CE88: Lecture 1

Marta C. Gonzalez

Associate Professor of City and Regional Planning and Civil and Environmental Engineering, Faculty Scientist at the LBNL

My team develops computational models to analyze digital traces to estimate the demand on urban infrastructure in relation to energy and mobility. Examples include: traffic gridlocks and the integration of electric vehicles in the power grid, policy of solar energy adoption, and habits in spending behavior.

Kseniya Usovich

Studies a Bachelor's Degree in Data Science and Cognitive Science
Modules Developer at UC Berkeley Division of Data Sciences

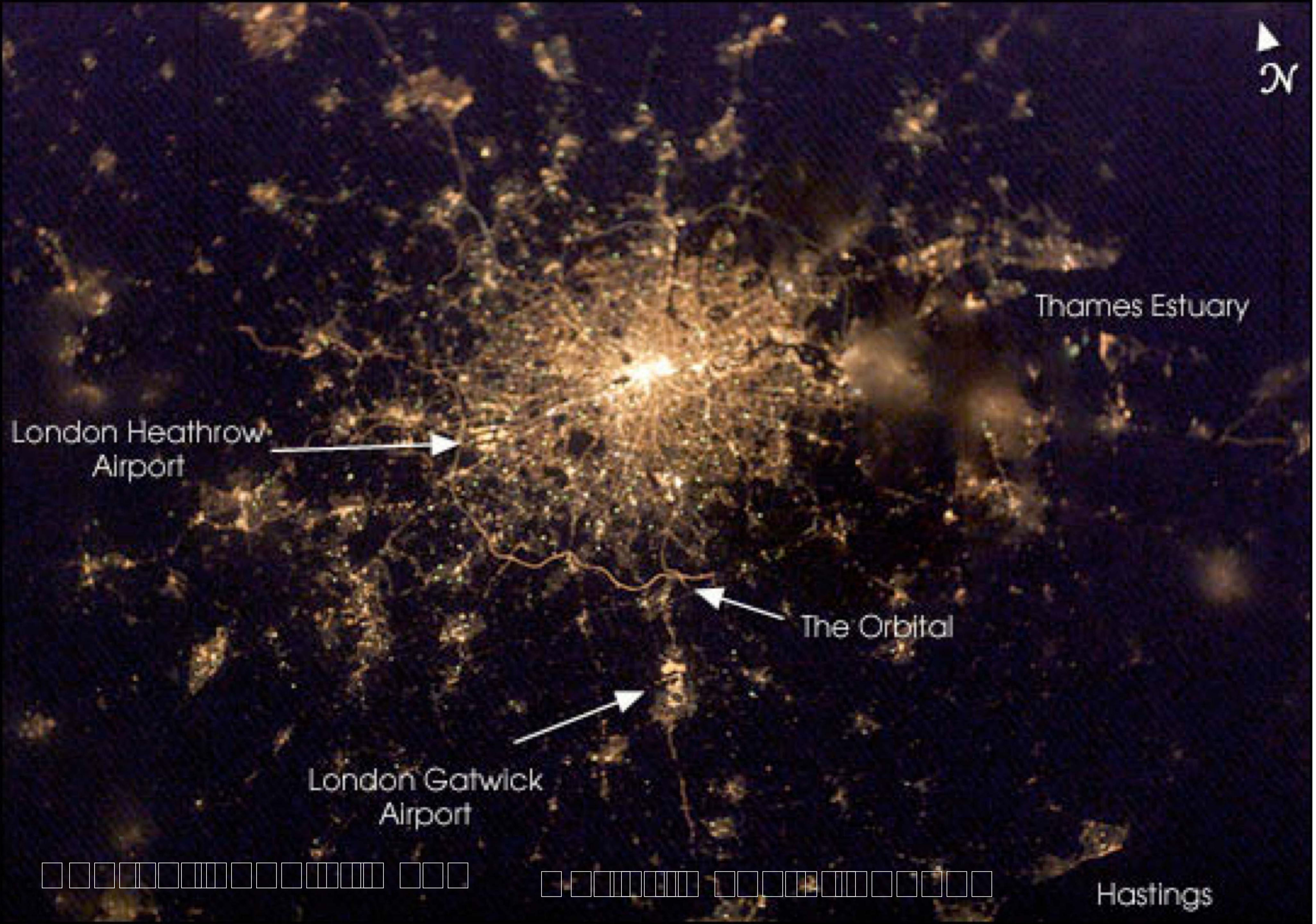
Hands-on experience working within projects using Python (Pandas, SciPy, Seaborn, NumPy, etc.), basic-level R, SQL, and Scheme. Produced actionable insights and recommendations based on data analysis. Lead through collaborative communication with team members at various levels of organizations.

CE88: Data Science for Urban Systems

Urban science seeks to understand the fundamental processes that drive, shape and sustain cities and urbanization.

It is a multi/transdisciplinary approach involving concepts, methods and research from the social, natural, engineering and computational sciences, along with the humanities.

source: Report for NSF: Urban Science: Integrated Theory from the First Cities to Sustainable Metropolises


Thames Estuary
London Heathrow
Airport
The Orbital
London Gatwick
Airport
Hastings

Note on the report:

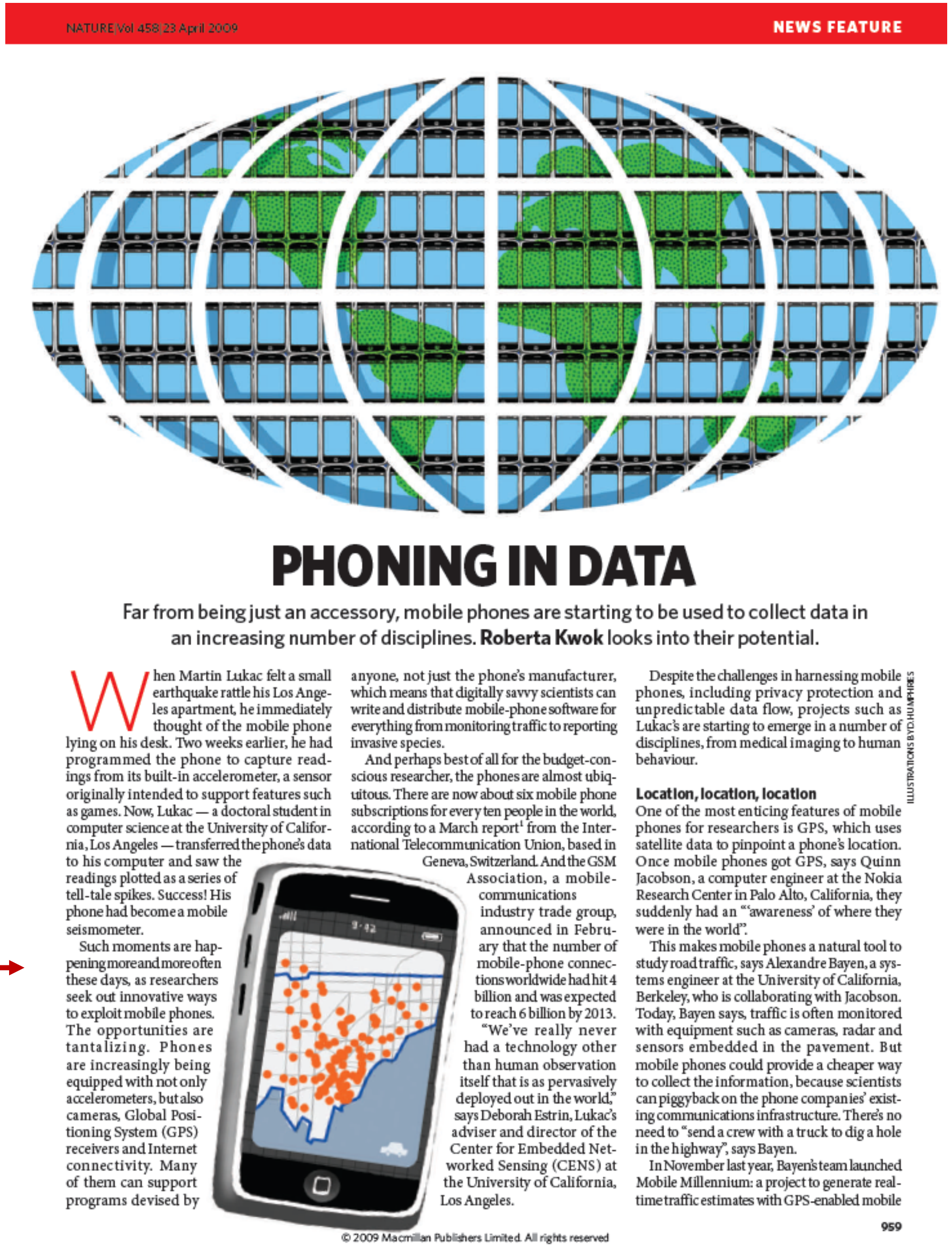
Urban analytics is a collection of tools used to analyze and map “urban big data” (generated by social media, crowd sourcing an sensor networks)

Note from the Report:

Technological and social developments have combined (intentionally but also as unintended consequences) to generate unprecedented amounts of data concerning what people (as individuals and in organizations) do when they agglomerate in cities.

A Trillion Points of Data

How tracking cell-phone users via GPS could do for the Real world what google did for the virtual world.



PHONING IN DATA

Far from being just an accessory, mobile phones are starting to be used to collect data in an increasing number of disciplines. **Roberta Kwok** looks into their potential.

When Martin Lukac felt a small earthquake rattle his Los Angeles apartment, he immediately thought of the mobile phone lying on his desk. Two weeks earlier, he had programmed the phone to capture readings from its built-in accelerometer, a sensor originally intended to support features such as games. Now, Lukac — a doctoral student in computer science at the University of California, Los Angeles — transferred the phone's data to his computer and saw the readings plotted as a series of tell-tale spikes. Success! His phone had become a mobile seismometer.

Such moments are happening more and more often these days, as researchers seek out innovative ways to exploit mobile phones. The opportunities are tantalizing. Phones are increasingly being equipped with not only accelerometers, but also cameras, Global Positioning System (GPS) receivers and Internet connectivity. Many of them can support programs devised by anyone, not just the phone's manufacturer, which means that digitally savvy scientists can write and distribute mobile-phone software for everything from monitoring traffic to reporting invasive species.

And perhaps best of all for the budget-conscious researcher, the phones are almost ubiquitous. There are now about six mobile phone subscriptions for every ten people in the world, according to a March report[1] from the International Telecommunication Union, based in Geneva, Switzerland. And the GSM Association, a mobile-communications industry trade group, announced in February that the number of mobile-phone connections worldwide had hit 4 billion and was expected to reach 6 billion by 2013.

"We've really never had a technology other than human observation itself that is as pervasively deployed out in the world," says Deborah Estrin, Lukac's adviser and director of the Center for Embedded Networked Sensing (CENS) at the University of California, Los Angeles.

Despite the challenges in harnessing mobile phones, including privacy protection and unpredictable data flow, projects such as Lukac's are starting to emerge in a number of disciplines, from medical imaging to human behaviour.

Location, location, location

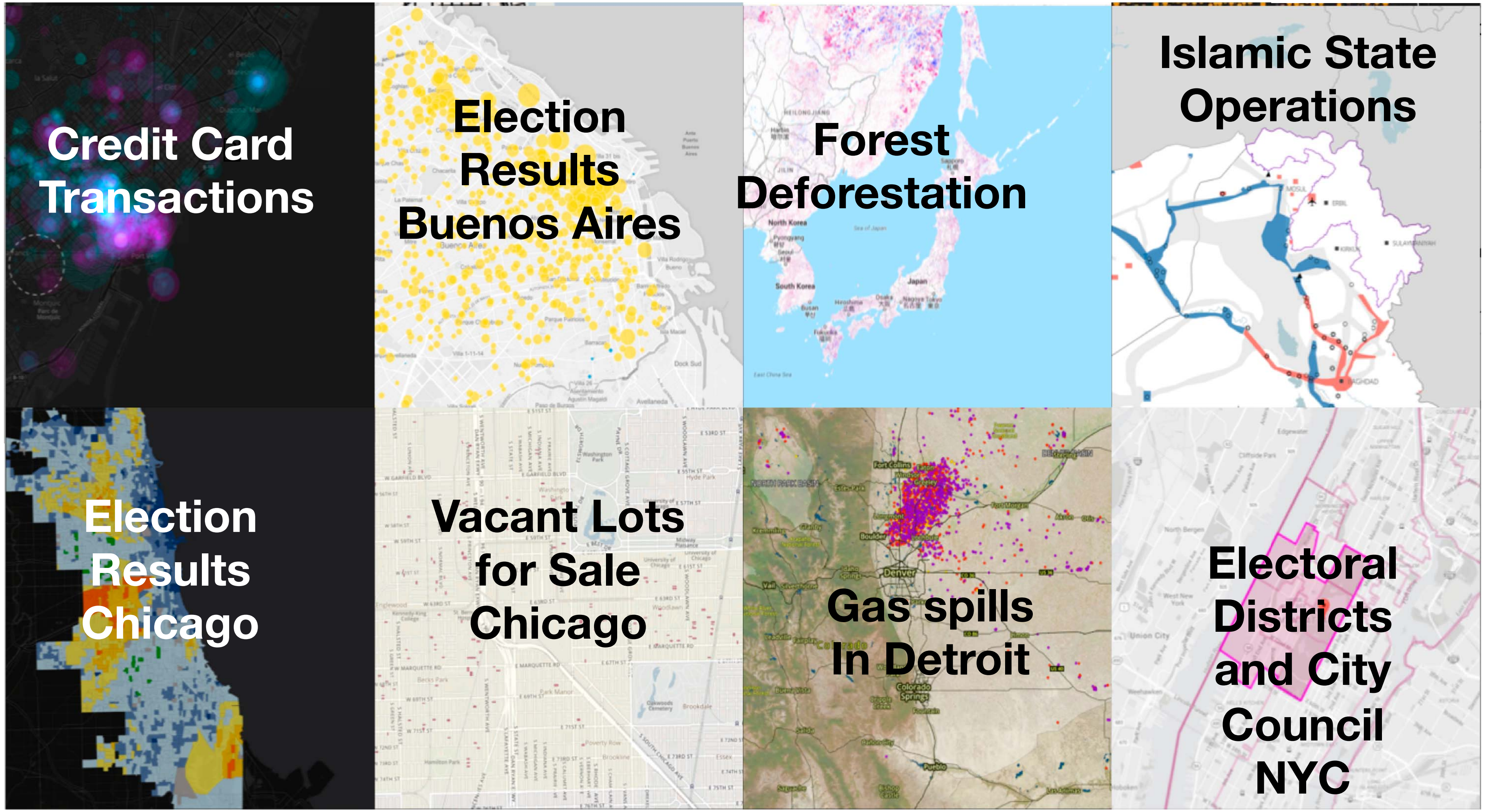
One of the most enticing features of mobile phones for researchers is GPS, which uses satellite data to pinpoint a phone's location. Once mobile phones got GPS, says Quinn Jacobson, a computer engineer at the Nokia Research Center in Palo Alto, California, they suddenly had an "'awareness' of where they were in the world".

This makes mobile phones a natural tool to study road traffic, says Alexandre Bayen, a systems engineer at the University of California, Berkeley, who is collaborating with Jacobson. Today, Bayen says, traffic is often monitored with equipment such as cameras, radar and sensors embedded in the pavement. But mobile phones could provide a cheaper way to collect the information, because scientists can piggyback on the phone companies' existing communications infrastructure. There's no need to "send a crew with a truck to dig a hole in the highway", says Bayen.

In November last year, Bayen's team launched Mobile Millennium: a project to generate real-time traffic estimates with GPS-enabled mobile

ILLUSTRATIONS BY D. HUMPHRIES

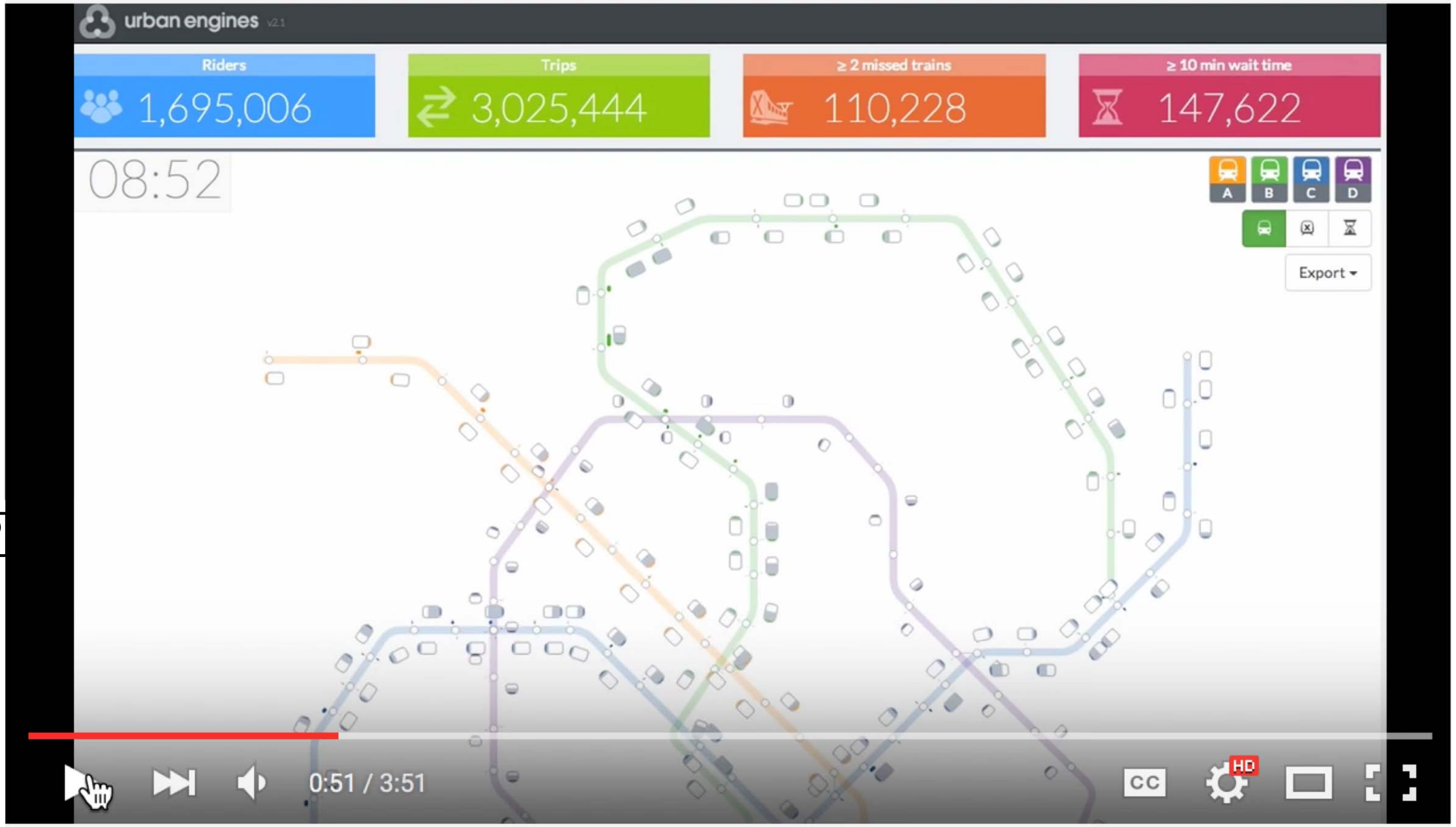
https://www.nature.com/articles/458959a


Credit Card Transactions
Election Results Buenos Aires
Forest Deforestation
Islamic State Operations
Election Results Chicago
Vacant Lots for Sale Chicago
Gas spills In Detroit
Electoral Districts and City Council NYC

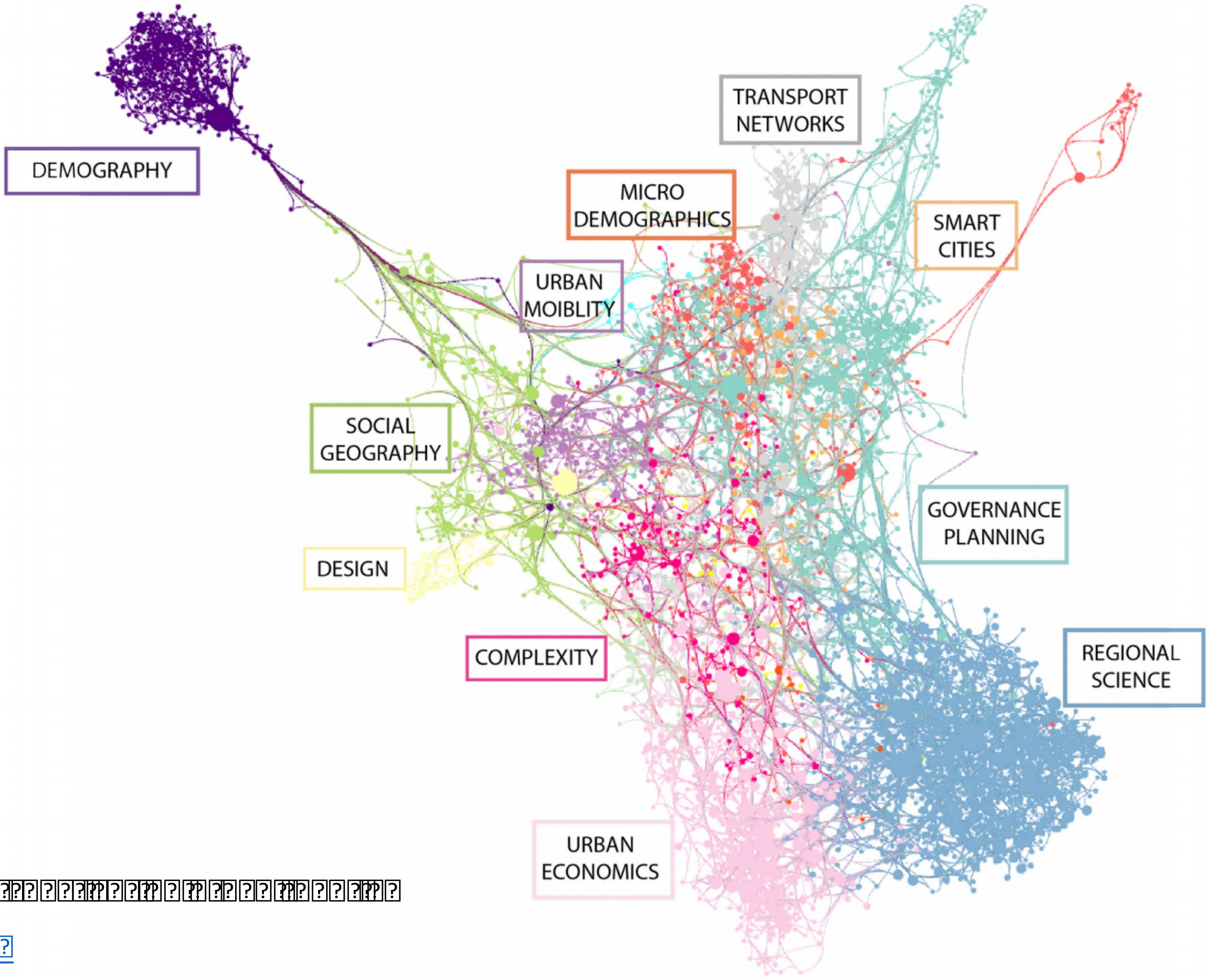
[illegible]

[illegible]

[illegible]


urban engines
Riders
1,695,006
Trips
3,025,444
≥ 2 missed trains
110,228
≥ 10 min wait time
147,622
08:52
A
B
C
D
Export
0:51 / 3:51
CC
HD

[illegible]



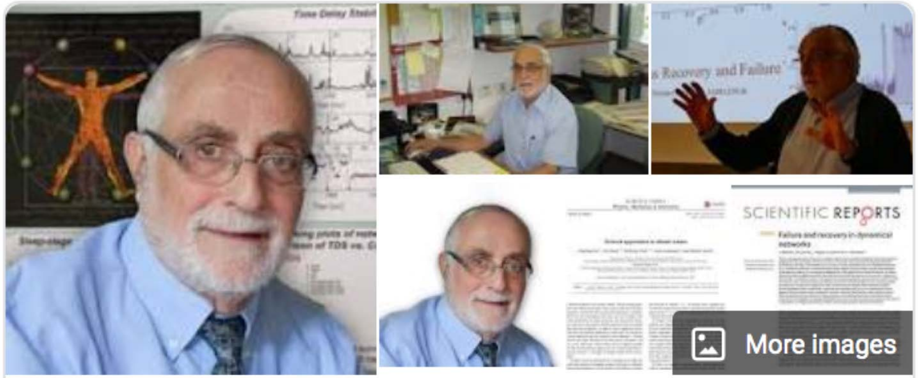
DEMOGRAPHY
TRANSPORT NETWORKS
MICRO DEMOGRAPHICS
SMART CITIES
URBAN MOIBLITY
SOCIAL GEOGRAPHY
GOVERNANCE PLANNING
DESIGN
COMPLEXITY
REGIONAL SCIENCE
URBAN ECONOMICS

Some History

Jane Jacobs as chairperson of a Greenwich Village civic group at a 1961 press conference

Jacobs fought to prevent Washington Square Park, pictured, from being demolished for a highway

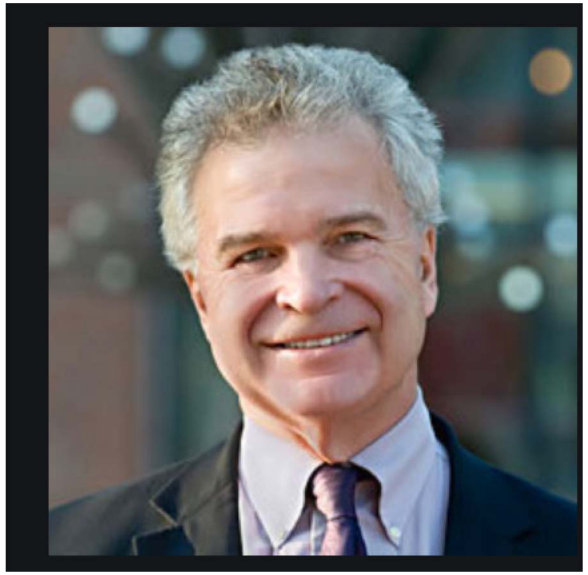

THE NEW SCIENCE OF CITIES
INVENTING FUTURE CITIES
Urban Modelling
The New Science of Cities
2013
Cities and Complexity
2005
Inventing Future Cities
2018
Fractal Cities: A Geometr...
1994
Urban Modelling: Algorith...
1976



Shlomo Havlin

Professor

Shlomo Havlin is a Professor in the Department of Physics at Bar-Ilan University, Ramat-Gan, Israel. He served as President of the Israel Physical Society, Dean of Faculty of Exact Sciences, Chairman, Department of Physics. In 2018 he won the Israel Prize for his accomplishments in physics. Wikipedia



H. Eugene Stanley

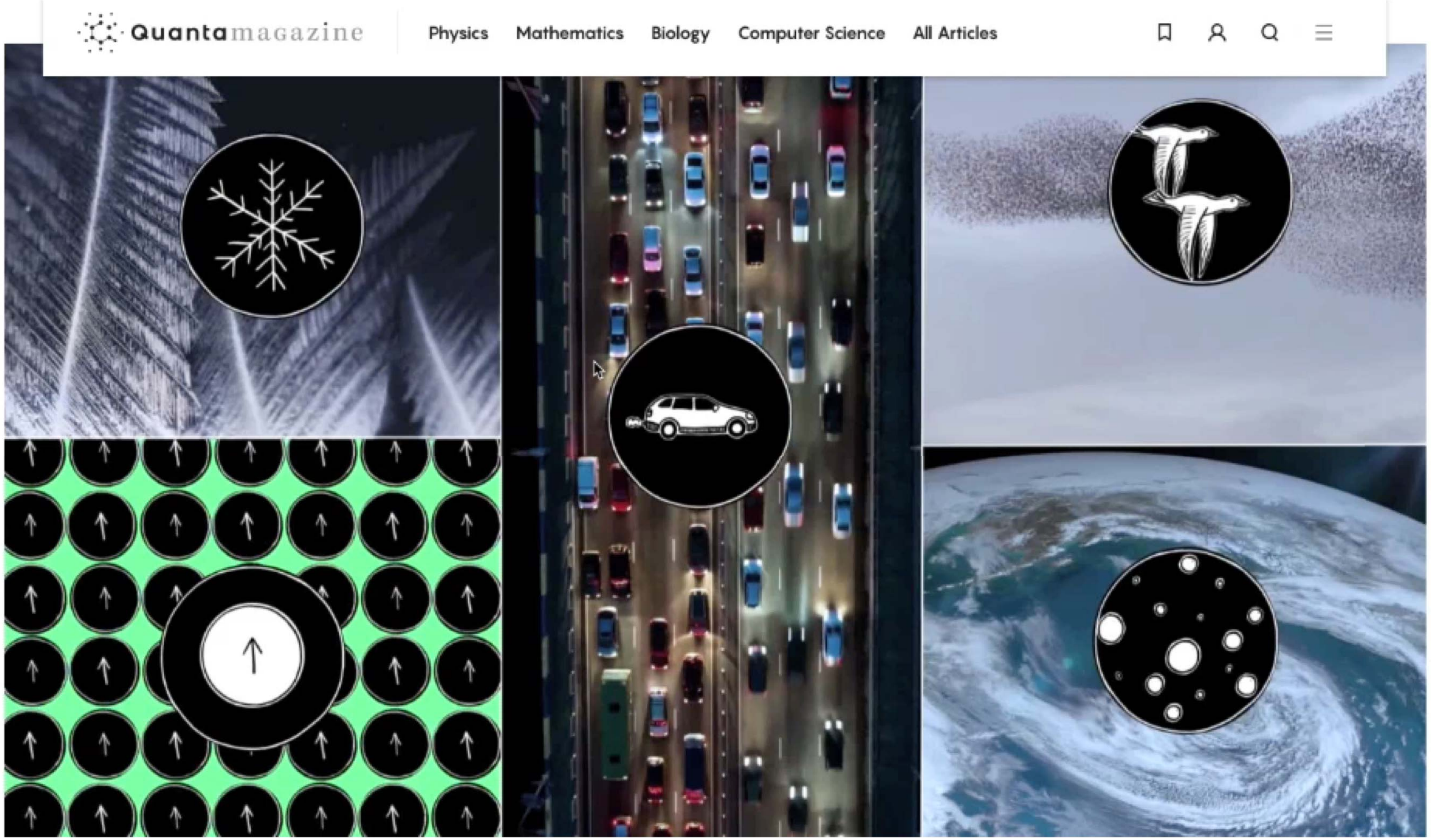
From Wikipedia, the free encyclopedia

Harry Eugene Stanley (born March 28, 1941) is an American physicist and University Professor at Boston University. He has made seminal contributions to statistical physics and is one of the pioneers of interdisciplinary science. His current research focuses on understanding the anomalous behavior of liquid water, but he had made fundamental contributions to complex systems, such as quantifying correlations among the constituents of the Alzheimer brain, and quantifying fluctuations in noncoding and coding DNA sequences, interbeat intervals of the healthy and diseased heart. He is one of the founding fathers of econophysics.

Complexity science, also called complex systems science, studies how a large collection of components – locally interacting with each other at small scales – can spontaneously self-organize to exhibit non-trivial global structures and behaviors at larger scales, often without external intervention, central authorities or leaders.

Examples: billions of interacting neurons in the human brain; computers communicating in the Internet; humans in multifaceted relationships.

Concepts: system, component, interactions, network, structure, heterogeneity, inter-relatedness, inter-connectedness, interdependence, subsystems, boundaries, environment, open/closed systems, systems of systems


Quanta magazine
Physics
Mathematics
Biology
Computer Science
All Articles

The core Method of CE88: Network Science in a framework of Complex Systems thinking to study urban systems.

Weeks 1-3: Network Measures

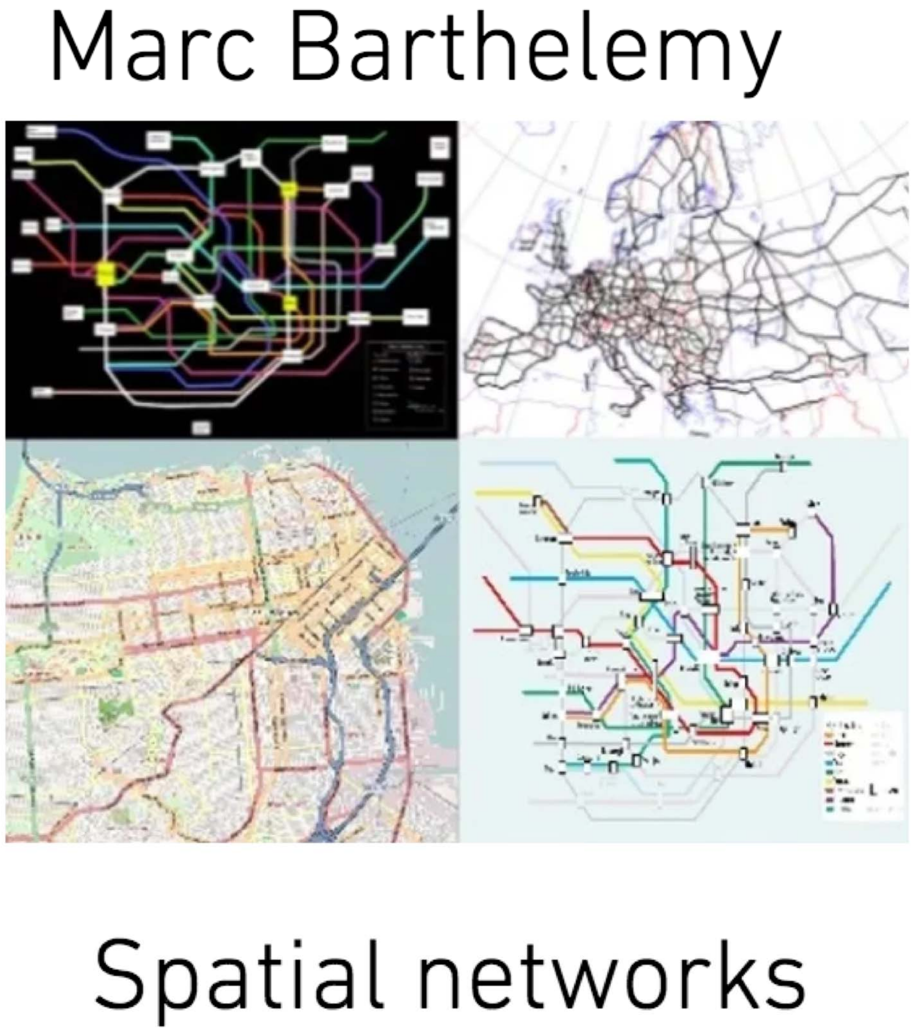
Weeks 3-6: Clusters and Centrality

Weeks 7-8: Tools Open Street Maps and Visualization

Mid Term: Project Proposal

Week 9-11: Methods and Papers based on projects:

Multigraphs, Robustness, Spreading Dynamics


Marc Barthelemy
Spatial networks

[illegible]

[illegible]

[illegible]

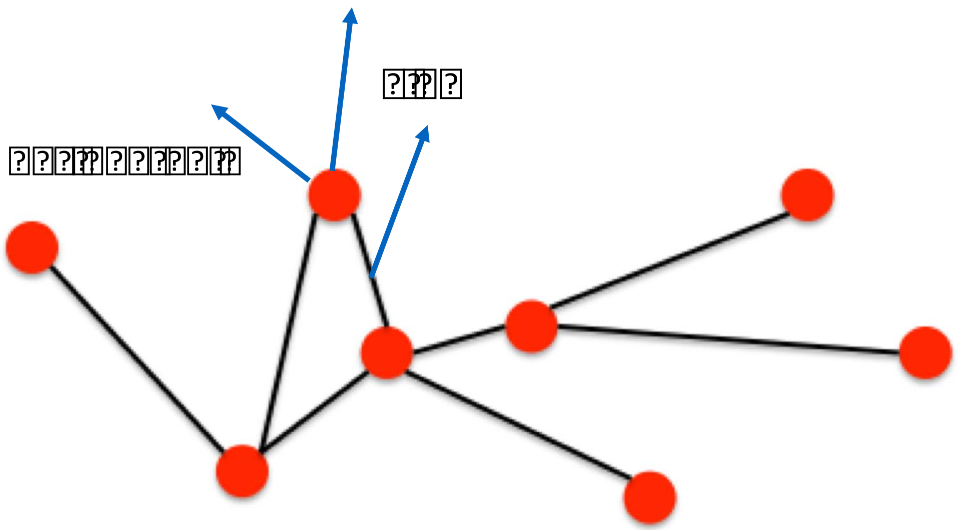
[illegible]

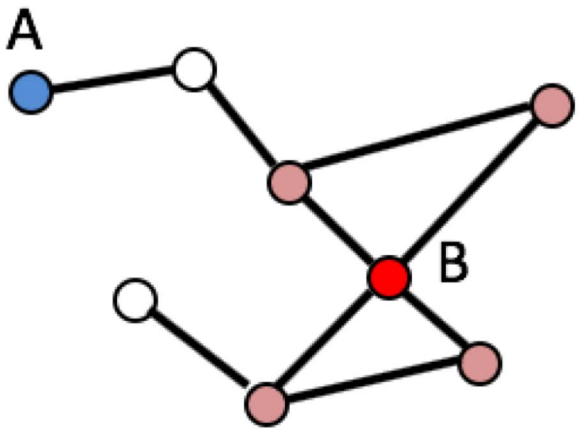
[illegible]

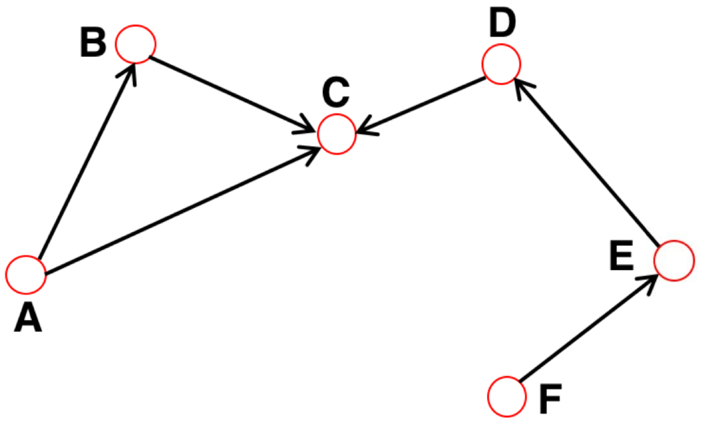
Table 1. Data sources according to their typical timescale and some phenomena occurring on these timescales.

Timescale	Data sources	Example phenomena
Minutes–days	• GPS • Mobile phones (private companies) • RFID (transport companies such as RATP in Paris or TfL in London)	Spatial structure, mobility, urban activity
Months–years	• Surveys • Censuses (US census bureau [109] or Eurostat [110]) • City administrations	Social organization, housing market
Decades–centuries	• Historical documents and maps (NYPL [11] or the geohistoricaldata research group [12])	Urban growth, self-organization, impact of planning

Comments in parentheses are example data providers.




A
B


A
B
C
D
E
F

Network Measures:

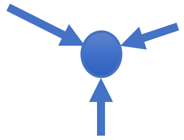
- Degree
- Degree Distribution
- Shortest path length
- Average path length

Node network properties

from immediate connections

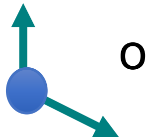
•indegree

how many directed edges (arcs) are incident on a node



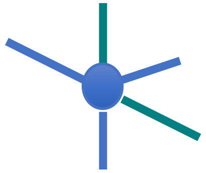
•outdegree

how many directed edges (arcs) originate at a node



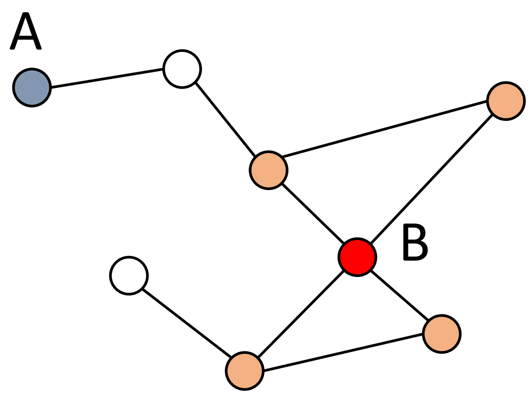
•degree (in or out)

number of edges incident on a node



NODE DEGREES

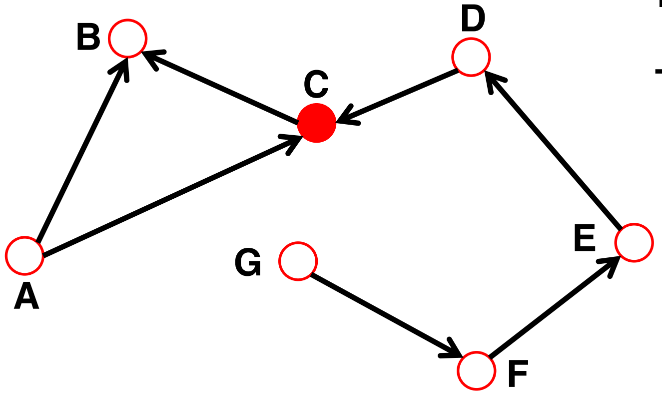
Undirected



Node degree: the number of links connected to the node.

$$k_A = 1 \qquad k_B = 4$$

Directed



In *directed networks* we can define an in-degree and out-degree.

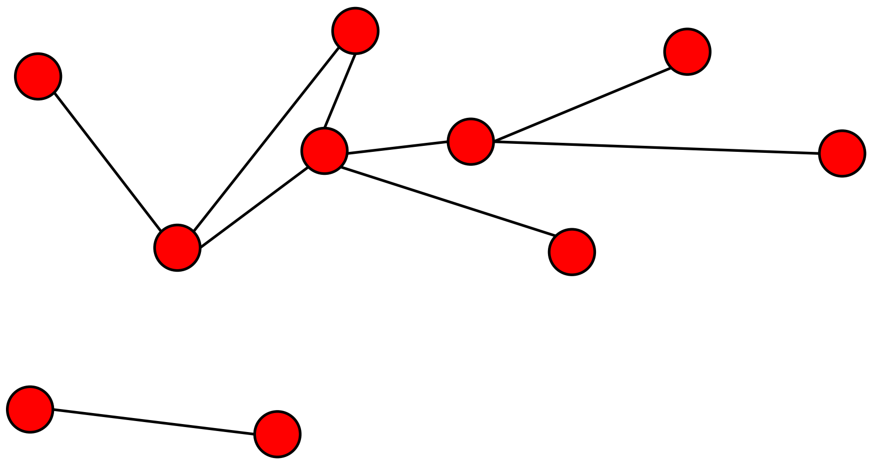
The (total) degree is the sum of in- and out-degree.

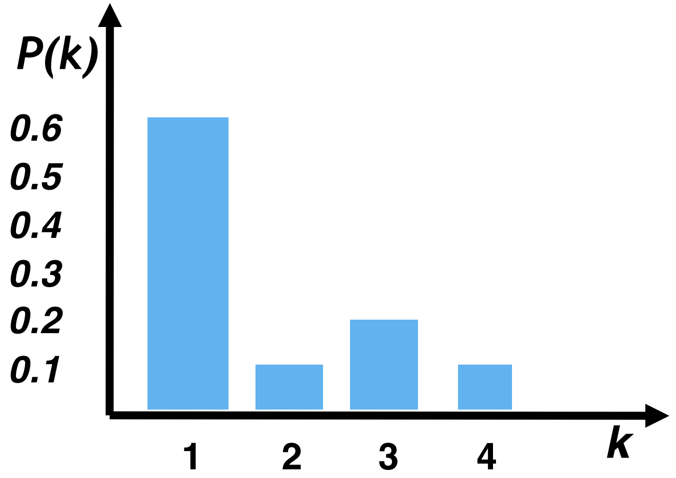
$$k_C^{in} = 2 \quad k_C^{out} = 1 \qquad k_C = 3$$

Degree distribution P(k): probability that a randomly chosen vertex has degree k

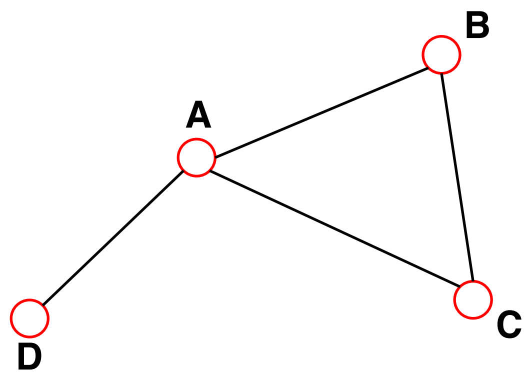
N_k = # nodes with degree k

$P(k) = N_k / N$ ⇨ plot



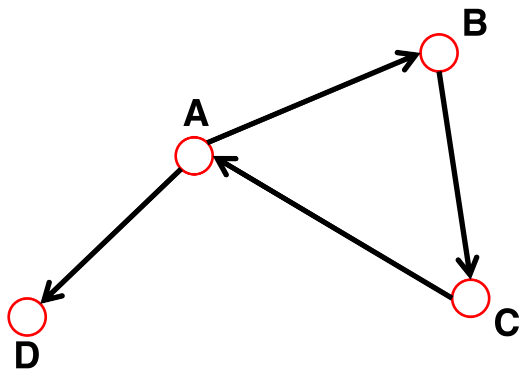


DISTANCE IN A GRAPH



The *distance (shortest path, geodesic path)* between two nodes is defined as the number of edges along the shortest path connecting them.

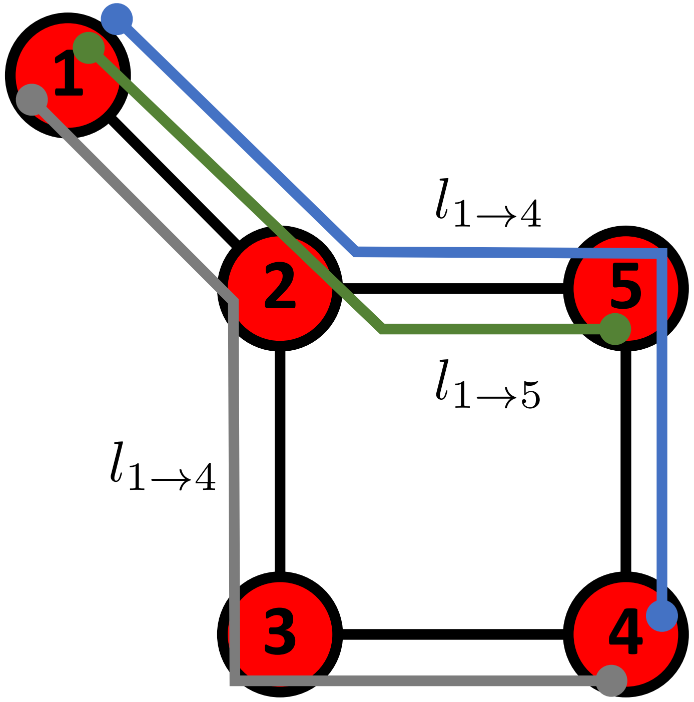
*If the two nodes are disconnected, the distance is infinity.



In directed graphs each path needs to follow the direction of the arrows.

Thus in a digraph the distance from node A to B (on an AB path) is generally different from the distance from node B to A (on a BCA path).

Shortest Path



$l_{1\to4} = 3$

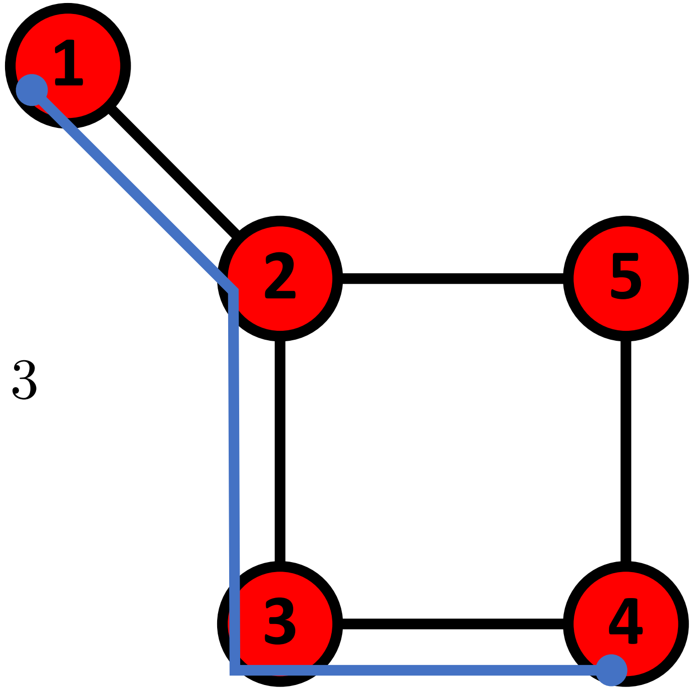
$l_{1\to5} = 2$

The path with the shortest length between two nodes (distance).

Average Path Length

$$(l_{1\to 2} + l_{1\to 3} + l_{1\to 4} + \\ + l_{1\to 5} + l_{2\to 3} + l_{2\to 4} + \\ + l_{2\to 5} + l_{3\to 4} + l_{3\to 5} + \\ + l_{4\to 5})\,/10 = 1.6$$

$l_{1\to 4} = 3$



The average of the shortest paths for all pairs of nodes.

Numbers of pairs for a network of size N is N(N-1)/2 `average_shortest_path_length`(*G, weight=None*) [source]

Return the average shortest path length.

The average shortest path length is

$$a = \sum_{s,t \in V} \frac{d(s, t)}{n(n-1)}$$

where `V` is the set of nodes in `G`, `d(s, t)` is the shortest path from `s` to `t`, and `n` is the number of nodes in `G`.

shortest_path(*G, source=None, target=None, weight=None*) [source]

Compute shortest paths in the graph.

Parameters:

- **G** (*NetworkX graph*)
- **source** (*node, optional*) – Starting node for path. If not specified, compute shortest paths for each possible starting node.
- **target** (*node, optional*) – Ending node for path. If not specified, compute shortest paths to all possible nodes.
- **weight** (*None or string, optional (default = None)*) – If None, every edge has weight/distance/cost 1. If a string, use this edge attribute as the edge weight. Any edge attribute not present defaults to 1.

Returns:

path – All returned paths include both the source and target in the path.

`shortest_path_length`(*G*, *source=None*, *target=None*, *weight=None*) [source]

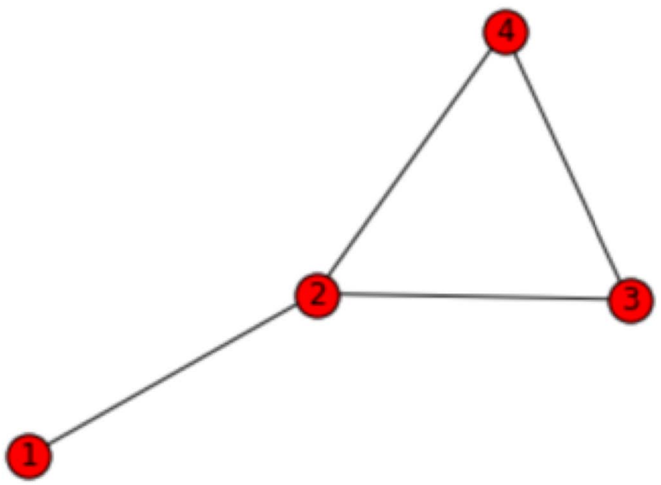
Compute shortest path lengths in the graph.

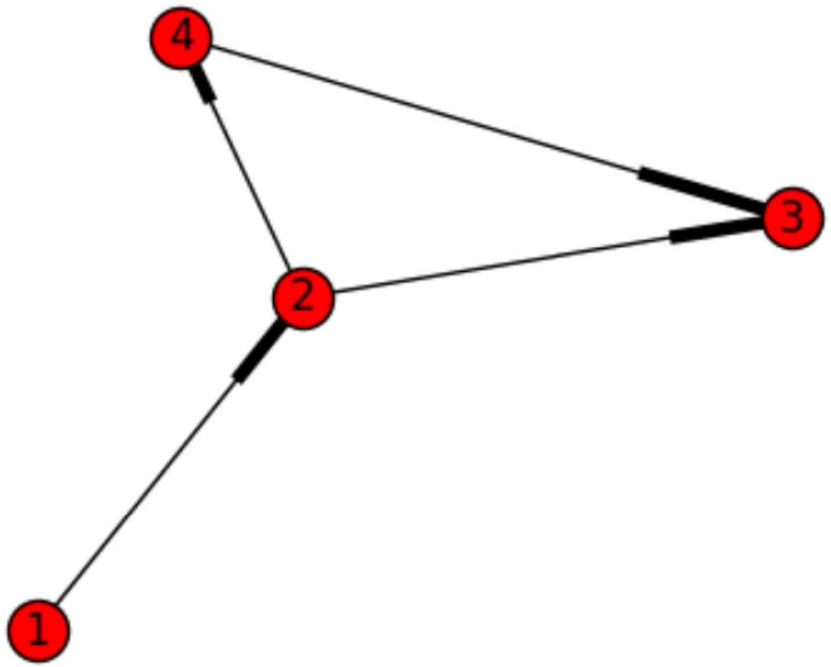
Parameters:

- **G** (*NetworkX graph*)
- **source** (*node, optional*) – Starting node for path. If not specified, compute shortest path lengths using all nodes as source nodes.
- **target** (*node, optional*) – Ending node for path. If not specified, compute shortest path lengths using all nodes as target nodes.
- **weight** (*None or string, optional (default = None)*) – If None, every edge has weight/distance/cost 1. If a string, use this edge attribute as the edge weight. Any edge attribute not present defaults to 1.

Returns:

length – If the source and target are both specified, return the length of the shortest path from the source to the target.


4
2
3
1


4
3
2
1