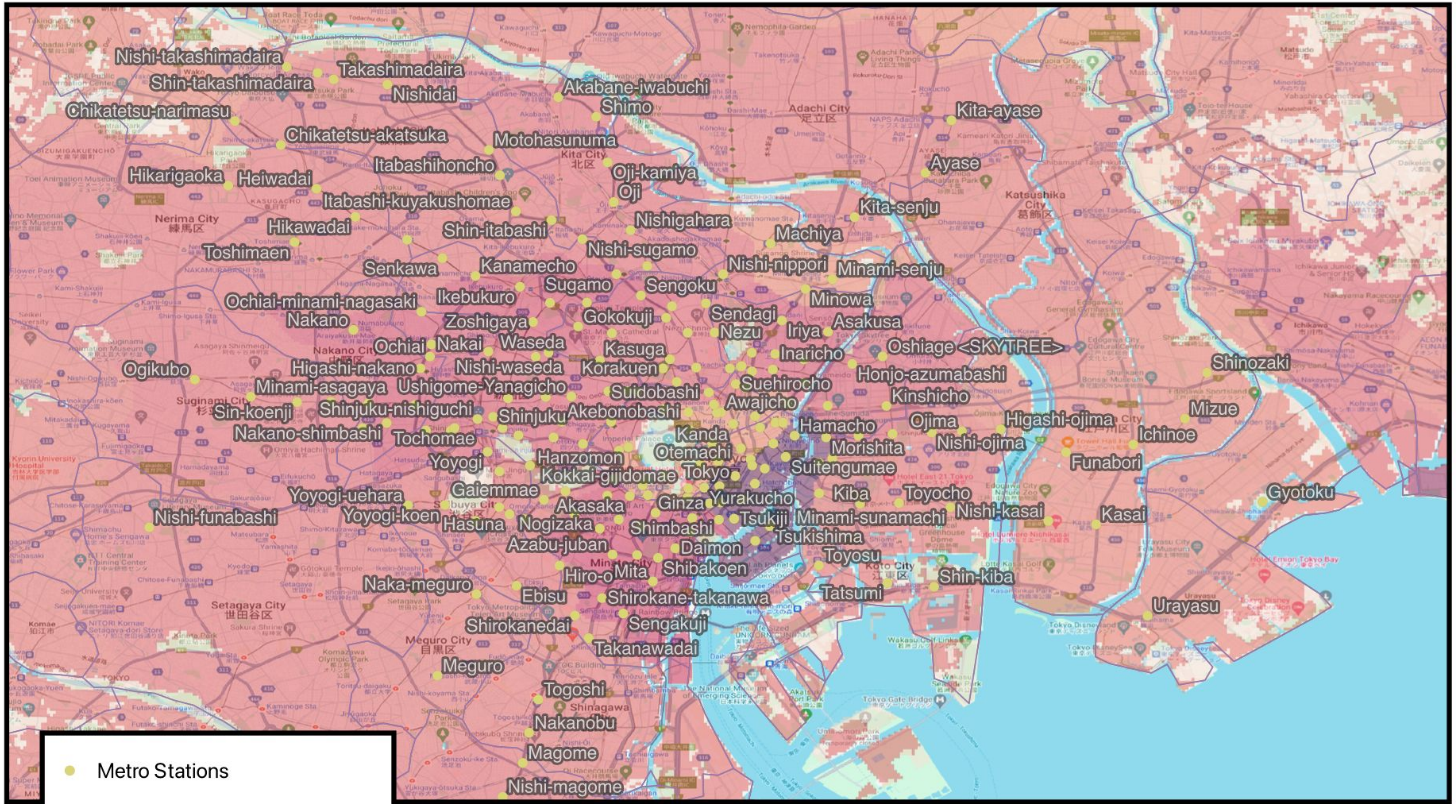
Comparisons of Important Nodes



- Aligns with Distance (Betweenness) vs. Weight (DI, Core) computations:
    - Core-periphery computations and dynamical importance both use ‘weight’ to compute measure (rather than distance)

- **Otemachi**
    - Based on it’s position in the network and in the city’s structure, it is both a part of the core and has high betweenness centrality

Tokyo Metro Stations and Population Density



Metro Stations

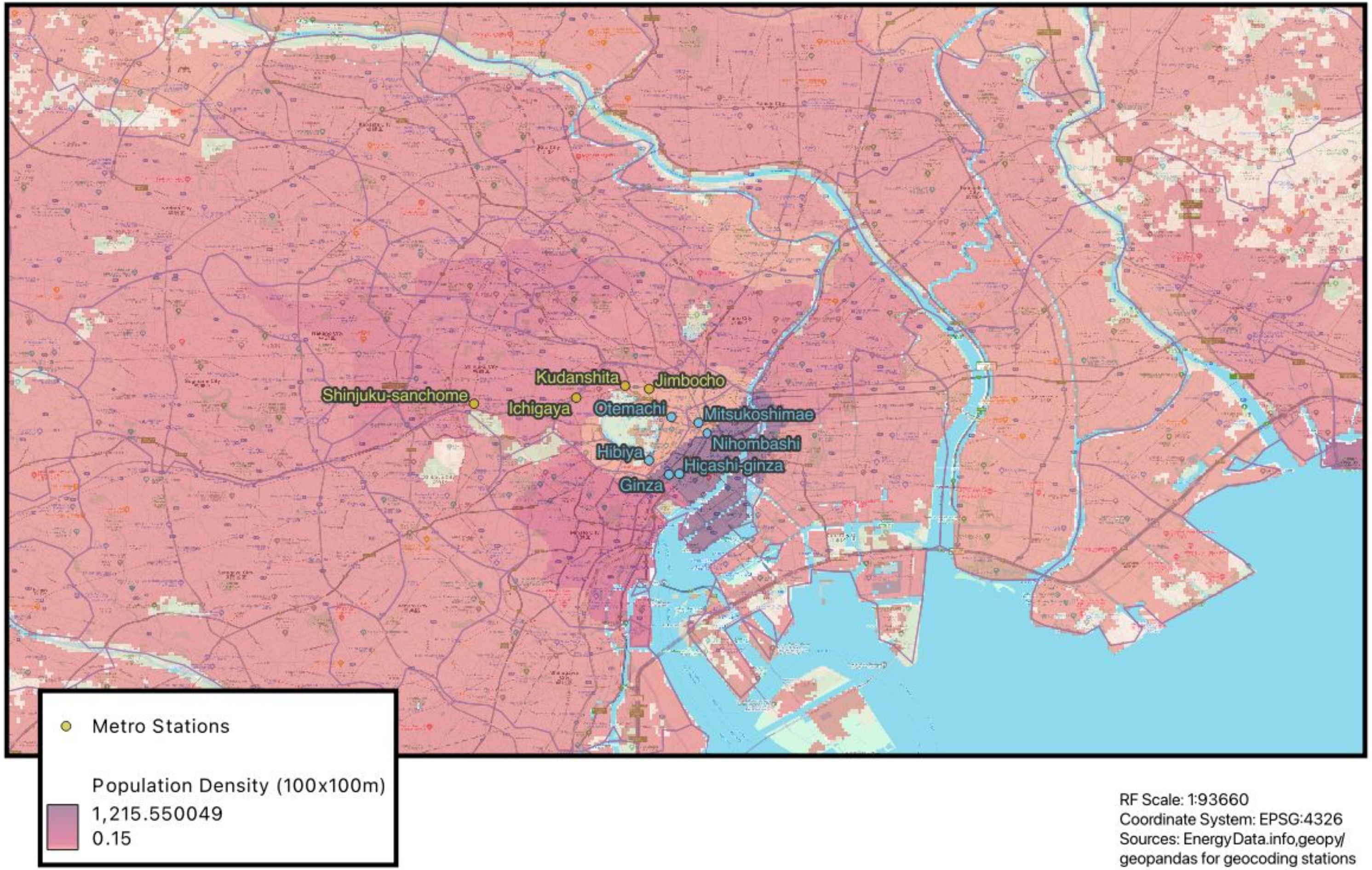
Population Density (100x100m)

1,215.550049

0.15

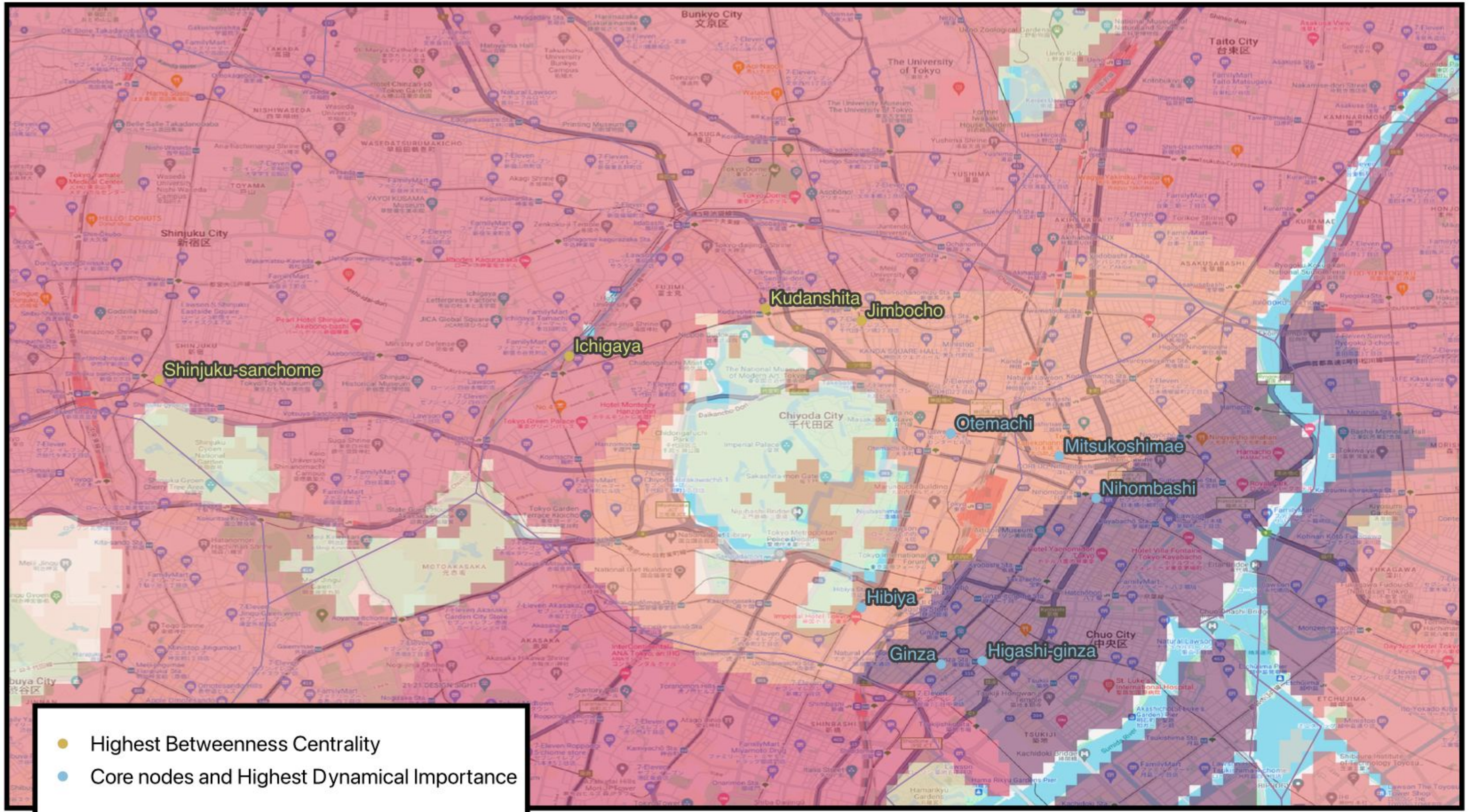
RF Scale: 1:93660
Coordinate System: EPSG:4326
Sources: EnergyData.info,geopy/
geopandas for geocoding stations

Tokyo Metro Stations and Population Density



RF Scale: 1:93660
Coordinate System: EPSG:4326
Sources: EnergyData.info,geopy/
geopandas for geocoding stations

Important Tokyo Metro Stations



Highest Betweenness Centrality
Core nodes and Highest Dynamical Importance

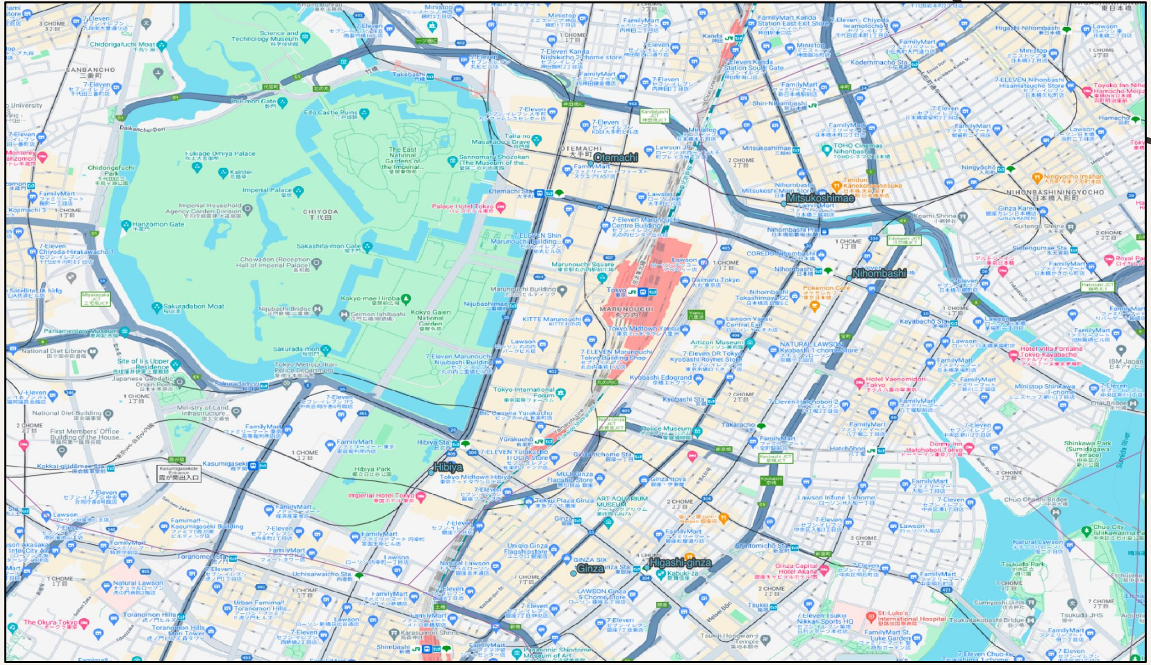
Population Density (100x100m)
1,215.550049
0.15

RF Scale: 1:25004
Coordinate System: EPSG:4326
Sources: EnergyData.info,geopy/
geopandas for geocoding stations

Observations

- Core nodes and nodes with the highest dynamical importance reside in the most densely populated areas in Tokyo (hotels, business Centers, shops, museums, etc)

- Nodes with highest betweenness centrality bridges busy downtown area (Otemachi, Higashi-ginza) to other densely populated areas further from the city center (Shinjuku, Nakano)



Highest Node Dynamical Importance and Core

- Positionally, Otemachi connects the core to other nodes with high betweenness centrality

Efficiency Measures

Gaster Newman Model

Measures efficiency by balancing # of edges vs mean geodesic distance

Greater λ → less edges → more sparse network

Lower λ → lower mean geodesic distance → denser network

For our purposes, we use λ =0.1

b_2 = average distance between nodes ÷ average speed of a metro

$T_{ij} = time\ to\ go\ from\ stop\ j\ to\ stop\ i = b_1 + b_2 r_{ij}$

$b_1 = constant = total\ waiting\ time\ (ex:\ time\ you\ wait\ in\ airport)$

$b_2 = constant = total\ amount\ of\ time\ traveling/moving$

Cost Function: $E(m, L) = \lambda m + (1 - \lambda)L,\ \lambda \in [0, 1]$

- $m = number\ of\ edges$
- $L = mean\ geodesic\ distance\ between\ nodes\ using\ T_{ij}\ for\ distance$

Analyzing Importance with Node Percolation

Individual removal

Recalculate the values after removing each node

Random removal

Individual & mass (5 at a time) random removal of nodes

Targeted random removal

Individual random removal of highest value nodes (top 10)

Tokyo Efficiency

b_1 = ~5 minutes = 0.08 hours

b_2 = ~11.927 km ÷ 30 km/hr ≈ 0.397

$E(m,L)$ = 31.541

average cost after removing random node: 31.365

delta: 0.175

average cost after removing important node: 31.156

delta: 0.385

Estimated average cost after mass removing random nodes: 30.701

delta: 0.84

lowest 10 values after removal:

	Node	Cost
94	Shimbashi	31.227181
10	Azabu-juban	31.202186
38	Kasumigaseki	31.197890
84	Omote-sando	31.187659
24	Higashi-ginza	31.178164
22	Hibiya	31.158469
63	Nagatacho	31.133206
68	Nihombashi	31.128378
6	Aoyama-itchome	31.105500
15	Ginza	31.042167

Madrid Efficiency

b_1 = ~11 minutes = 0.183 hours

b_2 = ~14.551 km ÷ 32 km/hr ≈ 0.454

$E(m,L)$ = 35.669

average cost after removing random node: 35.616

delta: 0.052

average cost after removing important node: 35.330

delta: 0.338

Estimated average cost after mass removing random nodes: 35.285

delta: 0.384

lowest 10 values after removal:

	Node	Cost
63	PARQUE DE LOS ESTADOS	35.357156
0	SAN BERNARDO	35.352509
107	BILBAO	35.345737
17	GOYA	35.344224
34	CUATRO CAMINOS	35.341858
37	NUÑEZ DE BALBOA	35.333329
21	CANAL	35.333263
25	CALLAO	35.321744
66	DIEGO DE LEON	35.302279
59	ALONSO MARTINEZ	35.275597

Global Efficiency

The value of E_g depends on the scale of path distance and, in general, $E_g \epsilon [0, \infty]$, which cannot be effortlessly generalized to weighted networks. For this reason, Latora et al.[38] rescaled the value of global efficiency in [0, 1] by considering an idealized proxy of G, G_{ideal}, having maximum efficiency. $E_{g,ideal}$ is based on pairwise physical distances l_{ij}:

$$E^t_{g,ideal} = \frac{1}{N(N-1)} \sum_{j \neq i} \frac{1}{l_{ij}}. \quad (2)$$

For all $i, j \epsilon V$, constraint $l_{i,j} \leq d_{i,j}$ is always satisfied whether G is disconnected or connected and, hence, $E_{g,ideal} \geq E_g$. A correct normalization of E_g is then possible to use $E_{g,ideal}$ resulting from a physically-grounded

Local Efficiency

Recall *L*, mean geodesic distance between nodes, from our cost function.

We will use this for our measure of local efficiency

Tokyo Global Efficiency

Lowest 10 values (Global Efficiency):

	Node	Global Efficiency
6	Aoyama-itchome	0.125449
63	Nagatacho	0.125393
59	Mitsukoshimae	0.125345
28	Hongo-sanchome	0.125324
15	Ginza	0.125225
38	Kasumigaseki	0.125152
51	Kudanshita	0.124873
68	Nihombashi	0.124632
102	Shinjuku-sanchome	0.123825
87	Otemachi	0.119312

Initial Global Efficiency: 0.1282

average efficiency after removing random node: 0.1274

delta: 0.0008

average efficiency after removing important node: 0.1244

delta: 0.0038

Estimated average efficiency after mass removing random nodes: 0.1236

delta: 0.0046

Madrid Global Efficiency

Initial Global Efficiency: 0.1332 (T:0.1282)

average efficiency after removing random node: 0.1323 (T:0.1274)

delta: 0.0009 (T:0.0008)

average efficiency after removing important node: 0.1292 (T:0.1244)

delta: 0.0039 (T:0.0038)

Estimated average efficiency after mass removing random nodes: 0.1283 (T:0.1236)

delta: 0.0049 (T:0.0046)

Lowest 10 values (Global Efficiency):

	Node	Global Efficiency
107	BILBAO	0.130911
54	TIRSO DE MOLINA	0.130864
49	MANUEL BECERRA	0.130741
80	LA LATINA	0.130643
34	CUATRO CAMINOS	0.130283
59	ALONSO MARTINEZ	0.129876
26	OPERA	0.128921
39	PRINCIPE PIO	0.128919
24	SOL	0.126218
36	AVENIDA DE AMERICA	0.125525

Tokyo Local Efficiency

Highest 10 values (Average Shortest Path Length):

	Node	Average Shortest Path Length
87	Otemachi	6.544449
102	Shinjuku-sanchome	6.049191
96	Shin-ochanomizu	5.953841
51	Kudanshita	5.901233
28	Hongo-sanchome	5.871073
101	Shinjuku-nishiguchi	5.869107
133	Yushima	5.846499
120	Ueno	5.823180
63	Nagatacho	5.814674
30	Ichigaya	5.809510

Initial *L* value = 5.601

average efficiency after removing random node: 5.676

delta: 0.075

average efficiency after removing important node: 5.531

delta: 0.070

Estimated average efficiency after mass removing random nodes: 6.782

delta: 0.380

Madrid Local Efficiency

Initial *L* value = 9.077 (5.601)

average efficiency after removing random node: 9.276 (T:5.676)

delta: 0.199 (T:0.075)

average efficiency after removing important node: 8.982 (T:5.531)

delta: 0.094 (T:0.070)

Estimated average efficiency after mass removing random nodes: 9.866 (T:6.782)

delta: 0.788 (T:0.380)

Highest 10 values (Average Shortest Path Length):

	Node	Average Shortest Path Length
123	SAN NICASIO	10.720574
3	PARQUE LISBOA	10.699256
39	PRINCIPE PIO	10.578071
64	LEGANES CENTRAL	10.476480
73	ALCORCON CENTRAL	10.417648
116	LAGO	10.369890
115	BATAN	10.324701
65	HOSPITAL SEVERO OCHOA	10.249529
33	PARQUE OESTE	10.160097
121	CASA DEL RELOJ	10.039472

Analysis of Distance Traveled

Shortest Path Calculation

Using Dijkstra's algorithm to find shortest paths in the metro network

```
def print_shortest_path(graph, source, target):
    path = nx.dijkstra_path(graph, source, target)
    path_length = nx.dijkstra_path_length(graph, source, target)
    print("Total path length:", path_length)
    for i in range(len(path) - 1):
        u, v = path[i], path[i + 1]
        weight = graph[u][v]['weight']
        print(f"{u} -> {v}: {weight}")
```

Geographical Distance Calculation

Using geodesic distance to measure actual distances between stations (coodinates, longitude, latitude)

```
def calculate_geographical_distance(station1, station2):
    coord1 = distances_G.nodes[station1]["coordinates"]
    coord2 = distances_G.nodes[station2]["coordinates"]
    return geodesic(coord1, coord2).kilometers
```

Efficiency Calculation:

$$\text{Efficiency} = \frac{\text{Shortest Path Distance}}{\text{Geographical Distance}}$$

Sample Output for All Pairs Shortest Path Lengths in Tokyo

Visualize

	Akabane-iwabuchi f. 0.0 - 37.50000000...	Shimo float64 0.0 - 36.40000000...	Akabanebashi floa... 0.0 - 34.80000000...	Azabu-juban float64 0.0 - 34.00000000...	Daimon float64 0.0 - 33.6	Akasaka float64 0.0 - 34.1	Kokkai-gijidomae fl... 0.0 - 33.50000000...
Aka...	0	1.1	17.7	17.6	16.4	17.1	16.3
Shi...	1.1	0	16.6	16.5	15.3	16	15.2
Oji-...	2.7	1.6	15	14.9	13.7	14.4	13.6
Oji	3.9	2.8	13.8	13.7	12.5	13.2	12.4
Nis...	4.9	3.8	12.8	12.7	11.5	12.2	11.4
Ko...	6.3	5.2	11.4	11.3	10.1	10.8	10
Ho...	7.7	6.6	10	9.9	8.7	9.4	8.6
Tod...	8.6	7.5	9.1	9	7.8	8.5	7.7
Kor...	9.9	8.8	7.8	7.7	6.5	7.2	6.4
Ho...	10.7	9.6	7	7.8	5.7	6.4	5.6

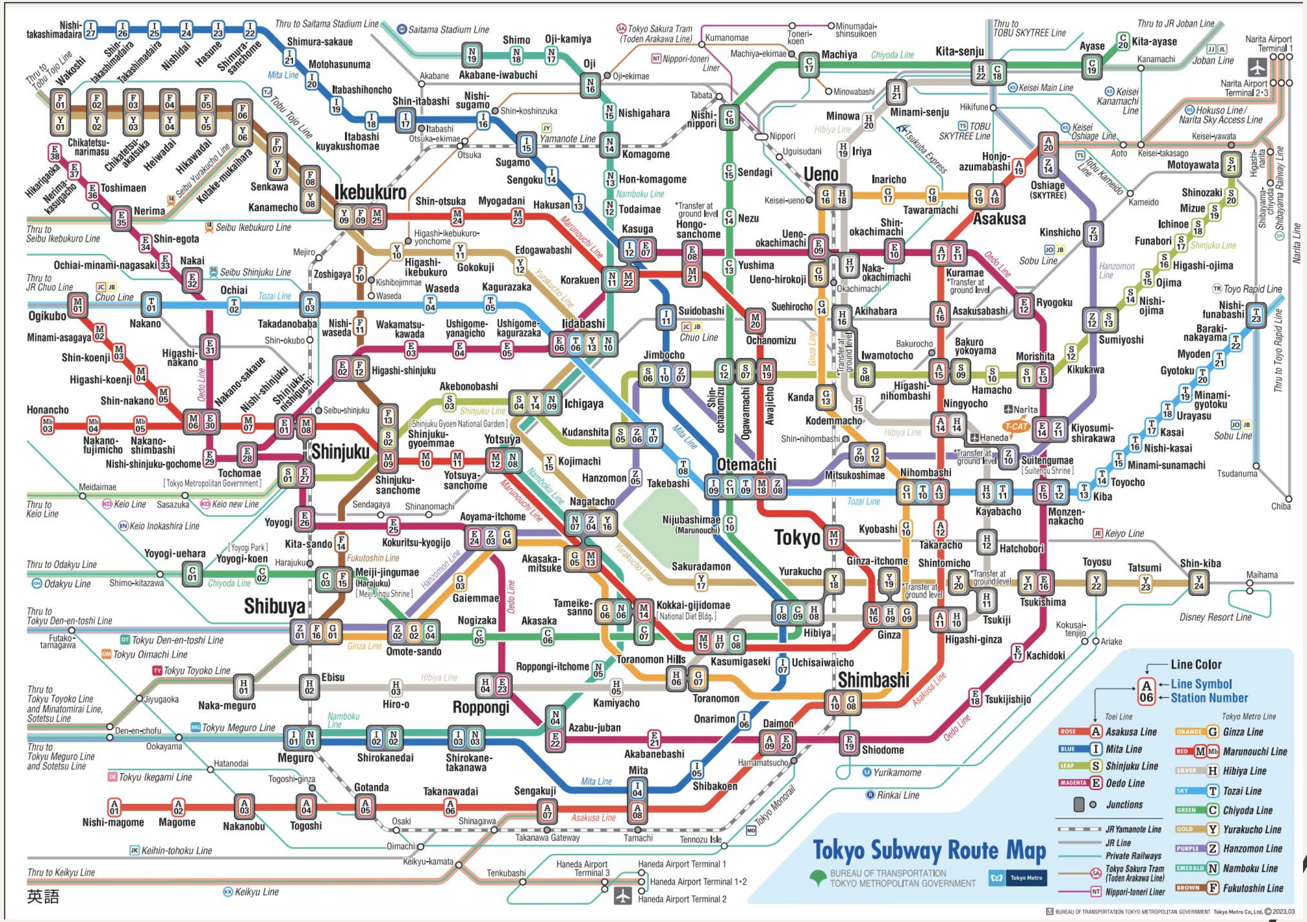
Tokyo Metro

Example calculation for stations **Tameiki-sanno** and **Akasaka-mitsuke**

- Geographical distance: 0.72 km
- Shortest path distance: 0.9 km
- Sample Efficiency: 1.24

Average efficiency for Tokyo: 1.65

Tokyo Metro Map



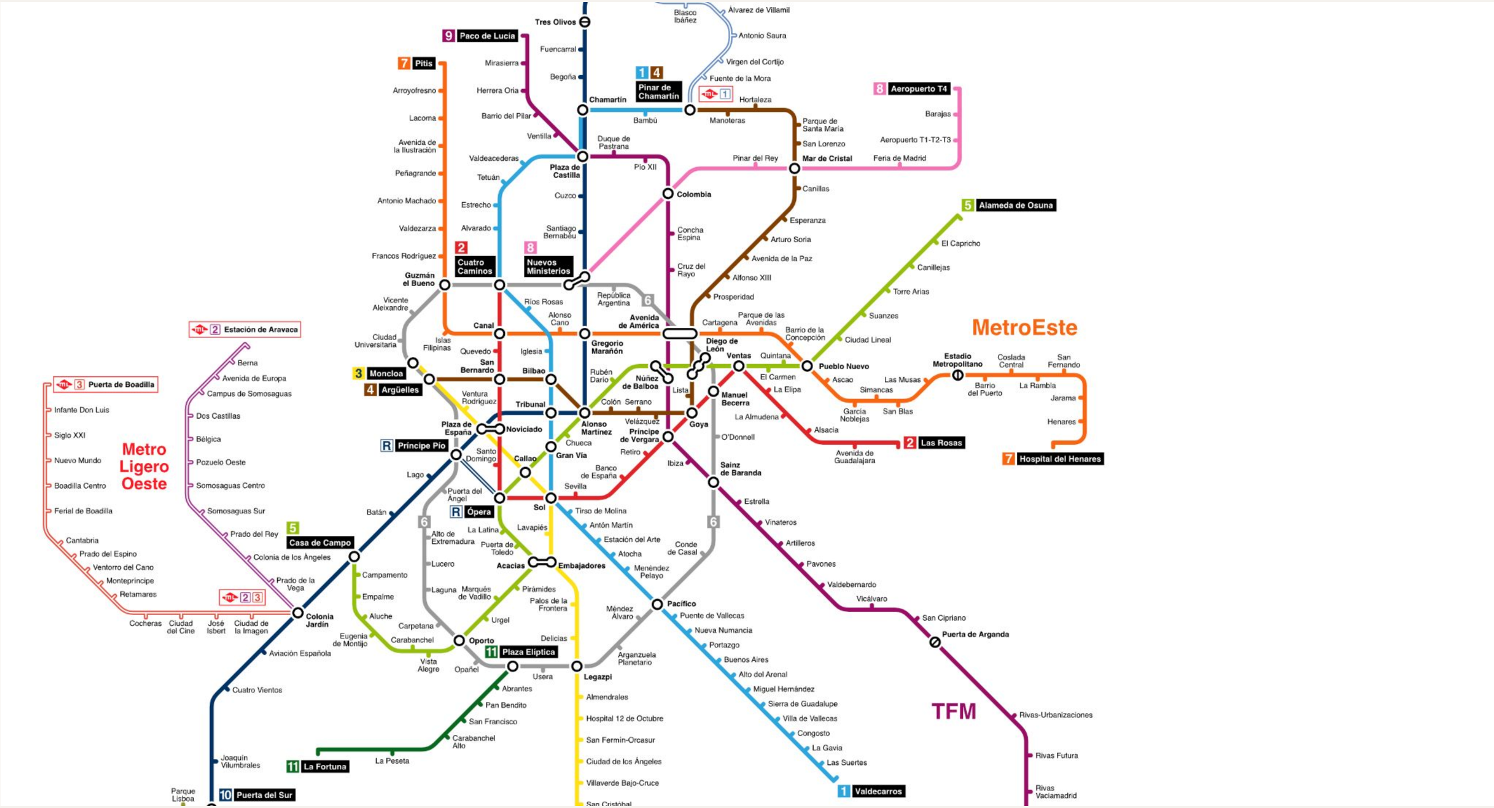
Madrid Metro

Example calculation for stations **Plaza de Castilla** and **Parque Europa**

- **Geographical distance: 22.50 km**
- **Shortest path distance: 33.96 km**
- **Sample Efficiency: 1.51**

- **Average efficiency for Madrid: 1.46**

Madrid Metro Map



Interpretation of a Fraction Close to 1

- Basically indicates that the shortest path distance is very close to the geographical distance.
- This suggests that the metro network is well-designed, with more direct and efficient routes.
- Higher efficiency values (much greater than 1) imply that the metro routes are more circuitous, increasing the travel distance for passengers.

Madrid's efficiency is closer to 1 compared to Tokyo's, suggesting that Madrid's metro network is slightly more efficient in terms of route directness and optimization.

Concluding Remarks

Key Findings

- If a metro system is efficient, nodes determined to be important by the network structure should reside in densely populated areas or should serve as a bridge between two densely populated areas.
- Efficient metro systems are more resistant to the negative effects of station closures compared to other metro systems
- Efficient networks on average, should not require traveling more than double the geographical distance between two stations when traveling between these stations on the metro

Limitations

- Availability of data
  - Difficult to find metro network data in an easy to use format
  - Possible to use a web scraper to gather metro network information, but can be very labor intensive
- Outdated data
  - Available data may be outdated, as many metro systems are constantly expanding

Future Work

- We can apply these methods to metro networks that are known to be efficient
  - London
  - Seoul
  - New York City
- We can perform a comparative analysis that compares multiple poorly ranked metro systems to highly ranked metro systems
  - Los Angeles

References

- Pei, Aihui, et al. "Efficiency in the evolution of Metro Networks." *Scientific Reports*, vol. 12, no. 1, 18 May 2022, https://doi.org/10.1038/s41598-022-12053-3.
- Rombach, Puck, et al. "Core-periphery structure in networks (revisited)." *SIAM Review*, vol. 59, no. 3, Jan. 2017, pp. 619–646, https://doi.org/10.1137/17m1130046.
- Newman, Mark E. J. *Networks: An Introduction*. Oxford University Press, 2010, pp. 1–740.

Thank You

Any questions?