Network Evolution

CS224W: Machine Learning with Graphs
Jure Leskovec and Baharan Mirzasoleiman, Stanford
http://cs224w.stanford.edu
Evolving Networks

- **Evolving Networks** are networks that change as a function of time.

- Almost all real world networks evolve either by **adding** or **removing nodes or links** over time.

- **Examples:**
  - **Social networks:** people make and lose friends and join or leave the network.
  - **Internet, web graphs, E-mail, phone calls, P2P networks, etc.**

Collaborations in the journal *Physical Review Letters (PRL)*  

[Perra et al. 2012]
Evolving Network Example

- **Visualization of the student collaboration network**

Nodes represent the students. An edge exists between two nodes if any of the two ever reported collaboration with the other in any of the assignments used to construct the network.

[Burstein et al. 2018]
Evolving Network Example

- Evolution of the pooled R&D network for the nodes belonging to the ten largest sectors


[Tomasello et al. 2017]
Evolving Network Example

- Evolution of five selected sectoral R&D networks

Blue nodes represent the firms strictly belonging to the examined sector, while orange nodes represent their alliance partners belonging to different sectors [Tomasello et al. 2017]
Evolving Network Example

- Evolving network structure of academic institutions

Community structure, indicated by color, for the networks from the three years 2011 to 2013. Different communities are indicated by different colors.

[Wang et al. 2017]
Evolving Network Example

- The largest components in Apple’s inventor network over a 6-year period

Each node reflects an inventor, each tie reflects a patent collaboration. Node colors reflect technology classes, while node sizes show the overall connectedness of an inventor by measuring their total number of ties/collaborations (the node’s so-called degree centrality).

[kenedict.com]
How do networks evolve?

- How do networks evolve at the macro level?
  - Evolving network models, densification
- How do networks evolve at the meso level?
  - Network motifs, communities
- How do networks evolve at the micro level?
  - Node, link properties (degree, network centrality)
Macroscopic Evolution of Networks
How do networks evolve at the macro level?

What are global phenomena of network growth?

Questions:

- What is the relation between the number of nodes $n(t)$ and number of edges $e(t)$ over time $t$?
- How does diameter change as the network grows?
- How does degree distribution evolve as the network grows?
Network Evolution

- $N(t)$ ... nodes at time $t$
- $E(t)$ ... edges at time $t$
- Suppose that
  \[ N(t + 1) = 2 \cdot N(t) \]
- Q: what is:
  \[ E(t + 1) = \text{?} \quad \text{Is it } 2 \cdot E(t)? \]
- A: More than doubled!
  - But obeying the **Densification Power Law**
Q1) Network Evolution

- What is the relation between the number of nodes and the edges over time?
- First guess: constant average degree over time
- Networks become denser over time
- Densification Power Law:

\[ E(t) \propto N(t)^a \]

\[ a \text{ ... densification exponent (1 ≤ } a \leq 2) \]
Densification Power Law

- the number of edges grows faster than the number of nodes – **average degree is increasing**

\[ E(t) \propto N(t)^a \]

or equivalently

\[
\frac{\log(E(t))}{\log(N(t))} = \text{const}
\]

\[ a \] ... densification exponent: \( 1 \leq a \leq 2 \):

- \( a=1 \): **linear growth** – constant out-degree (traditionally assumed)
- \( a=2 \): **quadratic growth** – fully connected graph
Prior models and intuition say that the network diameter slowly grows (like $\log N$)

Diameter shrinks over time

- As the network grows the distances between the nodes slowly decrease

How do we compute diameter in practice?

- **Long paths**: Take 90th-percentile or average path length (not the maximum)
- **Disconnected components**: Take only largest component or average only over connected pairs of nodes
Diameter of a Densifying $G_{np}$

Is shrinking diameter just a consequence of densification? (answer by simulation)

Densifying random graph has increasing diameter

$\Rightarrow$ There is more to shrinking diameter than just densification!
Does the changing degree sequence explain the shrinking diameter?

Compare diameter of a:
- Real network (red)
- Random network with the same degree distribution (blue)

Densification + degree sequence gives shrinking diameter
How does degree distribution evolve to allow for densification?

Option 1) Degree exponent $\gamma_t$ is constant:

- Fact 1: If $\gamma_t = \gamma \in [1, 2]$, then: $\alpha = 2/\gamma$

Power-laws with exponents $<2$ have infinite expectations.
So, by maintaining constant degree exponent $\alpha$ the average degree grows.
How does degree distribution evolve to allow for densification?

Option 2) $\gamma_t$ evolves with graph size $n$:

Fact 2: If $\gamma_t = \frac{4n_t^{x-1} - 1}{2n_t^{x-1} - 1}$, then: $a = x$

Notice: $\gamma_t \to 2$ as $n_t \to \infty$

Remember, the expected degree in a power law is:

$$E[X] = \frac{\gamma_t - 1}{\gamma_t - 2} x_m$$

So $\gamma_t$ has to decay as a function of graph size $n_t$ for the avg. degree to go up.
Want to model graphs that densify and have shrinking diameters

Intuition:
- How do we meet friends at a party?
- How do we identify references when writing papers?
The Forest Fire model has 2 parameters:
- $p$ ... forward burning probability
- $r$ ... backward burning probability

The model: Directed Graph
- Each turn a new node $v$ arrives
- Uniformly at random choose an “ambassador” node $w$
- Flip 2 coins sampled from a geometric distribution (based on $p$ and $r$) to determine the number of in- and out-links of $w$ to follow, i.e., to ”spread the fire” along
- “Fire” spreads recursively until it dies
- New node $v$ links to all burned nodes

Geometric distribution:
$$\Pr(X = k) = (1 - p)^{k-1} p$$
The Forest Fire model

1. \( v \) chooses an ambassador node \( w \) uniformly at random, and forms a link to \( w \)
2. Generate two random numbers \( x \) and \( y \) from geometric distributions with means \( p/(1-p) \) and \( rp/(1-rp) \)
3. \( v \) selects \( x \) out-links and \( y \) in-links of \( w \) incident to nodes that were not yet visited and form out-links to them
4. \( v \) applies step (2) to the nodes found in step (3)

Example:
1. Connect to a random node \( w \)
2. Sample \( x=2, y=1 \)
3. Connect to 2 out- and 1 in-links of \( w \), namely \( a, b, c \)
4. Repeat the process for \( a, b, c \)
Forest Fire Model

- Forest Fire generates graphs that **densify** and have **shrinking diameter**

![Graphs showing densification and diameter](Image)

- \( E(t) \)
- \( N(t) \)

**Equation:**
\[
E(t) = 5.2 \times 10^{-1} t^{1.32}
\]

**R²:** 1.00

**Densification:**
- linear growth
- \( N(t) \): number of nodes
- \( E(t) \): number of edges

**Diameter:**
- decreasing trend
Forest Fire Model

- Forest Fire also generates graphs with power-law degree distribution

![Graphs showing power-law degree distribution](image)

- Log count vs. log in-degree
- Log count vs. log out-degree
- Fix backward probability $r$ and vary forward burning prob. $p$

- Notice a sharp transition between sparse and clique-like graphs

- The “sweet spot” is very narrow
Temporal Networks
Temporal Networks

- **Temporal network**: A sequence of static directed graphs over the same (static) set of nodes $V$

- Each **temporal edge** is a timestamped ordered pair of nodes $(e_i = (u, v), t_i)$, where $u, v \in V$ and $t_i$ is the timestamp at which the edge exists
Temporal Networks

- Edges of a **temporal network** are active only at certain points in time.

- A **temporal network** is a sequence of static directed graphs over the same (static) set of nodes $V$.

\[
t = 1 
\]
\[
t = 2 
\]
\[
t = 3 
\]
**Temporal Network Examples**

- **Communication:** Email, phone call, face-to-face
- **Proximity networks:** Same hospital room, meet at conference, animals hanging out
- **Transportation:** train, flights...
- **Cell biology:** protein-protein, gene regulation

Email communication. A typical week of activity during Nov 2001 using 24-hour windows (left) and aggregated static graph (right). Nodes represent employees; a link between two employees exists if an email was sent by one of them to the other in that 24-hour window.

[Tang et al, 2010]
Temporal Network Examples

- **Communication**: Email, phone call, face-to-face
- **Proximity networks**: Same hospital room, meet at conference, animals hanging out
- **Transportation**: train, flights...
- **Cell biology**: protein-protein, gene regulation

Aggregated networks for two different days of the Science Gallery museum deployment. Nodes are colored according to the corresponding visitor's entry time slot. The network diameter is highlighted in each case.

[Isella et al, 2011]
Microscopic Evolution of Networks
Microscopic Evolution

- How do networks evolve at the micro level?
  - What are local phenomena of network growth?

- Questions:
  - How do we define paths and walks in temporal networks?
  - How can we extend network centrality measures to temporal networks?
Temporal Path

- A **temporal path** is a sequence of edges \((u_1, u_2, t_1), (u_2, u_3, t_2), \ldots, (u_j, u_{j+1}, t_j)\), for which \(t_1 \leq t_2 \leq \cdots \leq t_j\) and each node is visited at most once.

**Example:**
- The sequence of edges \([(5,2),(2,1)]\) together with the sequence of times \(t_1, t_3\) is a temporal path.
How to Find Temporal Shortest Paths?

**TPSP-Dijkstra algorithm**: An adaptation of Dijkstra using a priority queue

**Notation:**
- \( n_s \): source node
- \( n_t \): target node
- \( t_q \): time of the query (we calculate the distance from \( n_s \) to \( n_t \) between time \( t_s \) and time \( t_q \))
- \( t_s \): time that a node/edge joins the network
- \( t_e \): time that a node/edge leaves the network
- \( w \): edge weights
- \( d[v] \): distance of \( n_s \) to \( v \)
- \( PQ \): priority queue

**Algorithm 2.1**: TPSP-Dijkstra(\( TEG, n_s, n_t, t_q \))

**Input**: Temporal evolving graph \( TEG = (V, E, w, t_s, t_e) \), \( n_s, n_t \in \text{sub-}V(t_q) \), and query time \( t_q \)

**Output**: Distance of the shortest path \( p \subseteq \text{sub-TEG}(TEG, t_q) \)

1. \( PQ \leftarrow \emptyset \)  \(\triangleright\) Set distance to \( \infty \) for all nodes
2. for all \( v \in V \land t_s(v) \leq t_q \land t_q < t_e(v) \) do
3. \( d[v] \leftarrow \infty \)  \(\triangleright\) Set distance to 0 for \( n_s \)
4. end for
5. \( d[n_s] \leftarrow 0 \)
6. \( PQ.Insert(n_s, d[n_s]) \)  \(\triangleright\) Insert (nodes, distances) to \( PQ \)
7. while \( !PQ.empty() \) do
8. \( u \leftarrow PQ.ExtractMin() \)
9. if \( u = n_t \) then
10. \( \text{return } d[n_t] \)  \(\triangleright\) Extract the closest node from \( PQ \)
11. end if
12. for all \( e = (u, v) \in E \) do
13. if \( t_s(e) \leq t_q \land t_q < t_e(e) \land d[u] + w(e) < d[v] \) then
14. \( d[v] \leftarrow d[u] + w(e) \)  \(\triangleright\) If so, update \( v \)'s distance from \( n_s \)
15. if \( v \notin PQ \) then
16. \( PQ.Insert(v, d[v]) \)
17. else
18. \( PQ.DecreaseKey(v, d[v]) \)
19. end if
20. end if
21. end for
22. end while
23. return \( \infty \)
**TPSP-Dijkstra algorithm**: An adaptation of Dijkstra using a priority queue

Example of a temporally evolving graph. Shortest path from a to f are marked in thick lines.

**Algorithm 2.1**: TPSP-Dijkstra(TEG, n_s, n_t, t_q)

| Input: Temporal evolving graph TEG = (V, E, w, t_s, t_e), n_s, n_t ∈ sub-V(t_q), and query time t_q |
| Output: Distance of the shortest path p ∈ sub-TEG(TEG, t_q) |

1: \( PQ ← \emptyset \)
2: for all \( v ∈ V \land t_s(v) ≤ t_q \land t_q < t_e(v) \) do
3: \( d[v] ← \infty \)
4: end for
5: \( d[n_s] ← 0 \)
6: \( PQ.\text{Insert}(n_s, d[n_s]) \)
7: while \( !PQ.\text{empty()} \) do
8: \( u ← PQ.\text{ExtractMin}() \)
9: if \( u = n_t \) then
10: return \( d[n_t] \)
11: end if
12: for all \( e = (u, v) ∈ E \) do
13: if \( t_s(e) ≤ t_q \land t_q < t_s(e) \land d[u] + w(e) < d[v] \) then
14: \( d[v] ← d[u] + w(e) \)
15: if \( v \notin PQ \) then
16: \( PQ.\text{Insert}(v, d[v]) \)
17: else
18: \( PQ.\text{DecreaseKey}(v, d[v]) \)
19: end if
20: end if
21: end for
22: end while
23: return \( \infty \)
Temporal Closeness: Measure of how close a node is to any other node in the network at time interval \([0, t]\).

- Sum of shortest (fastest) temporal path lengths to all other nodes

\[
c_{clos}(x, t) = \frac{1}{\sum_y d(y, x|t)}
\]

length of the temporal shortest path from \(y\) to \(x\) from time 0 to time \(t\)

Example:

\[
c_{clos}(A, 2) = \frac{1}{1 + 1 + 2 + 2 + 2} = 0.1
\]

\[
d(A, C|t = 2)
\]
**Intuition:**

- Node $a$ initially receives many in-links and it should be considered important.
- After time $t = 8$, it does not receive any more in-links and thus its importance should diminish.
Temporal Walk

- A **temporal or time-respecting walk** is a sequence of edges \((u_1, u_2, t_1), (u_2, u_3, t_2), \ldots, (u_j, u_{j+1}, t_j)\), for which \(t_1 \leq t_2 \leq \cdots \leq t_j\).

![Diagram of temporal walk](image)

- **Example:**
  - The sequence of edges \([(5,2),(2,1),(1,5)]\) together with the sequence of times \(t_1, t_3, t_4\) is a temporal walk.
**Idea**: Make a random walk only on temporal or time-respecting paths

- Time-stamps increase along the path

- $c \rightarrow b \rightarrow a \rightarrow c$: time respecting
- $a \rightarrow c \rightarrow b \rightarrow a$: not time respecting
How can we calculate the probability of a temporal path?

- The probability $P[(u, x, t_2)|(v, u, t_1)]$ of taking $(u, x, t_2)$ given that we arrived via $(u, v, t_1)$ decreases as the time difference $(t_2 - t_1)$ increases.
- This can be modeled by an exponential distribution:
  $$P[(u, x, t_2)|(v, u, t_1)] = \beta^{\mid\Gamma_u\mid}$$

- $\Gamma_u$ is the set of all temporal edges from $u$ during the time $t' \in [t_1, t_2]$
  $$\Gamma_u = \{(u, y, t') | t' \in [t_1, t_2], y \in V\}$$

- Smaller values of $\beta$ increase the probability of walks stopping at high degree nodes, but we get a slower convergence.
As $t \to \infty$, the temporal PageRank converges to the static PageRank: why?

**Temporal PageRank is running regular PageRank on a time-augmented graph:**

- Connect graphs at different time steps via time hops, and run PageRank on this time-extended graph
  - Node $u$ at $t_1$ becomes a node $(u, t_1)$ in this new graph
  - Transition probabilities given by
    \[
    P[(u, t_1), (x, t_2)]|(v, t_0), (u, t_1)) = \beta|\Gamma_u|
    \]
- As $t \to \infty$, $\beta|\Gamma_u|$ becomes the uniform distribution
  - Graph looks as if we superimposed the original graphs from each time step → back to regular PageRank
Constructing the time-augmented graph:

Starting at A at t=1…
- can go to B at t=1, C at t=2, or B at t=3

Starting at E at t=1…
- can go to C at t=1, or C at t=3

Repeat for all nodes across all time steps!
**Temporal PageRank: Definition**

- **Temporal PageRank:**
  
  \[
  r(u, t) = \sum_{v \in V} \sum_{k=0}^{t} (1 - \alpha) \alpha^k \sum_{z \in Z(v, u | t), |z|=k} P[z | t]
  \]

  - \( Z(v, u | t) \) is a set of all possible temporal walks from \( v \) to \( u \) until time \( t \)
  - \( \alpha \) is the probability of starting a new walk

- As \( t \to \infty \), the temporal PageRank converges to the static PageRank

- **Temporal Personalized PageRank:**
  
  \[
  r(u, t) = \sum_{v \in V} \sum_{k=0}^{t} (1 - \alpha) \alpha^k \frac{h^*(v)}{h'(v)} \sum_{z \in Z(v, u | t), |z|=k} P[z | t]
  \]

  - \( h^* \): personalization vector
  - \( h' \): walk starting probability vector, 
    \[
    h'(u) = \frac{|(u,v,t) \in E: \forall v \in V|}{|E|}
    \]
The weighted number of temporal walks $z$ is then defined as

$$c(z|t) = (1 - \beta) \prod_{(u_i,u_{i+1},t_{i+1}),(u_{i-1},u_i,t_i) \in z} \beta^{\Gamma_{u_i}}$$

Temporal walk from $v = u_1$ to $u = u_5$

$$z = \{(u_1,u_2),(u_2,u_3),(u_3,u_4),(u_4,u_5)\}$$

- $Z(u, v|t)$ is a set of all possible temporal walks from $v$ to $u$ until time $t$
- $P[z|t]$ is the probability of a particular walk $z \in Z(v,u|t)$

$$P[z \in Z(v,u|t)] = \frac{c(z|t)}{\sum_{z' \in Z(v,x|t)} c(z'|t)}$$

number of all temporal walks that start at node $v$ and have the same length as $z$
Temporal PageRank: how to compute?

- $r(u)$: Temporal PageRank estimate of $u$
- $s(u)$: Count of active walks visiting $u$

```
Algorithm 1. Temporal PageRank

input : $E$, transition probability $\beta \in (0, 1]$, probability of initiating a new walk $\alpha$
1 $r = 0, s = 0;$
2 foreach $(u, v, t) \in E$ do
3 \hspace{1em} $r(u) = r(u) + (1 - \alpha);$ \\
4 \hspace{1em} $s(u) = s(u) + (1 - \alpha);$ \\
5 \hspace{1em} $r(v) = r(v) + s(u) \alpha;$ \\
6 \hspace{1em} if $\beta \in (0, 1)$ then
7 \hspace{2em} $s(v) = s(v) + s(u)(1 - \beta) \alpha;$ \\
8 \hspace{2em} $s(u) = s(u) / \beta;$ \\
9 \hspace{1em} else if $\beta = 1$ then \\
10 \hspace{2em} $s(v) = s(v) + s(u) \alpha;$ \\
11 \hspace{2em} $s(u) = 0;$ \\
12 normalize $r;$ \\
13 return $r$;
```

- $\triangleright$ Initiating a new walk with probability $1 - \alpha$
- $\triangleright$ With probability $\alpha$ we continue active walks that wait in $u$
- $\triangleright$ Increment the active walks (active mass) count in the node $v$ with appropriate normalization $1-\beta$
- $\triangleright$ decrements the active mass count in node $u$
- $\triangleright$ Increments the active walks (active mass) count in the node $v$ with appropriate normalization $1-\beta$
- $\triangleright$ decrements the active mass count in node $u$
Datasets:

- **Facebook**: A 3-month subset of Facebook activity in a New Orleans regional community. The dataset contains an anonymized list of wall posts (interactions).

- **Twitter**: Users’ activity in Helsinki during 08.2010–10.2010. As interactions we consider tweets that contain mentions of other users.

- **Students**: An activity log of a student online community at the University of California, Irvine. Nodes represent students and edges represent messages.
Experimental setup:

- For each network, static subgraph of n = 100 nodes is obtained by BFS from a random node.

- Edge weights are equal to the frequency of corresponding interactions and are normalized to sum to 1.

- Then a sequence of 100K temporal edges are sampled, such that each edge is sampled with probability proportional to its weight.

- In this setting, temporal PageRank is expected to converge to the static PageRank of a corresponding graph.

- Probability of starting a new walk is set to $\alpha = 0.85$, and transition probability $\beta$ for temporal PageRank is set to 0 unless specified otherwise.
Case Study — Temporal PageRank

- Comparison of temporal PageRank ranking with static PageRank ranking

  ![Comparison plots](image)

  Rank correlation between static and temporal PageRank is high for top-ranked nodes and decreases towards the tail of ranking.

- Rank quality (Pearson corr. coeff. Between static and temporal PageRank) and transition probability $\beta$

  ![Beta plots](image)

  Smaller $\beta$ corresponds to slower convergence rate, but better correlated rankings.
Case Study — Temporal PageRank

- **Adaptation to concept drift \((\beta = 0.5)\)**
  - We start with a temporal network sampled from some static network
  - After sampling 10K temporal edges \(E_1\), we change the weights of the static graph and sample another 10K temporal edges \(E_2\)
  - Similarly, a final set of edges \(E_3\) is sampled after changing the weights
  - The algorithm on the concatenated sequence \(E = \langle E_1, E_2, E_3 \rangle\)

Temporal PageRank is able to adapt to the changing distribution quite fast.
Error is the Euclidean distance on the PageRank vectors.
Mesoscopic Evolution of Networks
Mesoscopic Evolution

- **How do networks evolve at the mezo level?**
  - What are mesoscopic impact of network growth?

- **Questions:**
  - How does patterns of interaction change over time?
  - What can we infer about the network from the changes in temporal patterns?
- **$k$-node $l$-edge $\delta$-temporal motif**: is a sequence of $l$ edges $(u_1, v_1, t_1), (u_2, v_2, t_2), \ldots (u_l, v_l, t_l)$ such that
  - $t_1 < t_2 < \ldots < t_l$ and $t_l - t_1 \leq \delta$,
  - The induced static graph from the edges is connected and has $k$ nodes

- Temporal motifs offer valuable information about the networks’ evolution
  - For example to discover trends and anomalies in temporal networks
Temporal Motif Instances

- **Temporal Motif Instance**: A collection of edges in a temporal graph is an instance of a $\delta$-temporal motif $M$ if
  - It matches the same edge pattern, and
  - All of the edges occur in the right order specified by the motif, within a $\delta$ time window.
We study all 2- and 3- node motifs with 3 edges

- We do not discuss how to count temporal motifs here

The green background highlights the four 2-node motifs (bottom left) and the grey background highlights the eight triangles.
Case Study — Temporal Motifs

- Real-world temporal datasets

<table>
<thead>
<tr>
<th>Table 1: Summary statistics of datasets.</th>
</tr>
</thead>
<tbody>
<tr>
<td>dataset</td>
</tr>
<tr>
<td>EMAIL-EU</td>
</tr>
<tr>
<td>PHONECALL-EU</td>
</tr>
<tr>
<td>SMS-A</td>
</tr>
<tr>
<td>COLLEGEMsg</td>
</tr>
<tr>
<td>STACKOVERFLOW</td>
</tr>
<tr>
<td>BITCOIN</td>
</tr>
<tr>
<td>FBWall</td>
</tr>
<tr>
<td>WIKITalk</td>
</tr>
<tr>
<td>PHONECALL-ME</td>
</tr>
<tr>
<td>SMS-ME</td>
</tr>
</tbody>
</table>

[Paranjape et al. 2017]
**Case Study — Temporal Motifs**

- **Blocking communication**
  - If an individual typically waits for a reply from one individual before proceeding to communicate with another individual.

Fraction of all 2 and 3-node, 3-edge \( \delta \)-temporal motif counts that correspond to two groups of motifs (\( \delta = 1 \) hour). Motifs on the left capture “blocking” behavior, common in SMS messaging and Facebook wall posting, and motifs on the right exhibit “non-blocking” behavior, common in email.

[Paranjape et al. 2017]
Cost of Switching

- On Stack Overflow and Wikipedia talk pages, there is a high cost to switch targets because of peer engagement and depth of discussion.
- In the COLLEGE_MSG dataset there is a lesser cost to switch because it lacks depth of discussion within the time frame of \( \delta = 1\) hour.
- In EMAIL-EU, there is almost no peer engagement and cost of switching is negligible.

Distribution of switching behavior amongst the nonblocking motifs (\( \delta = 1\) hour). Switching is least common on Stack Overflow and most common in email.

[Paranjape et al. 2017]
Counts over various time scales for the motifs representing a node sending 3 outgoing messages to 1 or 2 neighbors in the COLLEGEMSG dataset

[Paranjape et al. 2017]
Case Study — Identifying Trends and Anomalies

- To spot trends and anomalies, we have to spot statistically significant temporal motifs
  - To do so, we must compute the expected number of occurrences of each motif
  - We study all 2- and 3-node motifs with 3 edges
Case Study — Financial Network

- A European country’s transaction log for all transactions larger than 50K Euros over 10 years from 2008 to 2018, with 118,739 nodes and 2,982,049 temporal edges ($\delta=90$ days)

Anomalies: We can localize the time the financial crisis hits the country around September 2011 from the difference in the actual vs. expected motif frequencies