Identifying Notable Objects from Spitzer Enhanced Imaging Astronomical Observations

IBhojwani, AlexanderTyan, Sun-Kev, tamos

June 1, 2018

CMSC 12300

The University of Chicago

Goals

- Identify and locate notable objects (outliers) within the sky
- Define an area of interest around that object
- Narrow down search area for manual analysis

Approach

Excess infrared light could mean:

- **Young Stellar Objects**
- Active Galactic Nucleii
- Colliding Galaxies

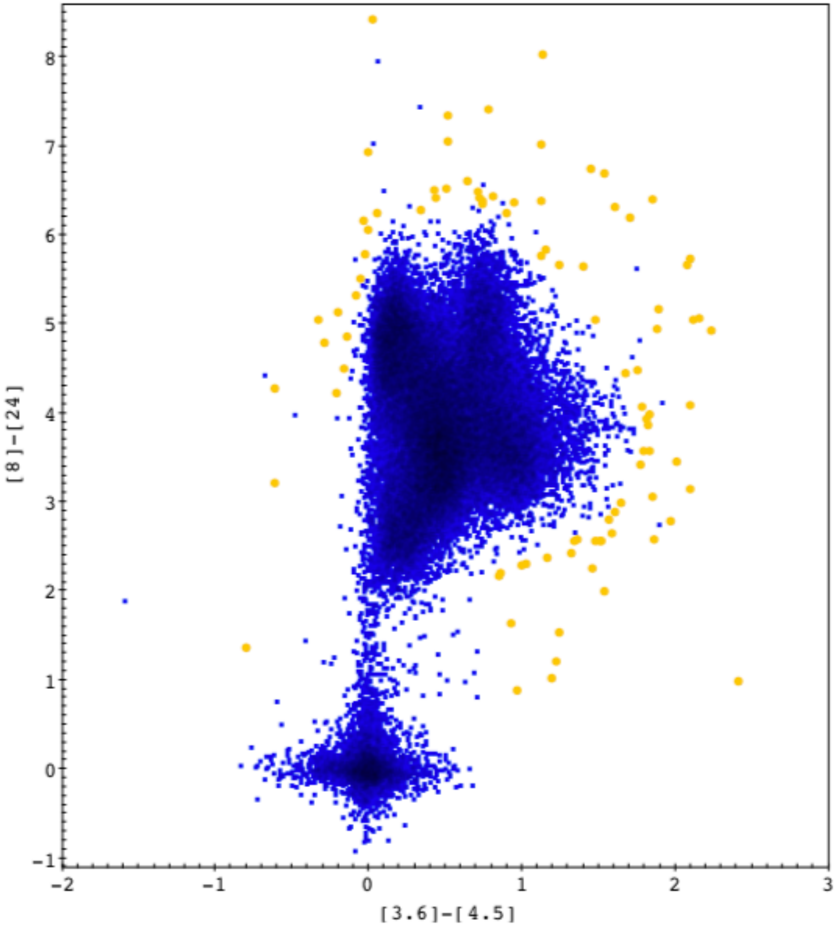
Hypotheses: Young Stellar Objects

- Notable objects can be identified as extrema in terms of infrared light
- Some notable objects are grouped into interesting structures

Data

- NASA/IPAC Infrared Science Archive:
  - Wide-field Infrared Survey Explorer (WISE)
    - Identifies objects, and readings on the energies they emit
  - 800m objects (records), 815 GB
  - Contains:
    - Location (right ascension, declination)
    - Movement
    - Colour
    - Readings across a number of bands

Unexpected Colour ➡ Interesting Object



Source: Gorjian et al.

FIGURE 4: Color-color plot of SEIP sources, SNR≥10, not in the galactic plane, and not in major survey areas (blue), vetted color outliers (yellow).

Algorithm I

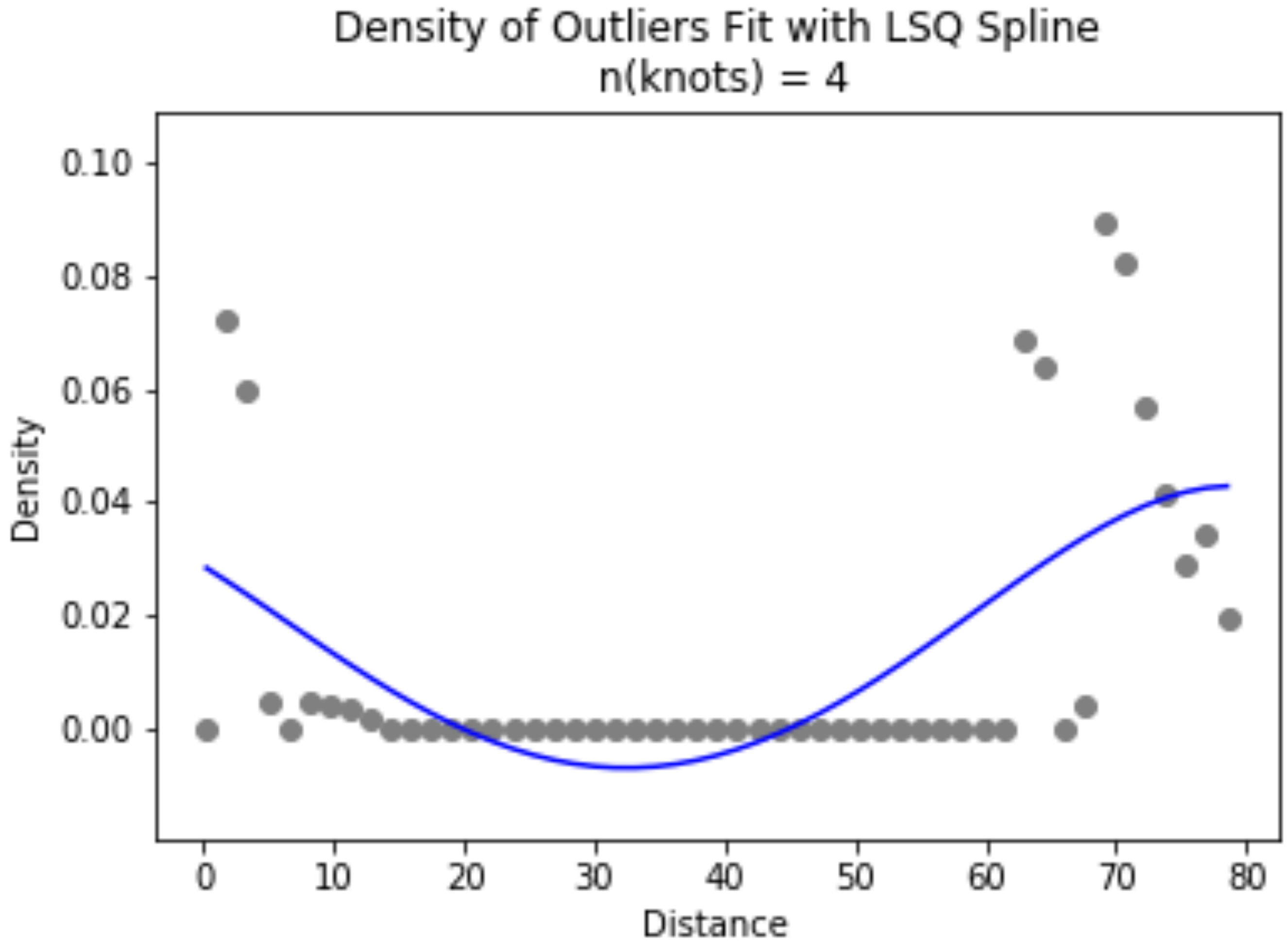
Algorithm I, Step 1: Preprocessing with K-Means Classification

- Canned K-Means (Dataproc Pending)
- Split points into two groups, ~ N within each

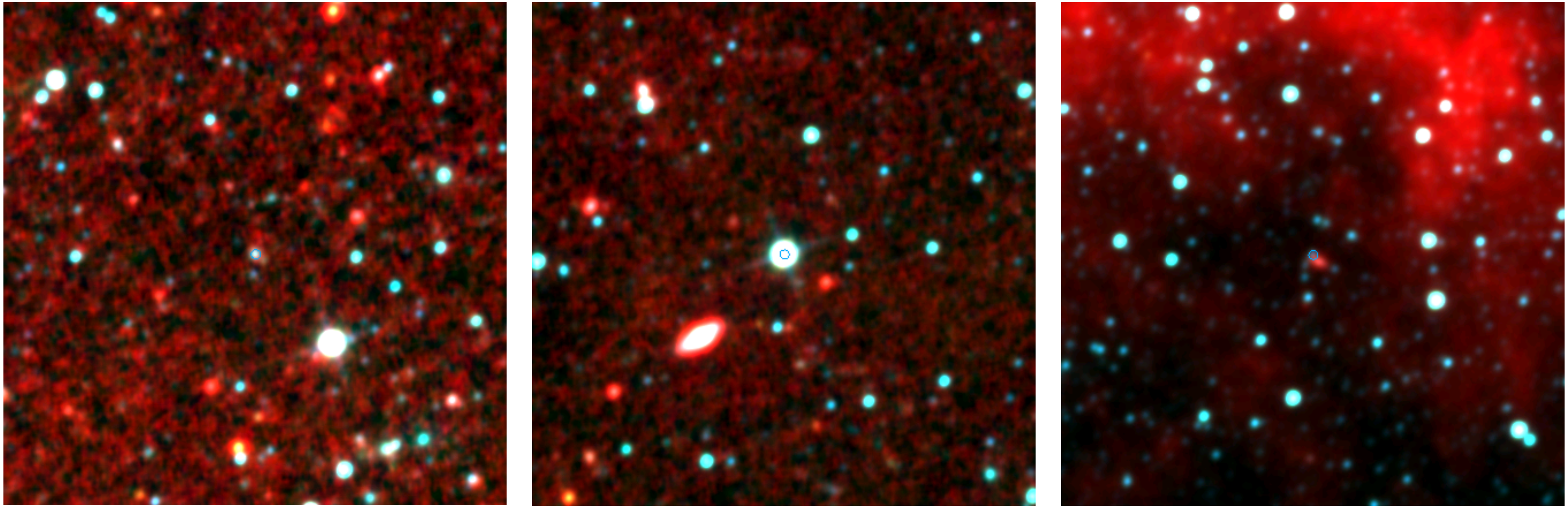
Algorithm I, Step 2: Outliers as Nodes, Distances as Edges

- MRJob
- Within each group, take outliers: x_i: $x_i \geq \mu \pm 2\sigma$
- Developed algorithm with parameters N, P, K.
- Complexity:
    - If number of cases < N(P): Does not run
    - Otherwise, approximately:
        - $(\sum_{i=1}^{(N-N*P)}(N*P+i)) * \frac{size-(N*P)}{(N*P)}$
    - Compare to:
        - $size^2$

Algorithm I: Distance Density with Fitted Spline



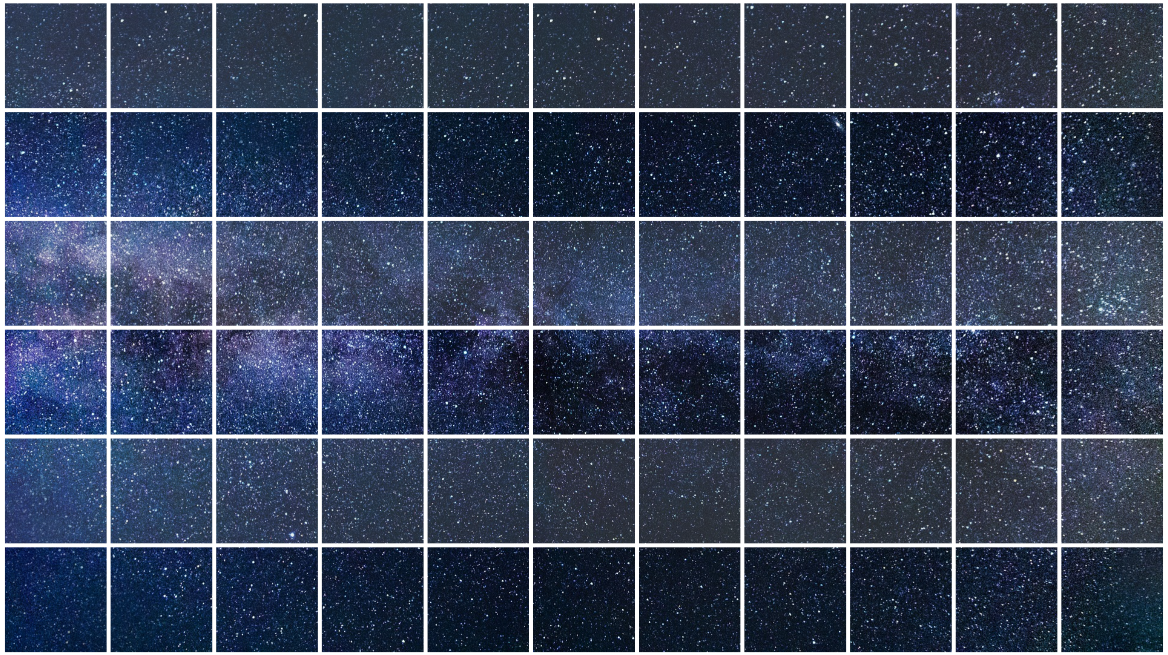
Algorithm I: Results



Algorithm II

Algorithm II (MapReduce)

- Step 1: Split Sky (Dataproc Implementation in Progress)
  - Create grids of the entire sky
  - Create graphs of each grid



Algorithm II (MapReduce)

- Step 1: Split Sky (Dataproc Implementation in Progress)
  - Create grids of the entire sky
  - Create graphs of each grid
- Step 2: Random Walk
- (Dataproc in Progress)
- Step 3: Find Probabilistic Clusters (Coding In Progress)

Runtime Comparison

Algorithm I

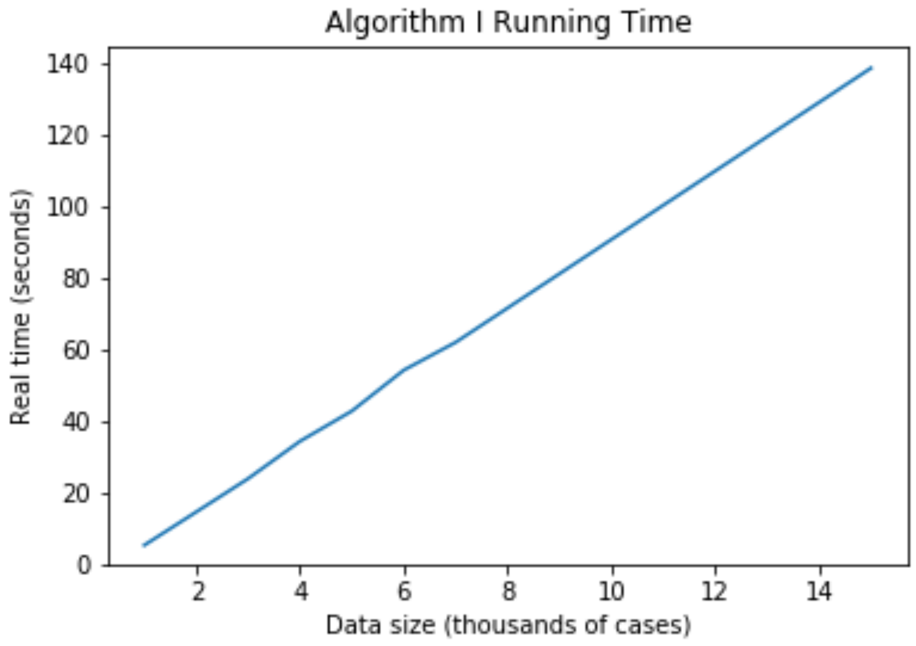
- Local:
  - > 85 min. 51 sec.
  - 365,601 rows
- Dataproc:
  - 8 min. 44 sec.
    - 25 workers with n1-standard-4 specs
    - 358,169 rows

Algorithm II (RandomWalk)

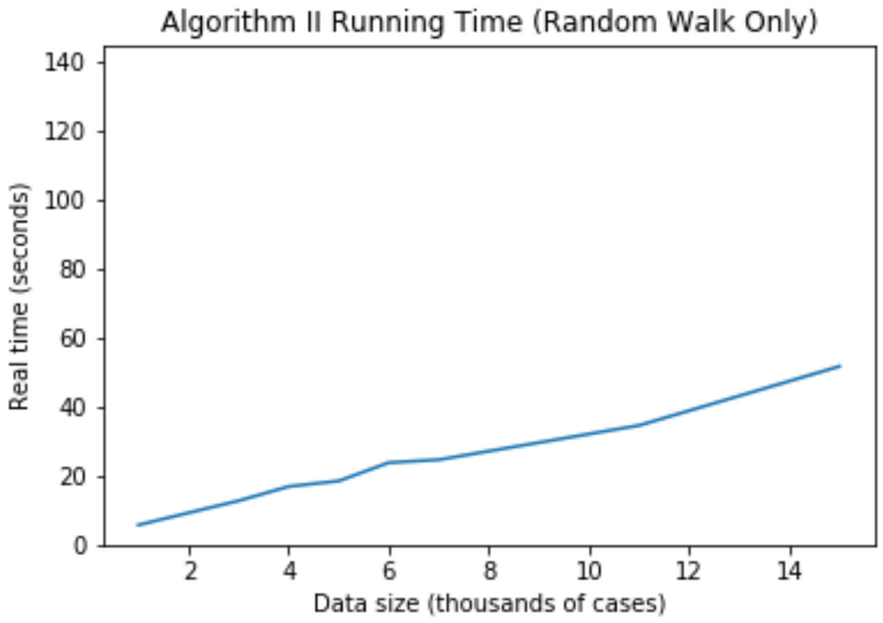
- Local:
  - 16 min. 9 sec.
  - 365,601 rows
- Dataproc:
  - In Progress

Runtime Comparison

Algorithm I



Algorithm II (RandomWalk)



Challenges

- **Problems:**
  - Graph algorithms without all data in memory (sky is too large)
  - Complexity too great for a fully connected graph
- **Solutions**:
  - Running random sample (Algorithm I)
  - Grids and streaming processing (Algorithm II)