Approximation of Decision problems

Michel de Rougemont
University Paris II
IRIF-CNRS
Plan

1. **Approximations**
   - Randomized algorithms
   - Interactive Proofs, PCPs
   - Testers

2. **Distribution of inputs**
   - Average complexity,
   - Specific distributions
   - Learning a distribution, machine learning

3. **Streaming algorithms for community detection in social graphs:** combining 1 + 2 (Joint work with Claire Mathieu)

Open problems
1. Approximations: a source of algorithms

- **Randomized algorithms**
  
  1970s
  
  If $x \in L$, then $\text{Prob} [A(x) \text{ accepts}] \geq \frac{2}{3}$
  
  If $\neg (x \in L)$, then $\text{Prob} [A(x) \text{ rejects}] \geq \frac{2}{3}$

- **Interactive Proofs, PCPs**
  
  1980-90s

- **Testers**
  
  2000s
  
  If $x \in L$, then $\text{Prob} [A(x) \text{ accepts}] = 1$
  
  If $x \in \text{far from } L$, then $\text{Prob} [A(x) \text{ rejects}] \geq \frac{2}{3}$
Bigdata (since 2010)

Volume, Velocity, Veracity, Variety

Social Networks: streams of edges

Twitter: @x send I’ll give a talk at #shanin100 in #petersburg with @y

@x

#shanin100

@y

#petersburg
Keyword: #bitcoin, stream of tweets, of edges $(10^3/\text{min})$
Is a graph $G$ 3-colorable?

3COL: assign some color to each node, no unicolor edge

- NP-complete, Testable (GGR 98) with $O(1/\varepsilon^2)$ queries
  - $\varepsilon$-noise: $\varepsilon.n$ insertions or deletions (Edit distance)
  - Sample on dense graphs: $k$ random vertices, $G_k$ the random subgraph

Let $G' = G + \varepsilon$

How can we decide if $G$ is 3COL?

- If $G'$ is $2.\varepsilon$-far from 3COL, then $G$ is at least $\varepsilon$-far from 3COL.
Does the graph $G$ has a large $\gamma$-clique?

**Problem $\gamma$-LS**: existence of a large dense subgraph, i.e. a set large subgraph $(S,E(S))$ such that $|E(S)| > \gamma \cdot |S|^2$

- NP-complete,
- Most Cluster detection algorithms use the Conductance

Let $G' = G + \varepsilon$

**How can we decide if $G$ has a large $\gamma$-clique?**

- Samples: uniform edges in a stream
- Arguments: Giant components in random graphs
Graphs and Streaming edges

- Samples in dynamic sparse graphs
- Global graph versus a stream of edges:

\[ e_1, e_2, \ldots, e_m \]

Graph and Samples

Stream and Samples
Samples

• **Fixed graphs:** let \( r \) random nodes *proportional to their degree*

• **Katzir's method:**
  \[
  \varphi_1 = \sum_{i=1}^{r} d_i \quad \varphi_{-1} = \sum_{i=1}^{r} \frac{1}{d_i} \quad C = \text{number of collisions}
  \]
  \[
  R = \varphi_1 \cdot \varphi_{-1} - r
  \]
  \[
  \frac{R}{C} \approx n, \text{the size of the graph}
  \]

• **Streaming:** count \( m \) the number of edges, \( \frac{m}{d_{\text{average}}} \) approximates \( n \)

• **Uniform edges yield random nodes proportional to their degree**

• **Reservoir sampling** \( R(k) \): each edge has probability \( \frac{k}{m} \) to be in \( R \).
2. Distributions on graphs

- Erdos-Renyi: $G(n,p)$
- Preferential Attachment: $PA(n,m)$
- **Configuration Model** (Any Degree distribution)

Social graphs follow a Power law: $\text{Prob}_\Omega[\ \text{degree}(x) = i\ ] = \frac{c}{i^2}$

Bigdata follow specific distributions

$m = n \cdot \log n$

$d_{\text{max}} = \sqrt{c \cdot n}$
Configuration model: (3, 2, 1): random $\pi(i)=j$
3. Algorithm to detect γ-cliques

• **Hypothesis:** graphs follow a power law degree distribution, large S

• Streaming probabilistic **Detection Algorithm**\((k,m, \gamma)\):
  1. Sample edges with a Reservoir \((k)\) (probability \(\frac{k}{m} > \sqrt{a/n} \))
  2. If largest connected component \(C\) has size > \(c_0\) then Yes else No

**Theorem 1:** If \(m\) is large enough and \(|S| > \frac{m}{\gamma.k}\), then:

(a) If \(S\) is a \(\gamma\)-clique then \(\text{Prob}_\Omega[\text{Detection Algorithm Accepts}] > 1 - \delta\),

(b) If there is no large \(\gamma\)-clique \(\text{Prob}_\Omega[\text{Detection Algorithm Rejects}] > 1 - \delta\)
Giant component and phase transition

• Erdos-Renyi: $G(n,p)$  
  \[ p > \frac{1}{n} \] \quad \text{giant component}

• Configuration Model  
  \[ E[D^2] - 2E[D] > 0 \] \quad \text{giant component (Molloy-}
  \text{Reed 2010)}

• $\gamma$ -clique: $G(n,p)$  
  \[ \text{Result 1: } p > \frac{1}{\gamma.n} \] \quad \text{then giant component}

• Proof of theorem 1 (a) $|S| = n$  
  \[ \frac{k}{m} > \frac{1}{\gamma.n} \text{ and } |S| = n > \frac{m}{\gamma.k} \]
Proof of theorem 1 (b)

- Use the statistical hypothesis
- Study the uniform configuration model
- Use fundamental results on random graphs (Molloy-Reed)

- Conclusion: almost surely, there is no giant component.
Generalization to dynamic graphs

- Sliding windows

- Erdos-Renyi $G(n,p)$: non-edges appear ($q$), edges disappear ($r$)

- Preferential Attachment: $PA(n,m)$, no edge removal

- **Configuration Model**: some edges disappear, some appear, the statistical degree distribution is fixed.
Sliding Windows

Assume a stream of edges with timestamps: $e_1, e_2, \ldots e_n \ldots$
Uniform Dynamics: remove random $q$ edges, rematch
S-concentrated model

\[ \pi(i) = j \quad \text{for } i, j \text{ in } S \quad (p = \gamma), \]

\[ \pi'(i) = j \quad \text{for } i, j \text{ in } V - S \]
Decide at any given time, if you choose the uniform dynamics or the S-concentrated dynamics.

Step dynamics of length $\Delta$ for windows of length $\tau$
Detection algorithm for the dynamic version

Temporal properties for general dynamics

◊ P : there is a t such that \( G_t \) satisfies P

□ P : for all t \( G_t \) satisfies P

Dynamic detection Algorithm. Let \( C(t) \) the largest connected component of \( R(k) \) at time before t. If \( |C(t)| > c_0 \), then Accept else Reject.

Theorem 2: If \( m \) is large enough and \( |S| > \frac{m}{\gamma k} \),

(a) For the S-step, \( \text{Prob}_\Omega[\text{Dynamic Algorithm Accepts}] > 1 - \frac{\delta^A}{\tau} \),

(b) For the uniform dynamics \( \text{Prob}_\Omega[\text{Dynamic Algorithm Rejects}] > 1 - t \cdot \frac{\delta}{\tau} \)
Correlation of two streaming graphs

Several streams: $S_1, S_2, \ldots$

$S_j$: we compress the stream to $C_j$ the union of the large connected components $C_j = \bigcup_i C_{i,j}$

The correlation between two streams is:

$$\rho(S_1, S_2) = J(C_1, C_2)$$

$J(C_1, C_2)$ is the Jaccard similarity (size of the intersection/size of the union)
Two streams
Correlations between 3 pairs

Correlation

Averaged correlation

20/05/2019
Search by correlation

1. Correlation between Streams of social graphs, based on their giant components

2. Distance between tags

3. Search by correlation: given tags $\sigma_1, \sigma_2 \ldots, \sigma_k$ find the closest tags.
Open problems: Explainability

1. **Search by correlation**: how to explain that a tag is not in the output list of the search?

2. Algorithm with a social function: admission to the University, to a service,...
   Explaination for a student not admitted?

3. Privacy. GDPR is a European standard for platforms. How to verify that a platform follows or is far from satisfying the GDPR?
Conclusion

1. Social graphs are dynamic data with error

2. They follow some statistical model: learning algorithms offer a representation for some distributions

3. Is the noise a problem or an asset?

4. Are good algorithms simple?

5. Explainations of an algorithm