Signal Processing on Databases

Jeremy Kepner

Lecture 5: Perfect Power Law Graphs: Generation, Sampling, Construction, and Fitting

This work is sponsored by the Department of the Air Force under Air Force Contract #FA8721-05-C-0002. Opinions, interpretations, recommendations and conclusions are those of the authors and are not necessarily endorsed by the United States Government.
Outline

- Introduction
- Sampling
- Sub-sampling
- Joint Distribution
- Reuter’s Data
- Summary
- Detection Theory
- Power Law Definition
- Degree Construction
- Edge Construction
- Fitting: $\alpha$, $N$, $M$
- Example
Goals

• Develop a background model for graphs based on “perfect” power law

• Examine effects of sampling such a power law

• Develop techniques for comparing real data with a power law model

• Use power law model to measure deviations from background in real data
DETECTION OF SIGNAL IN NOISE

ASSUMPTIONS
• Background (noise) statistics
• Foreground (signal) statistics
• Foreground/background separation
• Model ≈ reality

DETECTION OF SUBGRAPHS IN GRAPHS

Example subgraph of interest: Fully connected (complete)

Example background model: Powerlaw graph

Can we construct a background model based on power law degree distribution?
“Perfect” Power Law Matrix Definition

- Graph represented as a rectangular sparse matrix
  - Can be undirected, multi-edged, self-loops, disconnected, hyper edges, ...
- Out/in degree distributions are **independent** first order statistics
  - Only constraint: $\sum n(d_{\text{out}}) d_{\text{out}} = \sum n(d_{\text{in}}) d_{\text{in}} = M$
Power Law Distribution Construction

- Perfect power law matlab code

```matlab
function [di ni] = PPL(alpha,dmax,Nd)
logdi = (0:Nd) * log(dmax) / Nd;
di = unique(round(exp(logdi)));
logni = alpha * (log(dmax) - log(di));
ni = round(exp(logni));
```

- Parameters
  - alpha = slope
  - dmax = largest degree vertex
  - Nd = number of bins (before unique)

- Simple algorithm naturally generates perfect power law
- Smooth transition from integer to logarithmic bins
- “Poor man’s” slope estimator: $\alpha = \log(n_1)/\log(d_{\text{max}})$
Power Law Edge Construction

- Power law vertex list matlab code

```matlab
function v = PowerLawEdges(di,ni);
A1 = sparse(1:numel(di),ni,di);
A2 = fliplr(cumsum(fliplr(A1),2));
[tmp tmp d] = find(A2);
A3 = sparse(1:numel(d),d,1);
A4 = fliplr(cumsum(fliplr(A3),2));
[v tmp tmp] = find(A4);
```

- Degree distribution independent of
  - Vertex labels
  - Edge pairing
  - Edge order

- Algorithm generates list of vertices corresponding to any distribution
- All other aspects of graph can be set based on desired properties
Fitting $\alpha$, $N$, $M$

- Power law model works for any
  - $\alpha > 0$, $d_{\text{max}} > 1$, $N_d > 1$

- Desire distribution that fits
  - $\alpha$, $N$, $M$

- Can invert formulas
  - $N = \sum_i n(d_i)$
  - $M = \sum_i n(d_i) d_i$

- Highly non-linear; requires a combination of
  - Exhaustive search, simulated annealing, and Broyden’s algorithm

- Given $\alpha$, $N$, $M$ can solve for $N_d$ and $d_{\text{max}}$
- Not all combinations of $\alpha$, $N$, $M$ are consistent with power law
Example: Halloween Candy

Distribution parameters
- $M = 77$
- $N = 19$
- $M/N = 4.1$
- $n_1 = 8$
- $d_{\text{max}} = 15$
- $\alpha = 0.77$

Fit parameters
- $M = 77$
- $N = 21$
- $M/N = 3.7$

Procedure
- Estimate parameters from data
- Determine if viable power law fit
- Rebin measured to power law and compare
Outline

- Introduction

**Sampling**
- Graph construction
- Graphs from $E' \times E$
- Edge ordering and densification

- Sub-sampling

- Joint Distribution

- Reuter's Data

- Summary
Graph Construction Effects

• Generate a perfect power law NxN randomize adjacency matrix A
  – $\alpha = 1.3$, $d_{\text{max}} = 1000$, $N_d = 50$
  – $N = 18K$, $M = 84K$

• Make undirected, unweighted, with no self-loops
  $A = \text{triu}(A + A')$;
  $A = \text{double}(\text{logical}(A))$;
  $A = A - \text{diag}(\text{diag}(A))$;

• Graph theory best for undirected, unweighted graphs with no self-loops
• Often “clean up” real data to apply graph theory results
• Process mimics “bent broom” distribution seen in real data sets
Power Law Recovery

Procedure

• Compute $\alpha$, $N$, $M$ from measured

• Fit perfect power law to these parameters

• Rebin measured data using perfect power law degree bins

• Perfect power law fit to “cleaned up” graph can recover much of the shape of the original distribution
Correlation Construction Effects

• Generate a perfect power law NxN randomize incidence matrix E
  – $\alpha = 1.3$, $d_{\text{max}} = 1000$, $N_d = 50$
  – $N = 18K$, $M = 84K$

• Make unweighted and use to form correlation matrix A with no self-loops

  $E = \text{double}(\text{logical}(E))$;
  $A = \text{triu}(E' \times E)$;
  $A = A - \text{diag}(\text{diag}(A))$;

• Correlation graph construction from incidence matrix results in a “bent broom” distribution that strongly resembles a power law
Power Law Lost

Procedure

• Compute $\alpha$, $N$, $M$ from measured

• Fit perfect power law to these parameters

• Rebin measured data using perfect power law degree bins

• Perfect power law fit to correlation shows non-power law shape

• Reveals “witches nose” distribution
Power Law Preserved

- In degree is power law
  - $\alpha = 1.3$, $d_{\text{max}} = 1000$, $N_d = 50$
  - $N = 18K$, $M = 84K$

- Out degree is constant
  - $N = 16K$, $M = 84K$
  - Edges/row = 5 (exactly)

- Make unweighted and use to form correlation matrix $A$ with no self-loops

- Uniform distribution on correlated dimension preserves power law shape
Edge Ordering: Densification

- Compute $M/N$ cumulatively and piecewise for 2 orderings
  - Linear
  - Random

- By definition $M/N$ goes from 1 to infinity for finite $N$

- Elimination of multi-edges reduces $M$ and causes $M/N$ to grow more slowly

- “Densification” is the observation that $M/N$ increases with $N$
  - Densification is a natural byproduct of randomly drawing edges from a power law distribution
  - Linear ordering has constant $M/N$
Edge Ordering: Power Law Exponent ($\alpha$)

- Compute $\alpha$ cumulatively and piecewise for 2 orderings
  - Linear
  - Random

- Edge ordering and sampling have large effect on the power law exponent

- Power law exponent is fundamental to distribution
- Strongly dependent on edge ordering and sample size
Outline

- Introduction
- Sampling
  - Sub-sampling
- Joint Distribution
- Reuter’s Data
- Summary
Sub-Sampling Challenge

• Anomaly detection requires good estimates of background

• Traversing entire data sets to compute background counts is increasingly prohibitive
  – Can be done at ingest, but often is not

• Can background be accurately estimated from a sub-sample of the entire data set?
Sampling a Power Law

- Generate power law
- Select fraction of edges

Whole distribution

1/40 sample
Linear Degree Estimate

- Divide measured degree by fraction
- Accurate for high degree
- Overestimates low degree
- Can we do better?

Whole distribution

Linear estimate
Non-Linear Degree Estimate

- Assume power law input
- Create non-linear estimate
- Matches median degree

Whole distribution

Non-Linear estimate
Sub-Sampling Formula

- $f =$ fraction of total edges sampled
- $n_1 =$ # of vertices of degree 1
- $d_{\text{max}} =$ maximum degree
- Allowed slope: $\ln(n_1)/\ln(d_{\text{max}}/f) < \alpha < \ln(n_1)/\ln(d_{\text{max}})$

- Cumulative distribution
  $$P(\alpha, d) = \left( f^{1-\alpha} \frac{d_{\text{max}}^\alpha}{n_1} \right) \sum_{i<d} i^{1-\alpha} e^{-fi}$$

- Find $\alpha^*$ such that $P(\alpha^*, \infty) = 1$
- Find $d_{50\%}$ such that $P(\alpha^*, d_{50\%}) = 1/2$
- Compute $K = 1/(1 + \ln(d_{50\%})/\ln(f))$

- Non-linear estimate of true degree of vertex $v$ from sample $d(v)$
  $$d(v) = \frac{d(v)}{f^{1-1/(K d(v))}}$$
Outline

• Introduction

• Sampling

• Sub-sampling

• Joint Distribution
  • Measured
  • Expected
  • Time Evolution

• Reuter’s Data

• Summary
Joint Distribution Definitions

• Label each vertex by degree

• Count number of edges from $d_{out}$ to $d_{in}$: $n(d_{out}, d_{in})$

• Rebin based on perfect power law model

• Can compare measured vs. expected

• Power law model allows precise quantitative comparison of observed data with a model
**Measured Joint Distribution**

- Measured distribution is highly sparse
- Rebinning based on power law fit degree bins makes most bins not empty
Expected Joint Distribution

Using $n(d_{out})$ and $n(d_{in})$ can compute expected $n(d_{out},d_{in}) = n(d_{out}) \times n(d_{in})/M$
Measured/Expected Joint Distribution

- Ratio of measured to expected highlights surpluses △, deficits ▽, typical edges □
- Binning reduces Poisson fluctuations and allows for more meaningful selection

\[
\log_{10}(n)
\]
Measured/Expected Joint Distribution

- Ratio of measured to expected highlights surpluses $\triangle$, deficits $\triangledown$, typical edges $\Box$
- Binning reduces Poisson fluctuations and allows for more meaningful selection
Selected Edges

- Ratio of measured to expected highlights surpluses △, deficits ▽, typical edges □
- Can use to select actual edges that correspond to fluctuations
Measured/Expected Random Edge Order

- Ratio of measured to expected highlights unusual correlations
Measured/Expected Linear Edge Order

- Ratio of measured to expected highlights unusual correlations
Outline

• Introduction

• Sampling

• Sub-sampling

• Joint Distribution

• Reuter’s Data

• Degree distributions
  • Correlation Graph
  • Densification
  • Joint distributions

• Summary
Reuter’s Incidence Matrix

- Entities extracted from Reuter’s Corpus
  - \( E(i,j) = \# \text{times entity appeared in document} \)

  - \( N_{doc} = 797677 \)
  - \( N_{ent} = 47576 \)
  - \( M = 6132286 \)

- Four entity classes with different statistics
  - LOCATION
  - ORGANIZATION
  - PERSON
  - TIME

- Fit power law model to each entity class
E(:,LOCATION) Degree Distribution

### Document Distribution

<table>
<thead>
<tr>
<th></th>
<th>M</th>
<th>N</th>
<th>M/N</th>
<th>α</th>
<th>M_{fit}</th>
<th>N_{fit}</th>
<th>M_{fit}/N_{fit}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Document</td>
<td>4694260</td>
<td>796414</td>
<td>5.89</td>
<td>1.70</td>
<td>4699280</td>
<td>811364</td>
<td>5.79</td>
</tr>
</tbody>
</table>

### Entity Distribution

<table>
<thead>
<tr>
<th></th>
<th>M</th>
<th>N</th>
<th>M/N</th>
<th>α</th>
<th>M_{fit}</th>
<th>N_{fit}</th>
<th>M_{fit}/N_{fit}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity</td>
<td>4694260</td>
<td>1786</td>
<td>2628</td>
<td>0.47</td>
<td>4696734</td>
<td>3680</td>
<td>1276</td>
</tr>
</tbody>
</table>
E(:, ORGANIZATION) Degree Distribution

**E(:, ORGANIZATION) document distribution**

- Measured data points
- Fitted line with parameter $\alpha$
- Model fit
- Rebin

**E(:, ORGANIZATION) entity distribution**

- Measured data points
- Fitted line with parameter $\alpha$
- Model fit
- Rebin

<table>
<thead>
<tr>
<th></th>
<th>$M$</th>
<th>$N$</th>
<th>$M/N$</th>
<th>$\alpha$</th>
<th>$M_{fit}$</th>
<th>$N_{fit}$</th>
<th>$M_{fit}/N_{fit}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Document</td>
<td>192390</td>
<td>69919</td>
<td>2.75</td>
<td>2.22</td>
<td>185800</td>
<td>85835</td>
<td>2.16</td>
</tr>
<tr>
<td>Entity</td>
<td>192390</td>
<td>141</td>
<td>1364</td>
<td>0.32</td>
<td>191943</td>
<td>205</td>
<td>936</td>
</tr>
</tbody>
</table>
E(:, PERSON) Degree Distribution

<table>
<thead>
<tr>
<th></th>
<th>M</th>
<th>N</th>
<th>M/N</th>
<th>α</th>
<th>M_{fit}</th>
<th>N_{fit}</th>
<th>M_{fit}/N_{fit}</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Document</strong></td>
<td>299333</td>
<td>170069</td>
<td>1.76</td>
<td>1.92</td>
<td>302478</td>
<td>170066</td>
<td>1.78</td>
</tr>
<tr>
<td><strong>Entity</strong></td>
<td>299333</td>
<td>37191</td>
<td>8.05</td>
<td>1.21</td>
<td>299748</td>
<td>37449</td>
<td>8.00</td>
</tr>
</tbody>
</table>
E(:,TIME) Degree Distribution

<table>
<thead>
<tr>
<th>Entity</th>
<th>M</th>
<th>N</th>
<th>M/N</th>
<th>(\alpha)</th>
<th>(M_{\text{fit}})</th>
<th>(N_{\text{fit}})</th>
<th>(M_{\text{fit}}/N_{\text{fit}})</th>
</tr>
</thead>
<tbody>
<tr>
<td>Document</td>
<td>946299</td>
<td>797677</td>
<td>1.19</td>
<td>2.37</td>
<td>944653</td>
<td>797734</td>
<td>1.18</td>
</tr>
<tr>
<td>Entity</td>
<td>946299</td>
<td>8444</td>
<td>112</td>
<td>0.83</td>
<td>947711</td>
<td>19848</td>
<td>47.7</td>
</tr>
</tbody>
</table>
**Procedure**

- Make unweighted and use to form correlation matrix $A$ with no self-loops

\[
E = \text{double}(\text{logical}(E)); \\
A = \text{triu}(E' \ast E); \\
A = A - \text{diag}(\text{diag}(A));
\]

- Perfect power law fit to correlation shows non-power law shape
- Reveals “witches nose” distribution
E(:,TIME)ᵀ x E(:,TIME)

**Procedure**

- Make unweighted and use to form correlation matrix A with no self-loops

\[
E = \text{double}(\text{logical}(E)));
\]

\[
A = \text{triu}(E' * E);
\]

\[
A = A - \text{diag}(\text{diag}(A));
\]

- Perfect power law fit to correlation shows non-power law shape
- Reveals “witches nose” distribution
Document Densification

- Constant M/N consistent with sequential ordering of documents
Entity Densification

• Increasing M/N consistent with random ordering of entities
Document Power Law Exponent ($\alpha$)

- Increasing $\alpha$ consistent with sequential ordering of documents
Entity Power Law Exponent (\(\alpha\))

- Decreasing \(\alpha\) consistent with random ordering of entities
$E(:,\text{LOCATION})$ Joint Distribution

- Ratio of measured to expected highlights surpluses $\triangle$, deficits $\nabla$, typical edges $\Box$
**E(:,ORGANIZATION) Joint Distribution**

- **Ratio of measured to expected highlights surpluses △, deficits ▽, typical edges □**
E(:,PERSON) Joint Distribution

- Ratio of measured to expected highlights surpluses $\Delta$, deficits $\nabla$, typical edges $\Box$
E(:,TIME) Joint Distribution

- Ratio of measured to expected highlights surpluses △, deficits ▽, typical edges □
Selected Edges $E(:,\text{LOCATION})$

- Highlights anomalous edges

**Typical**
- Document (low degree)
- Entity (medium degree)
- $1, 2, 3, \ldots$

**Deficit**
- Document (very low degree)
- Entity (medium degree)
- All $\sim 1$

**Surplus**
- Document (very high degree)
- Entity (medium degree)

<table>
<thead>
<tr>
<th>Entity (medium degree)</th>
<th>aruba</th>
<th>isle of man</th>
<th>tahiti</th>
</tr>
</thead>
<tbody>
<tr>
<td>19970425_538281.txt</td>
<td>3</td>
<td>6</td>
<td>2</td>
</tr>
</tbody>
</table>
Selected Edges $E(:, PERSON)$

- **Typical**
- **Deficit**

<table>
<thead>
<tr>
<th>Document (low degree)</th>
<th>Jeremy Smith</th>
<th>Samir Shah</th>
</tr>
</thead>
<tbody>
<tr>
<td>19970106_289115.txt</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>19970313_439431.txt</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

- **Entity (high degree)**
- **Entity (low degree)**
- **Surplus**

- **Highlights anomalous edges**
Summary

- Develop a background model for graphs based on “perfect” power law
  - Can be done via simple heuristic
  - Reproduces much of observed phenomena

- Examine effects of sampling such a power law
  - Lossy, non-linear transformation of graph construction mirrors many observed phenomena

- Traditional sampling approaches significantly overestimate the probability of low degree vertices
  - Assuming a power law distribution it is possible to construct a simple non-linear estimate that is more accurate

- Develop techniques for comparing real data with a power law model
  - Can fit perfect power-law to observed data
  - Provided binning for statistical tests

- Use power law model to measure deviations from background in real data
  - Can find typical, surplus and deficit edges
Example Code & Assignment

- Example Code
  - d4m_api/examples/2Apps/3PerfectPowerLaw

- Assignment 4
  - Compute the degree distributions of cross-correlations you found in Assignment 2
  - Explain the meaning of each degree distribution