Inverse Problems in Evolving Graphs

Andrew Fraser

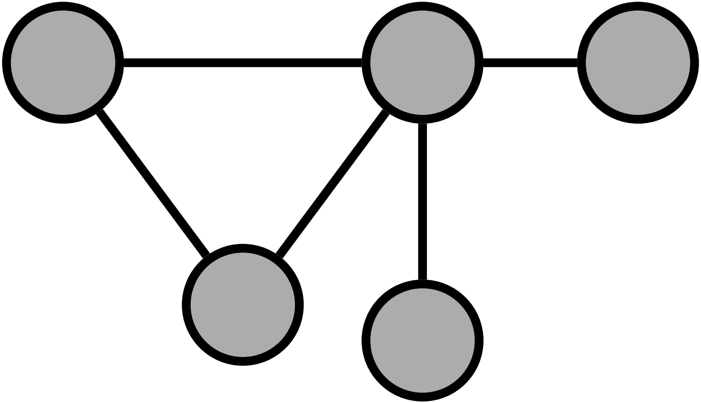
6 December 2021

Outline

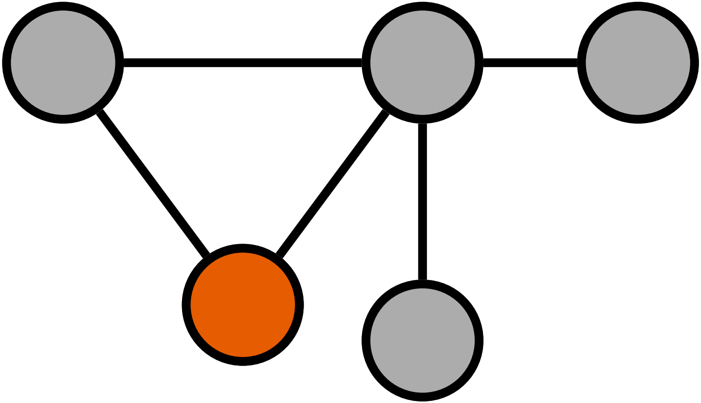
1. Introduction to evolving graphs
2. Introduction to the inverse problem[1]
3. My approach and results

[1]Grindrod Peter and Higham Desmond J. 2010, Evolving graphs: dynamical models, inverse problems and propagation

Graphs

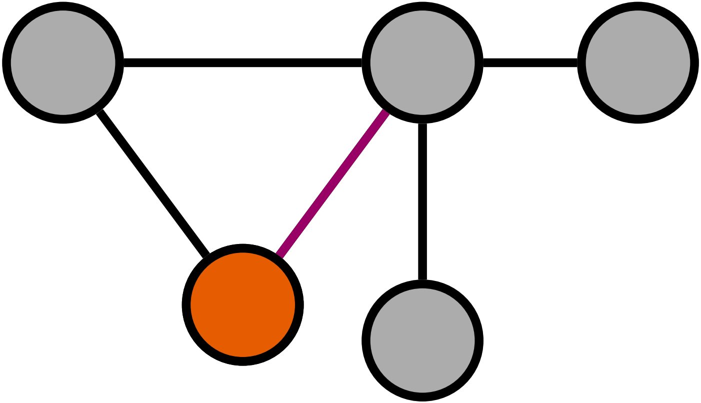


Graphs



Vertex: Someone/something with connections.

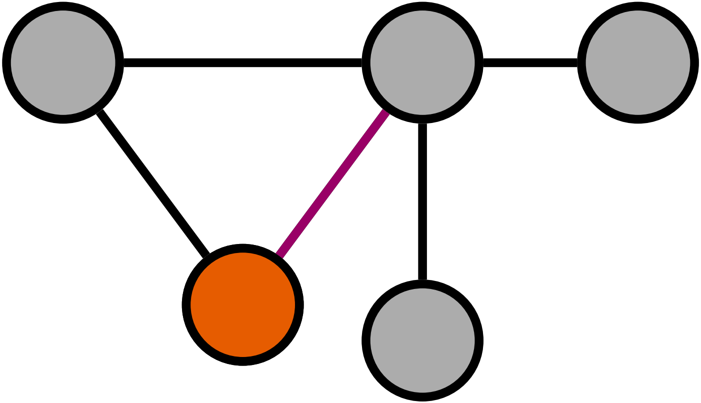
Graphs



Vertex: Someone/something with connections.

Edge: A connection between two vertices.

Graphs

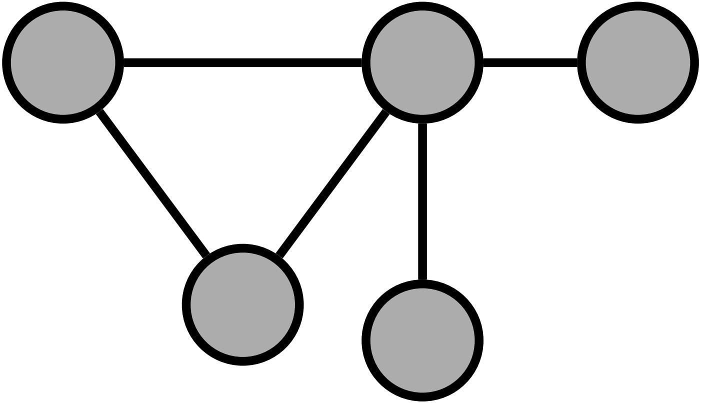


Vertex: Someone/something with connections.

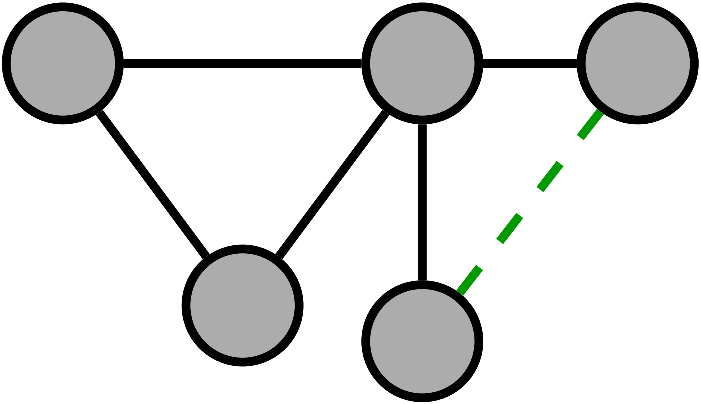
Edge: A connection between two vertices.

Examples: Social media, airport network, the internet

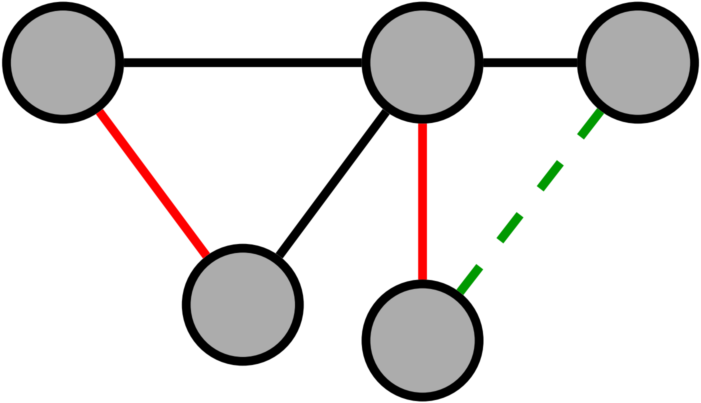
Evolving Graphs



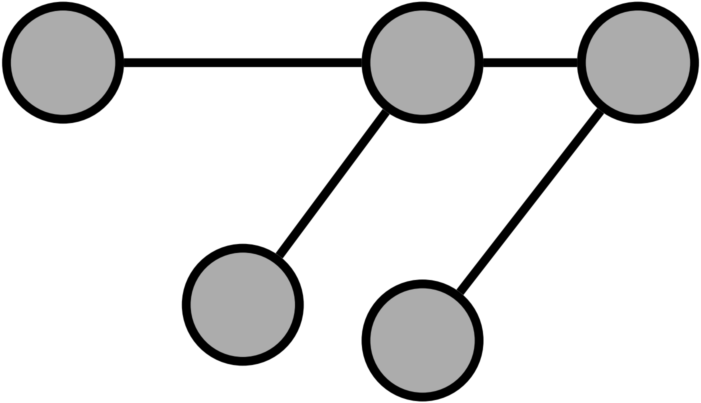
Evolving Graphs



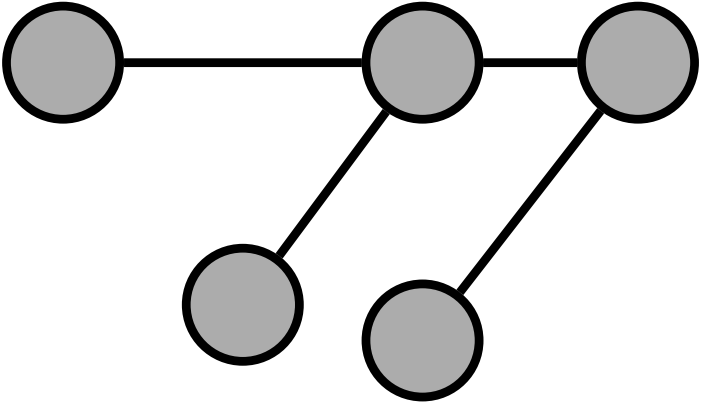
Evolving Graphs



Evolving Graphs



Evolving Graphs



We assign each edge two probabilities:

Birth Rate: $\hat{\alpha}(e)$, the probability that the edge will appear.

Death Rate: $\hat{\omega}(e)$, the probability that the edge will disappear.

Range-Dependent Graphs

Suppose some assignment of the integers 1...n

i_v = the integer assigned to vertex v

Range-Dependent Graphs

Suppose some assignment of the integers 1...n
i_v = the integer assigned to vertex v

Range: An edge (u, v) has range $|i_u - i_v|$

Range-Dependent Graphs

Suppose some assignment of the integers 1...n
i_v = the integer assigned to vertex v

Range: An edge (u, v) has range $|i_u - i_v|$

We can define birth rates and death rates as a function of range:
$\hat{\alpha}(e) = \alpha(|i_u - i_v|)$

$\hat{\omega}(e) = \omega(|i_u - i_v|)$

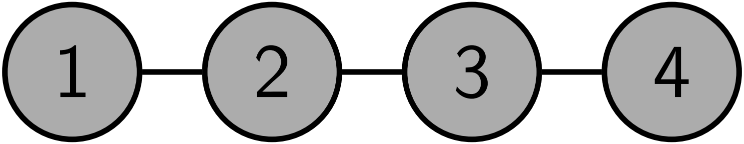
Range-Dependent Graphs

Suppose some assignment of the integers 1...n
i_v = the integer assigned to vertex v

Range: An edge (u, v) has range $|i_u - i_v|$

We can define birth rates and death rates as a function of range:
$\hat{\alpha}(e) = \alpha(|i_u - i_v|)$

$\hat{\omega}(e) = \omega(|i_u - i_v|)$



Range of (v1, v4) $= |1 - 4| = 3$
Range of (v2, v3) $= |2 - 3| = 1$

Problem to Solve

Forward Problem: Given a graph, $\hat{\alpha}$, and $\hat{\omega}$: compute states of the graph for $i = 1, 2, \ldots J$

Inverse Problem: Given a graph and J states of the graph: compute $\hat{\alpha}$ and $\hat{\omega}$.

Defining the Inverse Problem

EVOLVING GRAPH MODEL

Input: A graph G, J states of the graph F, and a matrix R containing the frequencies of each edge in F

Question: What mapping on the integers q minimizes $\sum_{v_1, v_2 \in G} R[v_1][v_2](q(v_1) - q(v_2))^2$?

Defining the Inverse Problem

Evolving Graph Model

Input:	A graph G, J states of the graph F, and a matrix R containing the frequencies of each edge in F
Question:	What mapping on the integers q minimizes $\sum_{v_1, v_2 \in G} R[v_1][v_2](q(v_1) - q(v_2))^2$?

Problem: Since q is not differentiable, this problem is intractable.

Solution: We relax q to be a vector instead of a function

Defining the Inverse Problem

Evolving Graph Model	
Input:	A graph G, J states of the graph F, and a matrix R containing the frequencies of each edge in F
Question:	What mapping on the integers q minimizes $\sum_{v_1, v_2 \in G} R[v_1][v_2](q(v_1) - q(v_2))^2$?

Problem: Since q is not differentiable, this problem is intractable.
Solution: We relax q to be a vector instead of a function

This leads to the form: $q^T \Delta_R q$.
Note: Δ_R is the Laplacian matrix of R

Defining the Inverse Problem

EVOLVING GRAPH MODEL	
Input:	A graph G, J states of the graph F, and a matrix R containing the frequencies of each edge in F
Question:	What mapping on the integers q minimizes $\sum_{v_1, v_2 \in G} R[v_1][v_2](q(v_1) - q(v_2))^2$?

Problem: Since q is not differentiable, this problem is intractable.
Solution: We relax q to be a vector instead of a function

This leads to the form: $q^T \Delta_R q$.
Note: Δ_R is the Laplacian matrix of R

Big problem: $q^T \Delta_R q$ is a nonlinear (quadratic) inverse problem.

Solving the Quadratic Inverse Problem

Problem to solve: $q^T \Delta_R q$

Solving the Quadratic Inverse Problem

Problem to solve: $q^T \Delta_R q$

One solution: eigenanalysis by cases on the matrix Δ_R
Runtime: $O(n^3)$
By: Grindrod and Higham (2009)

Solving the Quadratic Inverse Problem

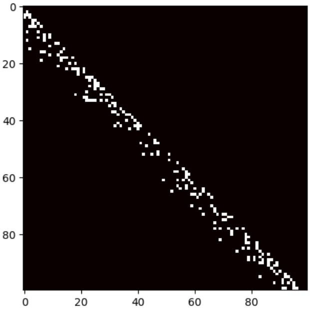
Problem to solve: $q^T \Delta_R q$

One solution: eigenanalysis by cases on the matrix Δ_R
Runtime: $O(n^3)$
By: Grindrod and Higham (2009)

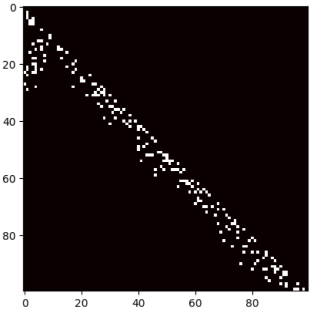
My solution: convert to optimization, perform gradient descent
Derivative: $q^T(A + A^T)$
Compute the derivative and take a step towards it, then repeat.
Runtime: $O(sn^2)$, where s is the number of steps to converge

Results: Model-Based Test

Original Mapping:

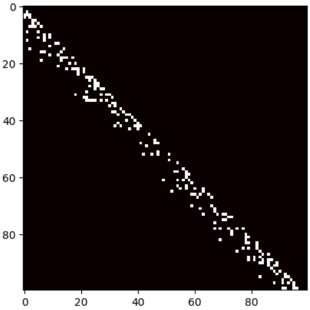


Gradient Descent:

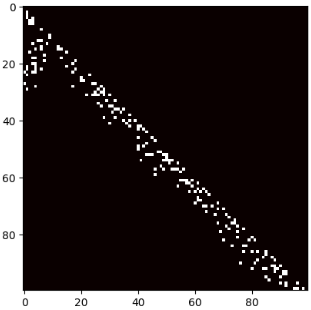


Results: Model-Based Test

Original Mapping:



Gradient Descent:

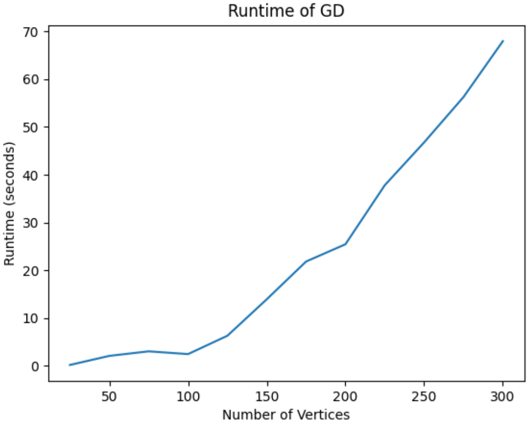


Original Error: ≈1.2M

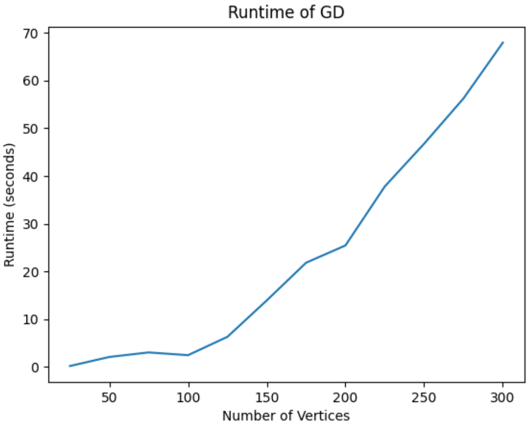
Gradient Descent Error: ≈1.6M

Results: On Real Data?

Results: On Real Data?



Results: On Real Data?



Most large internet models have millions of vertices and **tens of millions** of edges.
This is nowhere near feasible on large graphs!

Next Steps

- Investigate into applying fourier transforms
- Find smaller dataset applications
- Experiment with parameters of gradient descent

Thank you!

Questions?