Online Community Detection

Sébastien Loustau - Artfact

joint work with all the team !

Real Time Machine Learning

Abstract

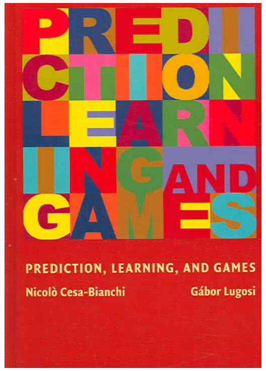
- Gentle start
  - *Online learning, graph notations and examples*
- Modularity-based approach
  - *"More weight intra-communities than inter-communities" ...*
  - *Louvain algorithm*
- Metropolis Hastings framework
  - *Approximate a global optimum with a stationary Markov chain*
  - *Improve convergence with an agglomerative hierarchical proposal*
  - *REal TIme Network Algorithm - RETINA -*
- Demo over synthetic and real datasets
  - *Synthetic example - SBM and PA models -*
  - *Digg/Enron example - news and emails -*
  - *Ticketing example - marketing automation -*

Sébastien Loustau, Yves Darmaillac.
MCMC Louvain for Online Community Detection.
arXiv:1612.01489, 2016.

Machine Learning

- Statistical Learning
  We observe a training sample $\{(X_i, Y_i), i = 1, \ldots, n\}$.
  We build a model/algorithm based on this dataset.
  New x *arrives. We predict* $\hat{y}$ *thanks to the model/algorithm.*

- Online Learning
  Data arrives sequentially.
  At eatch time t, *we want to update the decision based on past observations.*
  No assumption over the data mechanism.

- 

Graph clustering

- Higher density of edges within groups than between them.
- Spectral methods
  M. E. J. Newman.
  Finding community structure in networks using the eigenvectors of matrices.
  Phys. Rev. E (3), 74(3):036104, 19, 2006.
- Modularity maximization : a NP-hard problem
  Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre.
  Fast unfolding of communities in large networks.
  Journal of statistical mechanics: theory and experiment, 2008, P10008.
- Dynamic graph clustering : a greedy algorithm
  Robert Görke, Pascal Maillard, Christian Staudt, and Dorothea Wagner.
  Modularity driven clustering of dynamic graphs. In Proceedings of the 9th International Conference
  on Experimental Algorithms, SEA'10, pages 436-448, Berlin, Heidelberg, 2010. Springer-Verlag.

Graph notations

- Let $G = (V, E)$ and undirected graph where $V = \{1 \dots N\}$ is a set of N vertices and E a set of edges $(i, j) \in V^2$
- G may be weighted :

  $w : E \rightarrow \mathbb{R}_+$
- Adjacency matrix

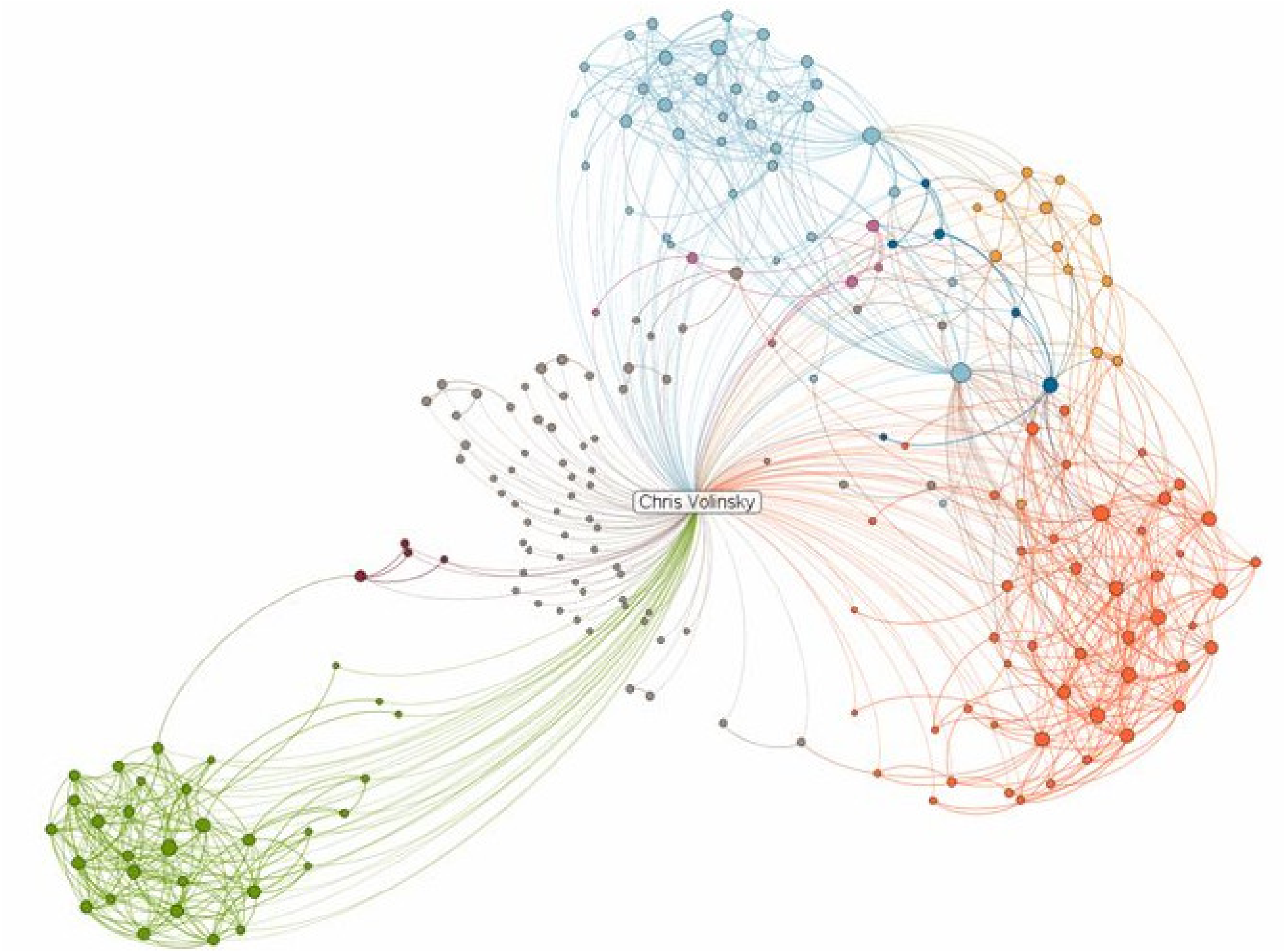
$$A = \begin{pmatrix} w(1,1) & w(1,2) & \dots & w(1,N) \\ & w(2,2) & \dots & w(2,N) \\ & & \dots & \dots \\ & & & w(N,N) \end{pmatrix} \quad = \quad \left(A_{ij}\right)_{i,j \in \{1 \dots N\}}$$

- Degree of vertex i : $k_i = \sum_{j=1}^{N} w(i,j) = \sum_{j=1}^{N} A_{ij}$
- Degree sum formula : $m = \frac{1}{2} \sum_{i=1}^{N} k_i$
- Coloration : $C \in \mathcal{C}$ any partition $C = \{c_1 \dots c_K\}$ of V ; c_i are called communities of coloration C.
- For convenience, $C(i) \in \{1 \dots K\}$ denotes the community which vertex i belongs to based on coloration C.
- $\mathcal{N}^C$ denotes a neighborhood of coloration C and is defined as :

  $\mathcal{N}^C := \{C' \in \mathcal{C} : \text{for exactly one vertex } i, C(i) \neq C'(i)\}$

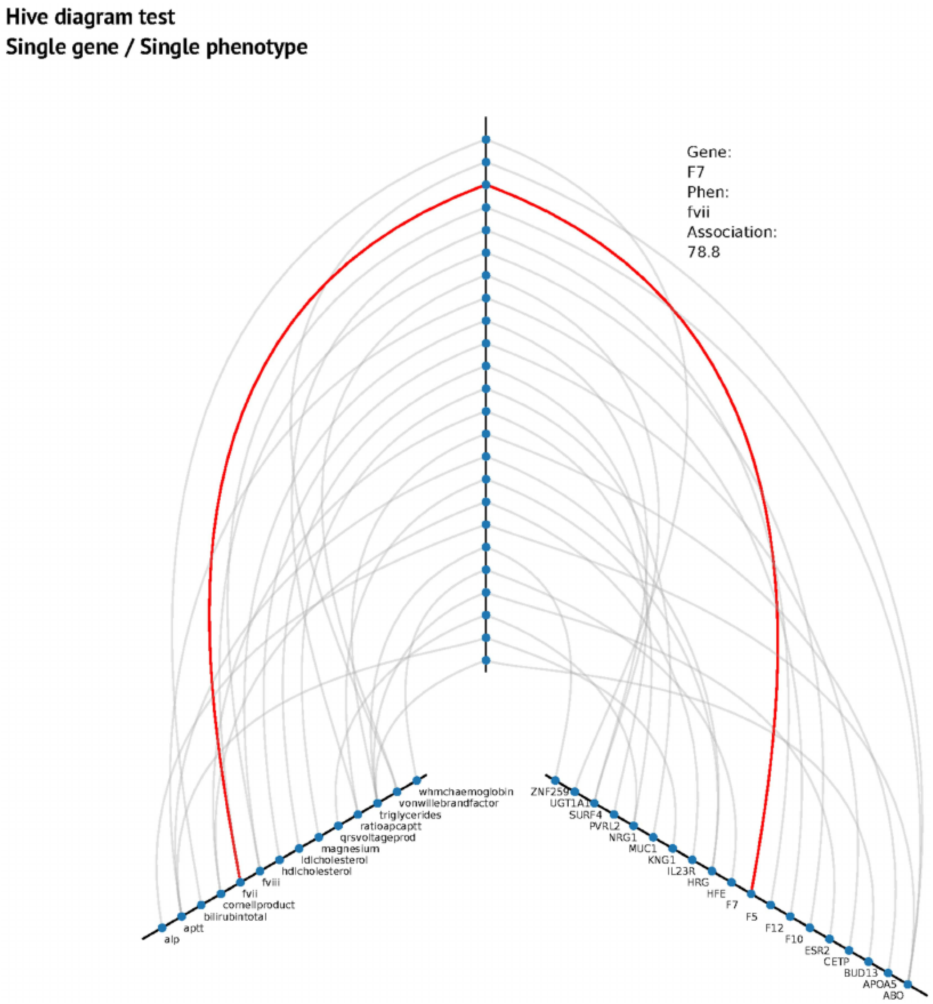
LinkedIn communities

- Try your own network with Socilab project
    - *Nodes : profiles*
    - *Edges : Connection*



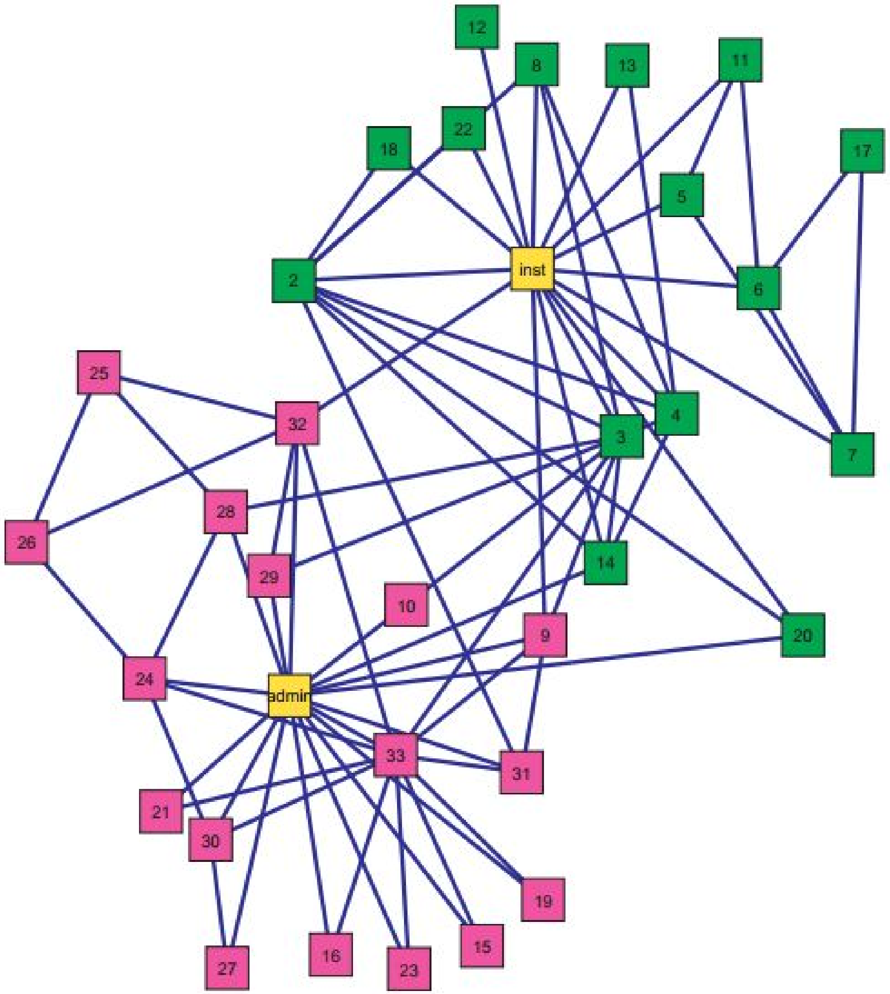
Hiveplots

- Hiveplot project from Martin Krzywinski, Genome Sciences Center, Vancouver
  - *Nodes : genes / proteins*
  - *Edges : biological interplay*



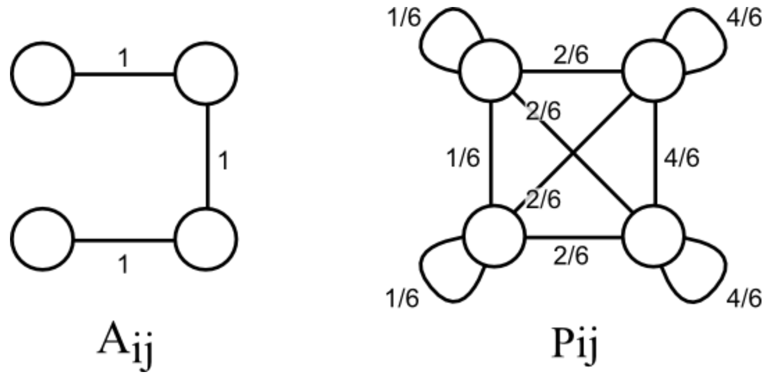
Zachary's Karate Club

- Source file and story
- Nodes : Karate club members
- Edges : interactions



Modularity

- We define $P = \dfrac{1}{2m}\mathbf{k}\mathbf{k}^T$
  - $\mathbf{k}$ *is the* N *elements vector of vertex degrees* k_i
  - $P_{ij} = \dfrac{k_i k_j}{2m}$



- Expected weight of edge (i, j) given i and j degrees, under uniform distribution.
  - $\mathbb{E}\left[w(i,j) \mid k_i, k_j\right] = P_{ij}$
  - $k_i = \sum_j P_{ij} = \sum_j A_{ij}$
- **Modularity of a coloration C is defined as** :

$$Q_G^{(C)} = \frac{1}{2m}\sum_{c \in C}\left[\sum_{i,j \in c} A_{ij} - \sum_{i,j \in c} P_{ij}\right] = \frac{1}{2m}\sum_{i,j \in V}\left[A_{ij} - P_{ij}\right]\delta\left(C(i), C(j)\right)$$

- Matrix notation : $Q_G^{(C)} = \mathrm{Tr}(S^T(A-P)S)$ where $S_{ik} = 1$ if i belongs to community k.
- Optimal coloration : $C^* = \underset{C \in \mathcal{C}}{\arg\max}\, Q_G^{(C)}$

Modularity gain

- Let C a coloration and $\mathcal{N}^C$ its neighborhood. Then any moove $C \to C' \in \mathcal{N}^C$ could be described:
  1. *by removing a vertex $i \in V$ from its community to create a new single node community $c^S = \{i\}$,*
  2. *by giving to a single node community $c^S = \{i\}$ an existing community $c \in C$.*
- We note $k_{i,c}^C = \sum_{j \in c} A_{ij}$ and $k_c^C = \sum_{i \in c} k_i$
- The delta of modularity for the first step is given by :

$$\Delta Q^{C \to C''} = -\frac{1}{m} \sum_{j \in V} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(C(j), C(i))$$
$$= -\frac{1}{m} \left(k_{i,C(i)}^C - A_{ii} - \frac{k_i}{2m} (k_{C(i)}^C - k_i) \right)$$

- The delta of modularity for the second step is given by :

$$\Delta Q^{C'' \to C'} = \frac{1}{m} \sum_{j \in V} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(C(j), c) = \frac{1}{m} \left(k_{i,c}^C - \frac{k_i k_c^C}{2m} \right)$$

- Then, any moove $C \to C' \in \mathcal{N}^C$ is computed thanks to:

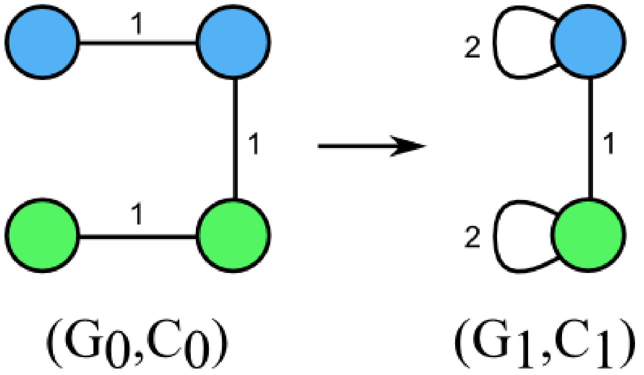
$$\Delta Q^{C \to C''} = \Delta Q^{C \to C'} + \Delta Q^{C' \to C''} \tag{1}$$

- We note $\mathcal{N}_i^C \subset \mathcal{N}^C$ the set of colorations obtained by moving vertex i.

Aggregation

- Given a graph G_0, an adjacency matrix A_0 and a coloration $C_0 = \{c_1 \dots c_K\}$ we build a new graph G_1 :
  - $V_1 = \{1 \dots K\}$
  - $E_1 = \{i', j' \in V_1 : A_{1_{i'j'}} > 0\}$
  - $A_{1_{i'j'}} = \sum_{\substack{i \in c_{i'} \\ j \in c_{j'}}} A_{0_{ij}}$

    $\Rightarrow P_{1_{i'j'}} = \sum_{\substack{i \in c_{i'} \\ j \in c_{j'}}} P_{0_{ij}}$



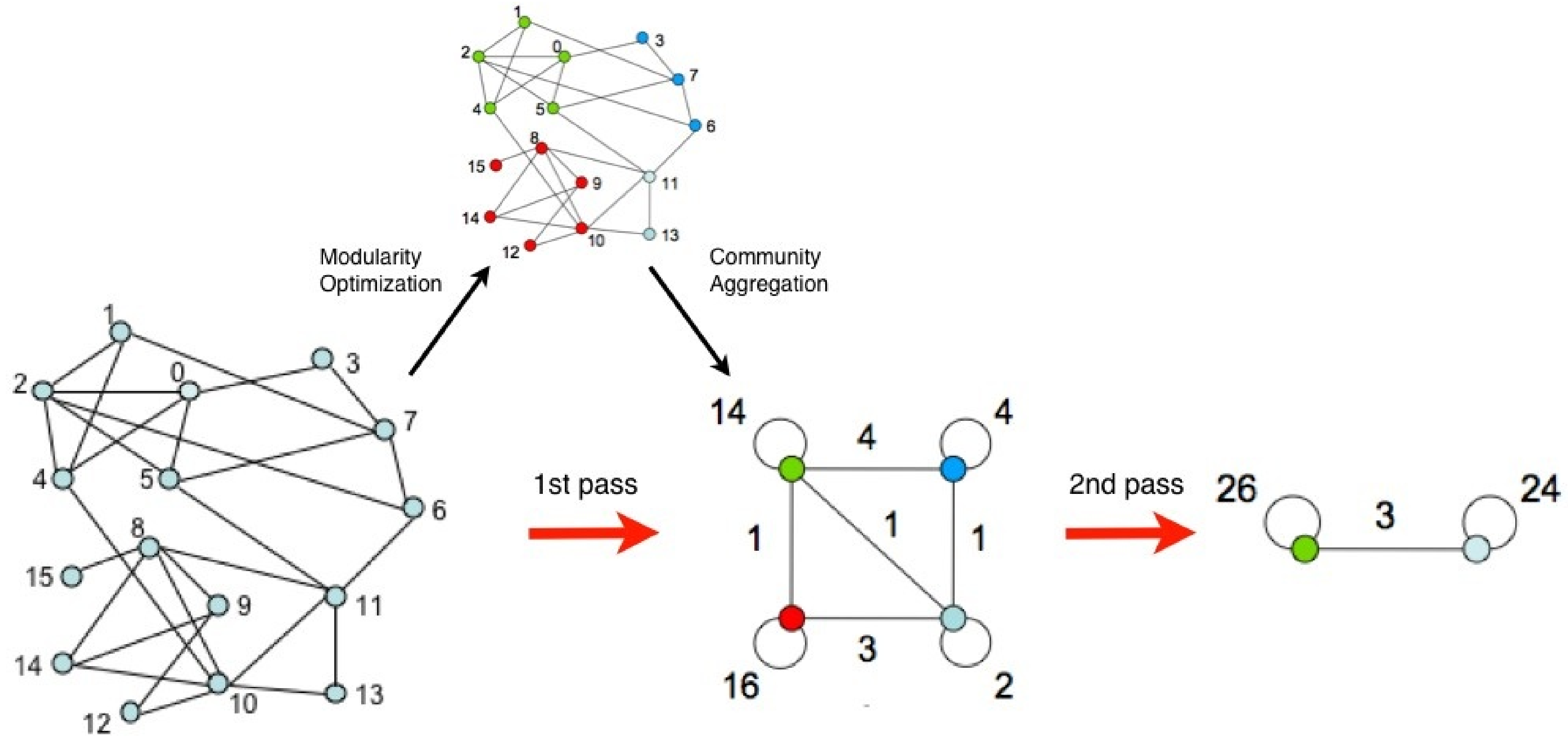
- We also build the coloration $C_1^S = \{c_1 \dots c_K\}$ on G_1 where $c_{i'} = \{i'\}$ for $i' = \{1 \dots K\}$.
- $Q_{G_0}^{(C_0)} = Q_{G_1}^{(C_1^S)}$

$$Q_G^{(C)} = \frac{1}{2m} \sum_{c \in C_0} \left[\sum_{i,j \in c} A_{0_{ij}} - \sum_{i,j \in c} P_{0_{ij}} \right]$$

$$= \frac{1}{2m} \sum_{i' \in \{1 \dots N'\}} \left[A_{1_{i'i'}} - P_{1_{i'i'}} \right]$$

$$= \frac{1}{2m} \sum_{i',j' \in V_1} \left[A'_{i'j'} - P'_{i'j'} \right] \delta\left(C_1^S(i'), C_1^S(j') \right) \blacksquare$$

Louvain algorithm



Louvain algorithm

- Initialization : $\epsilon > 0$, $G_0 = G$, $C_0 := \{c_1, \ldots, c_N\}$ single node coloration.
- For $l = 0, \ldots$
- First step : **Optimization**
  - *for any $i \in \sigma V_l$, i joins community $C(j^\star)$ where $j^\star \in \{j : \{i,j\} \in E\}$ maximizes $Q_G^{(C)}$. Repeat until no gain of modularity*
- Second step : **Aggregation**
  - *Construct G_{l+1} based on G_l and C_l.*
- Repeat until gain of modularity is smaller than a given $\varepsilon > 0$

General MH algorithm

- Approximate a global optimum with a stationary Markov chain
- Given a neighborhood $\mathcal{N}^C$,

General MH algorithm	
1 :	*given a graph* $G = (E, V)$, $C^0 \in \mathcal{C}$ *and* $\lambda > 0$
2 :	*For* $k = 1, \ldots, N$
3 :	*Generate a proposal* $\tilde{C} \sim p(\cdot \vert \mathcal{N}^{C^{k-1}})$
4 :	$C^k = \tilde{C}$ *with probability* $\rho(\lambda, \tilde{C}, C^{k-1})$
5 :	$C^k = C^{k-1}$ *otherwise.*

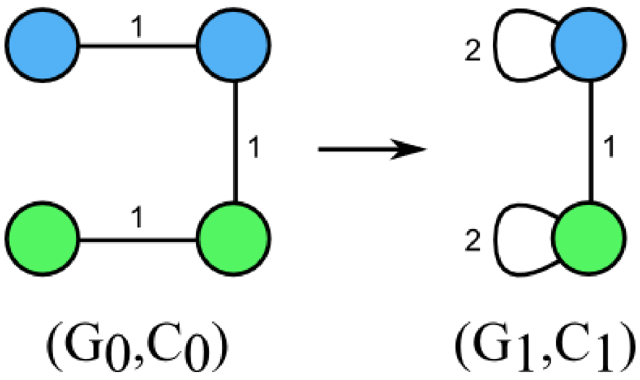
- Performances are driven by choices of $\mathcal{N}^C$, proposal $p(\cdot|\mathcal{N}^C)$, calculation of $\rho(\lambda, \cdot, \cdot)$ and adaptive temperature $\lambda > 0$.
- We define the Neighborhood $\mathcal{N}^C$ of coloration $C \in \mathcal{C}$ by:

$$\mathcal{N}^C = \{C' \in \mathcal{C} : \text{for exactly one vertex } i, C(i) \neq C'(i)\}$$

- Good news : $\Delta Q^{C \to C'} = Q_G^{(C')} - Q_G^{(C)}$ can be computed easily.

Hierarchical Graphs

- Let $(E_l, V_l)_{l=1}^{L}$ a sequence of weighted and undirected graphs such that :
  - *V_{l+1} is a partition of V_l*
  - *$E_{l+1} = \mathrm{agg}_{V_{l+1}}(E_l)$, ie edges are summed based on V_{l+1}.*
- We denote by $(A^{(l)})_{l=1}^{L}$ the sequence of corresponding adjacency matrices.
- Degree of vertex $i \in V_l$: $k_i^{(l)} = \sum_{j=1}^{N_l} A_{ij}^{(l)}$.
- Degree sum formula : $m_l = \frac{1}{2} \sum_{i=1}^{N} k_i^{(l)}$.
- Coloration of level l: $C_l \in \mathcal{C}_l$ any partition $C_l = \{v_1^{(l)} \dots v_K^{(l)}\}$ of V_l.
- We denote by $\mathrm{map}^{C_l} : V_l \to V_{l+1}$ the mapping of all nodes of V_l in the same community according to C_l into a node of V_{l+1}.



MH Hierarchical algorithm

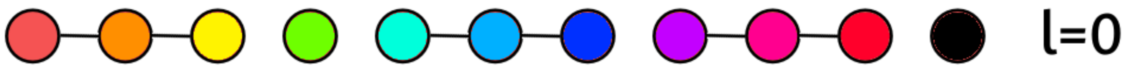
General Hierarchical MH algorithm	
1 :	given $L \geq 1$, a sequence $(G_l^{(0)}, C_l^{(0)})_{l=1}^L$, and $\lambda > 0$
2 :	For $k = 1, \dots, N$:
3 :	Generate a proposal $\tilde{C} \sim p(\cdot \vert (G_l^{(k-1)}, C_l^{(k-1)})_{l=1}^L) \in \mathcal{P}(\otimes_{l=1}^L \mathcal{C}_l)$
4 :	$C^k = \tilde{C}$ with probability $\rho(\lambda, \tilde{C}, C^{k-1})$
5 :	$C^k = C^{k-1}$ otherwise.
6 :	If $C^{k)}$ has been accepted, maintain $(G_l)_{l=1}^L$ with $\mathrm{map}^{\tilde{C}}$.

- The proposal is chosen as follows :

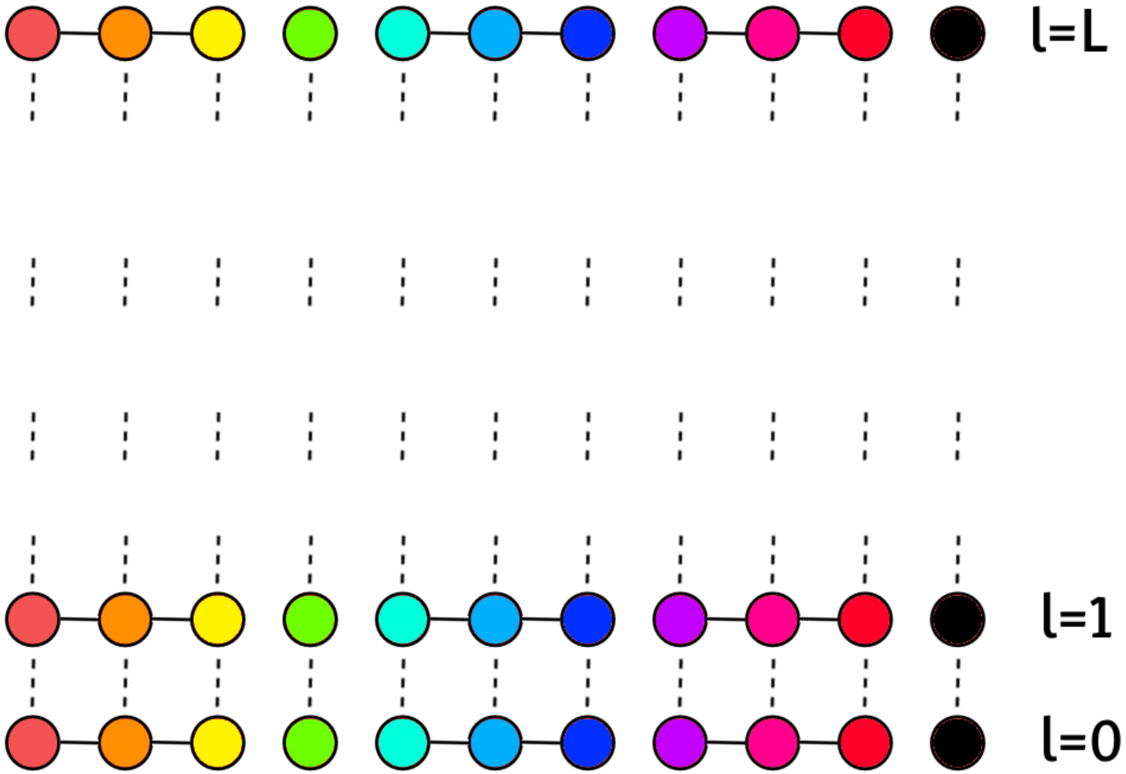
$$p(\tilde{C} \vert (G_l^{(k-1)}, C_l^{(k-1)})_{l=1}^L) := \sum_{l=1}^{L} \alpha_l p_l(\tilde{C}_l \vert G_l, C_l), \forall \tilde{C} = (\tilde{C}_1, \dots, \tilde{C}_L),$$

- where $\sum \alpha_l = 1$ and $p_l(\cdot \vert G_l, C_l)$ is defined on G_l as before.

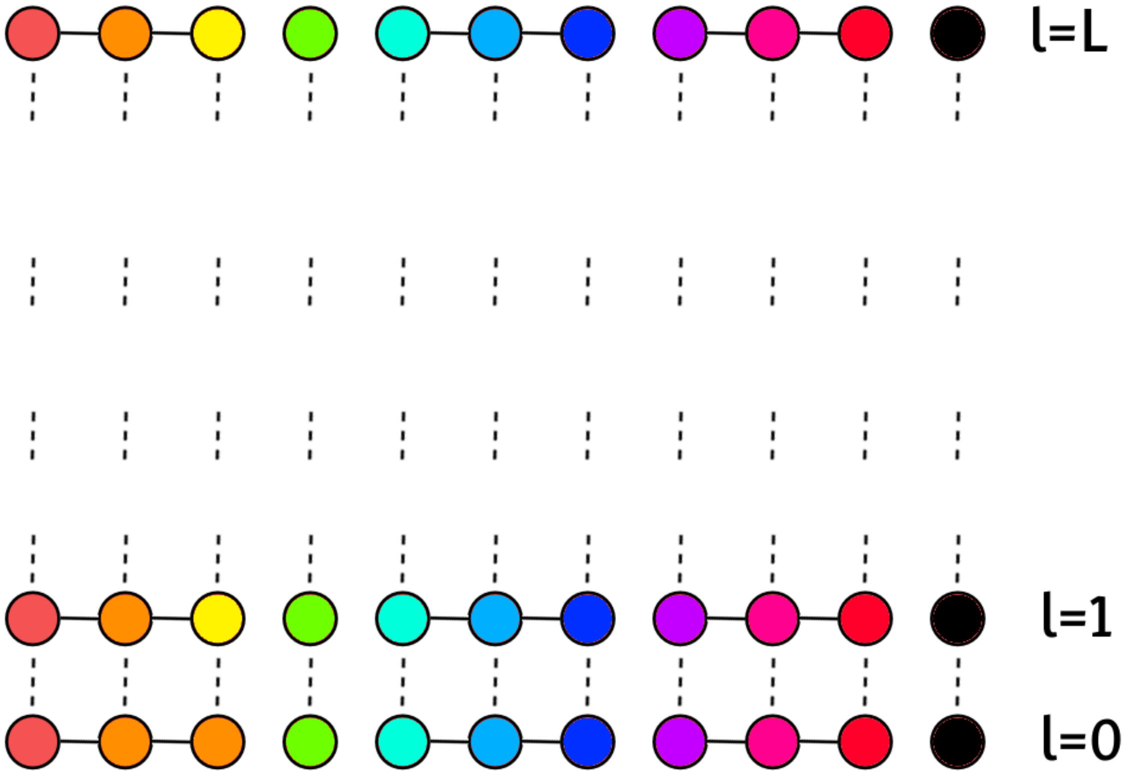
Illustration



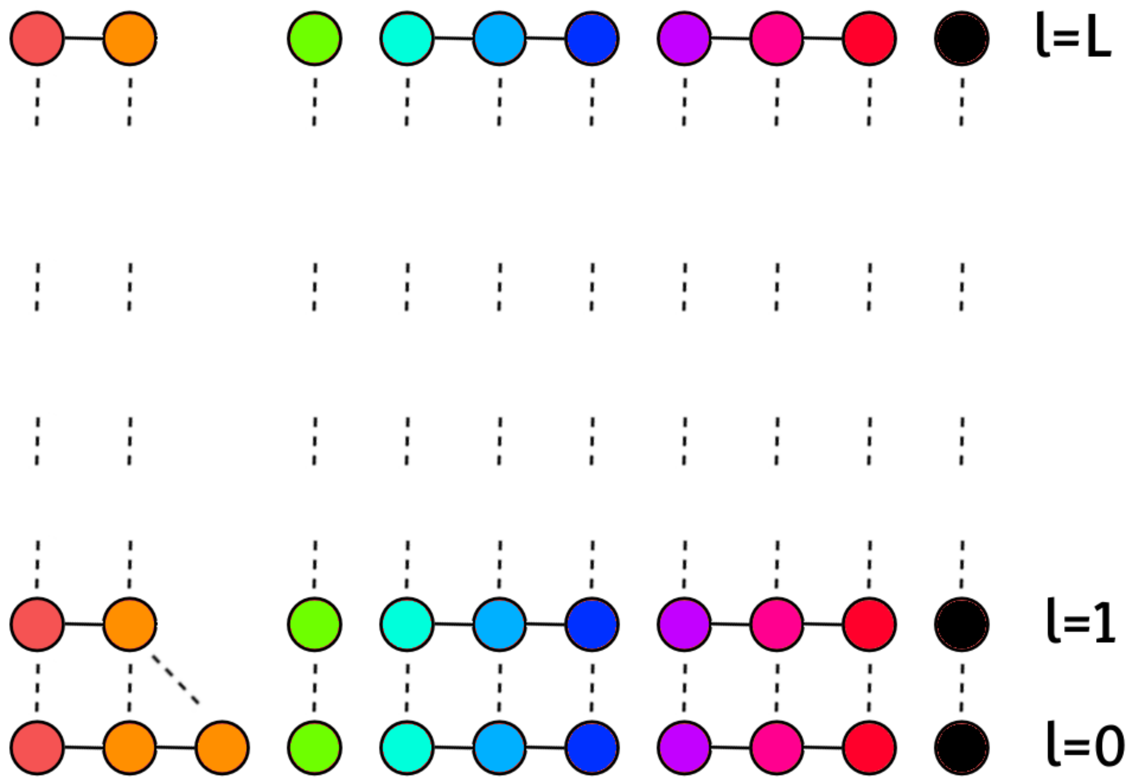
Illustration



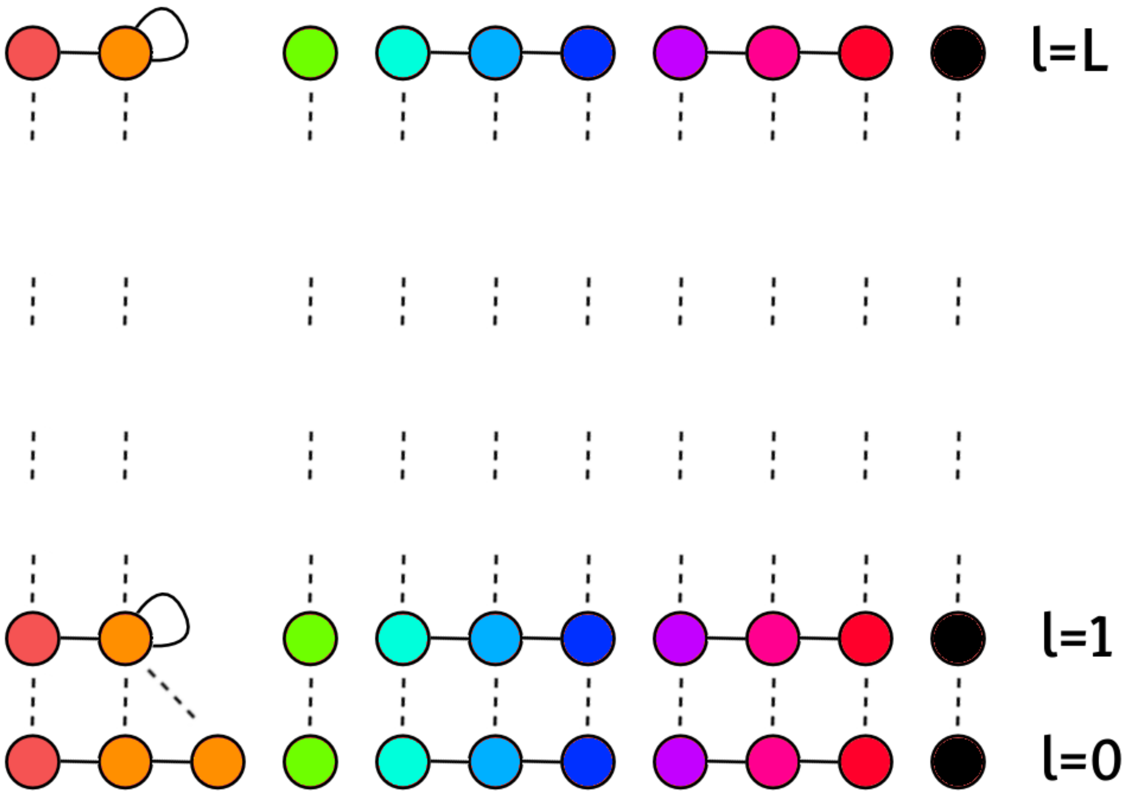
Illustration



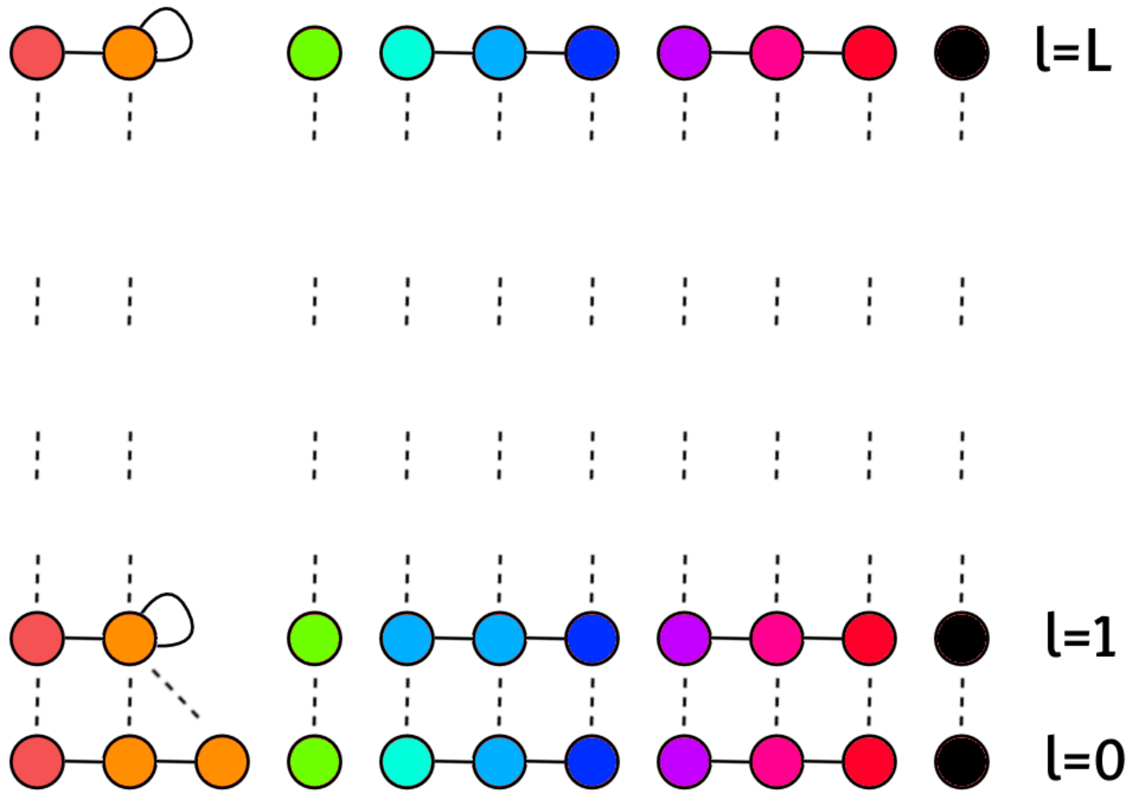
Illustration



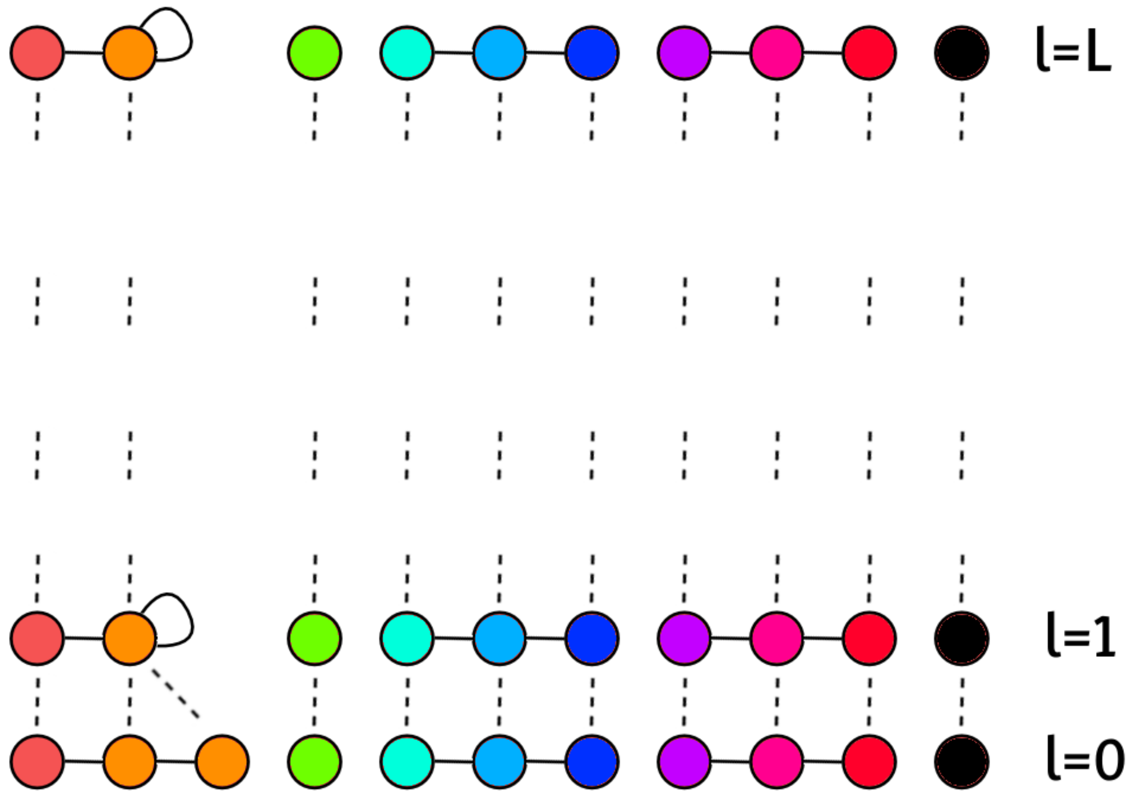
Illustration



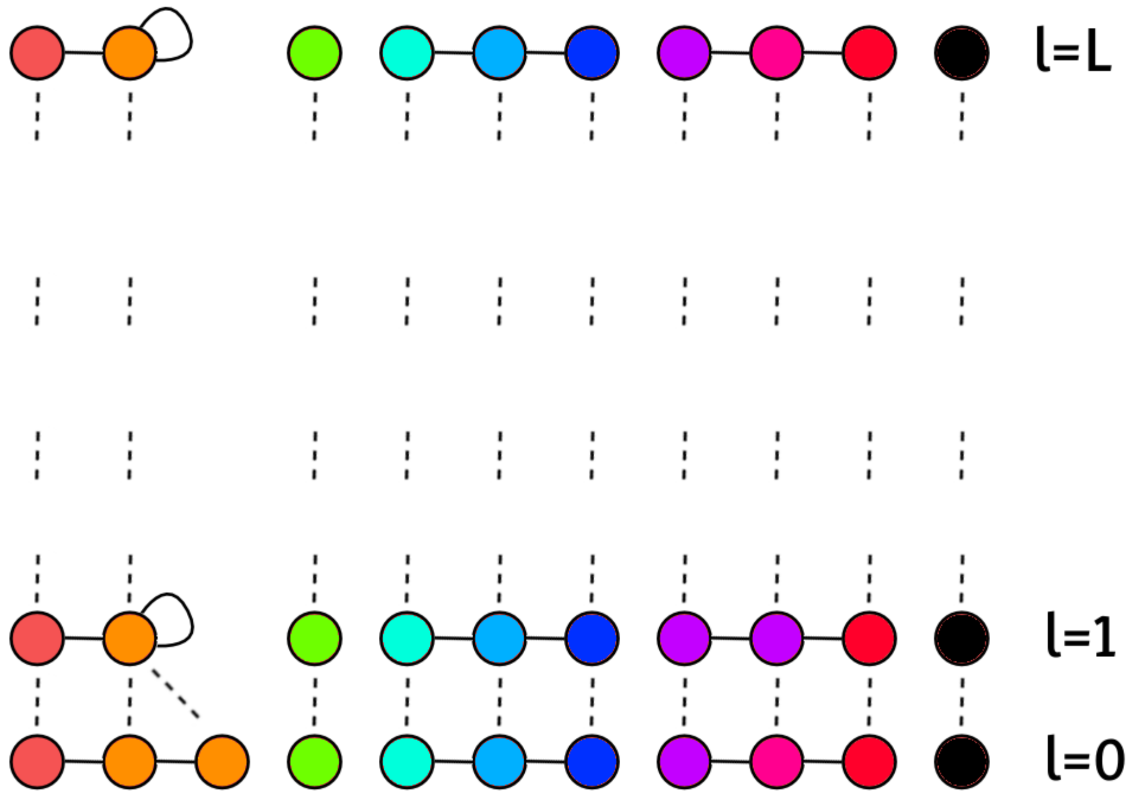
Illustration



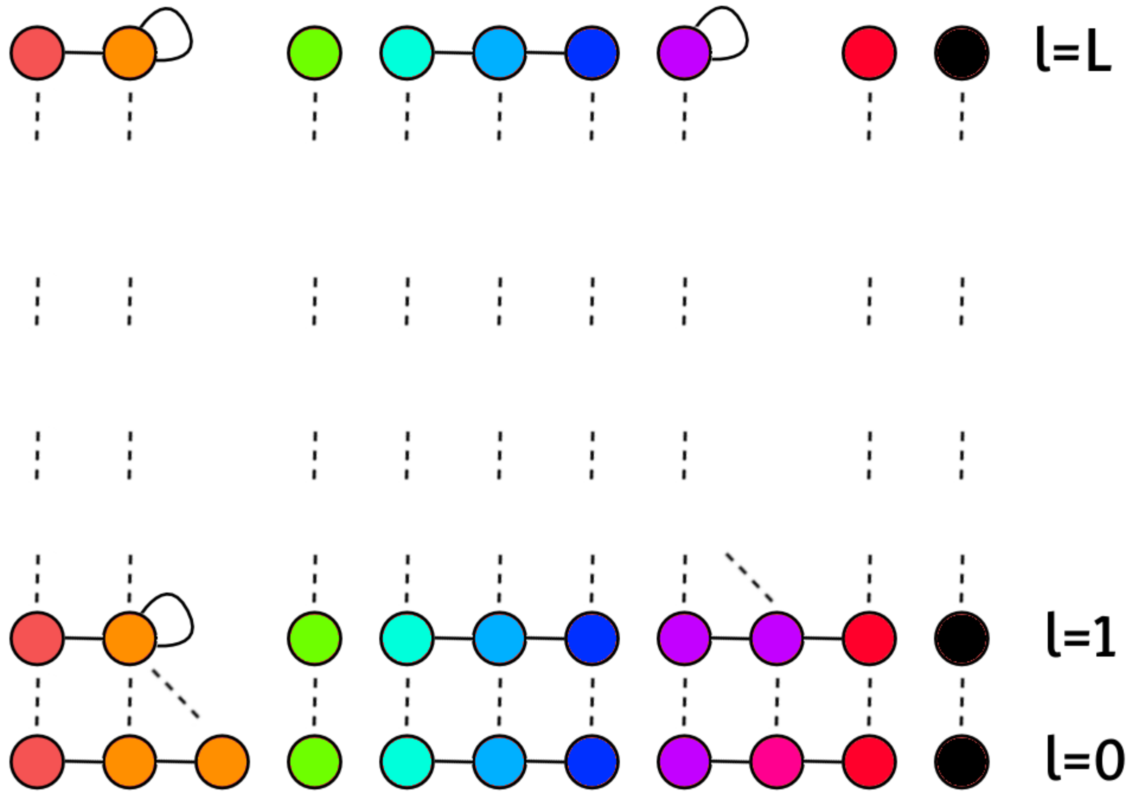
Illustration



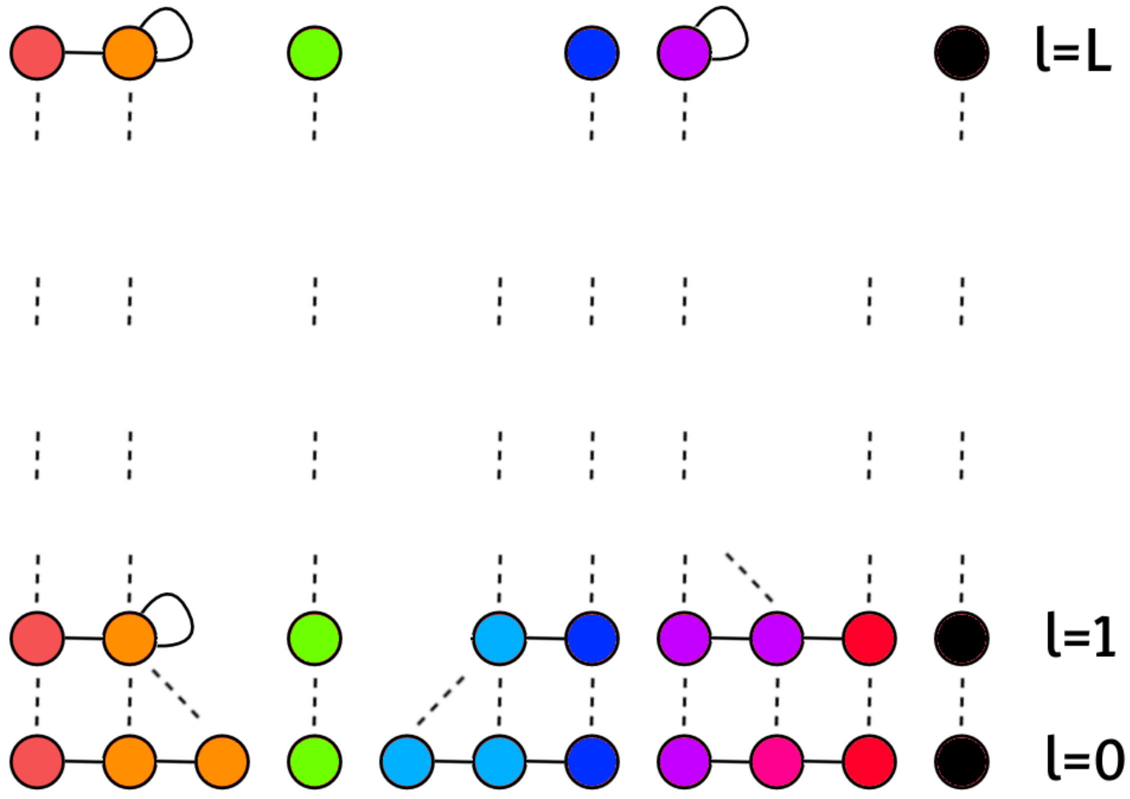
Illustration



Illustration



Illustration



From BATCH to ONLINE

- Hierarchical MCMC $\Leftrightarrow$ Online setting !
- New proposal accepted at level $l \Leftrightarrow$ new observation at level $l+1$
- We just need two boxes :
  - $\forall l = 0, \ldots, L$, *new observation* $\rightarrow$ $\boxed{\text{change\_graph\_l}}$ $\rightarrow$ *modularity change*
  - $\boxed{\text{MCMC iteration}}$ $\rightarrow$ *accept/reject and call* $\boxed{\text{change\_graph\_l}}$.

Illustration using the Stochastic Block Models $\mathrm{SB}(P)$

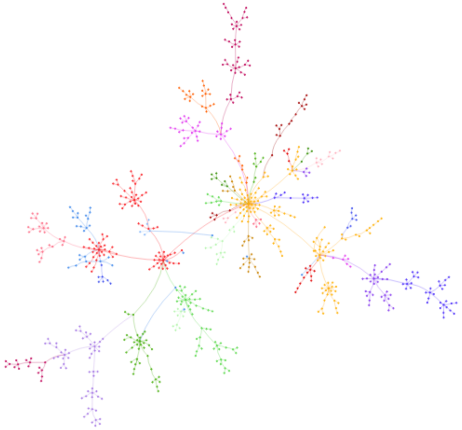
- The Stochastic Block Model (SBM) $SB(n, C, P)$ is defined as follows :
  - $n = |V|$ *is the number of vertices,*
  - C *is a given partition of* V*,*
  - P *is a* $|C| \times |C|$ *matrix of edges probabilities : given a pair of nodes* (i, j)*, the probability to link* i *to* j *is given by* $P_{C(i)C(j)}$.
- SBM are standard models to test community detection algorithms.
- Demo in real time here

Illustration using the Preferential attachment model $\mathrm{PA}(n)$

- The Preferential attachment model $PA(n)$ is defined by induction as follows :
  - $PA(2)$ *is the unique tree on two vertices,*
  - $PA(n+1)$ *is given from* $PA(n)$ *by adding a new node* j *and an edge* $\{i,j\}$ *according to the following probability distribution:*

$$\mathbb{P}(\{i,j\}|PA(n)) = \frac{k_i}{2(n-1)}.$$

- $PA(n)$ have the "Small world" and "scale free" properties.



- Demo in real time here

Results over Digg dataset

- Reply network of the social news website Digg,
- Timestamps, 279,630 vertices (users) and 1,731,653 edges (votes),
- Details: Friendship graph of Digg
- Demo in real time here

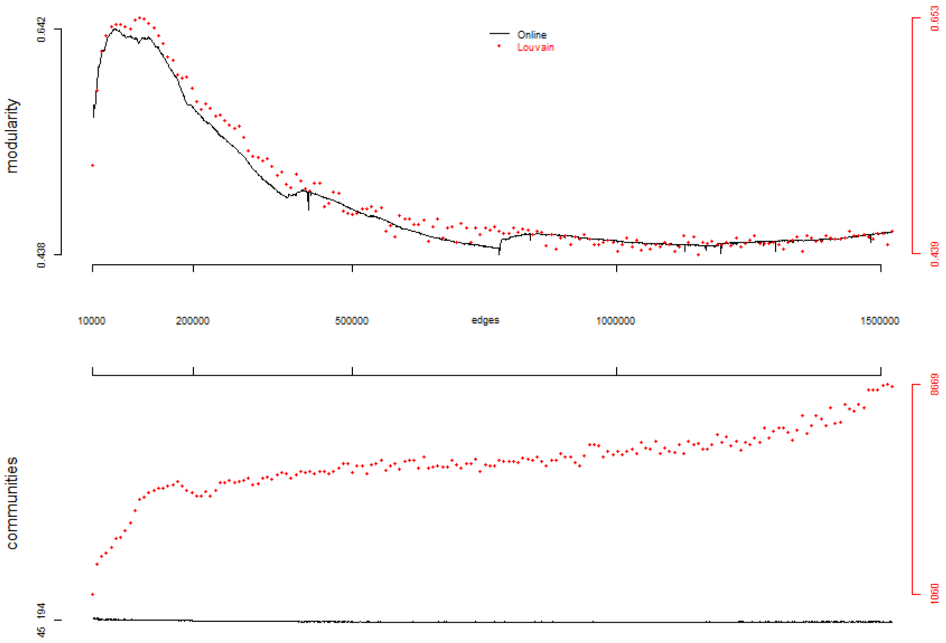
Results over Enron dataset

- Email network of Enron board before and during Enron scandal,
- 5000 mails, 1.5M available fy FERC
- Timestamps, 184 vertices (Kenneth Lay, Jeffrey Skilling, Luise Kitchen, ...)
- 117,977 emails,
- Details: FERC Enron Email Dataset
- Demo in real time here

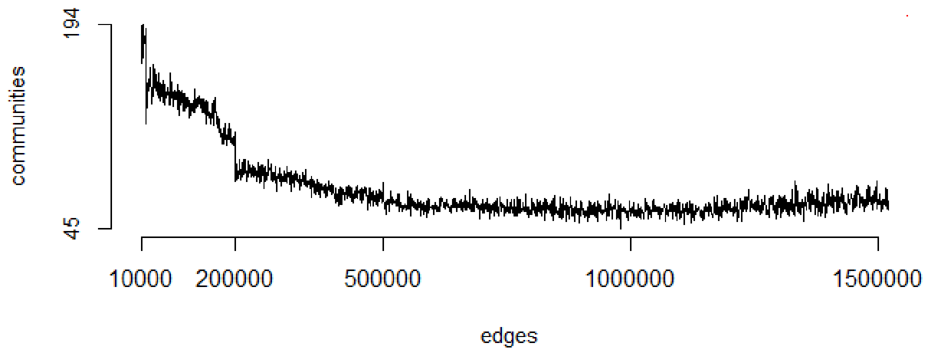
Ticketing example

- Sales activity of 48 stores accross Germany (datetime, product_id, quantity, shop_id),
- 1,654,754 lines from January, 2th to June, 30th,
- Construction of shop vectors and scalar products (kernels),
- Construction of adjacency matrices with suitable ϵ-threshold,
- Demo in real time here

MH vs Louvain for Digg



Communities



Regret bounds

- We expect regret bounds as follows:

$$\sum_{t=1}^{T} \mathbb{E} Q_{G^{(t)}}^{\hat{C}_t} \leq \inf_{C \in \mathcal{C}} \left\{ \sum_{t=1}^{T} Q_{G^{(t)}}^{C} \right\} + r_\lambda(T),$$

  where :

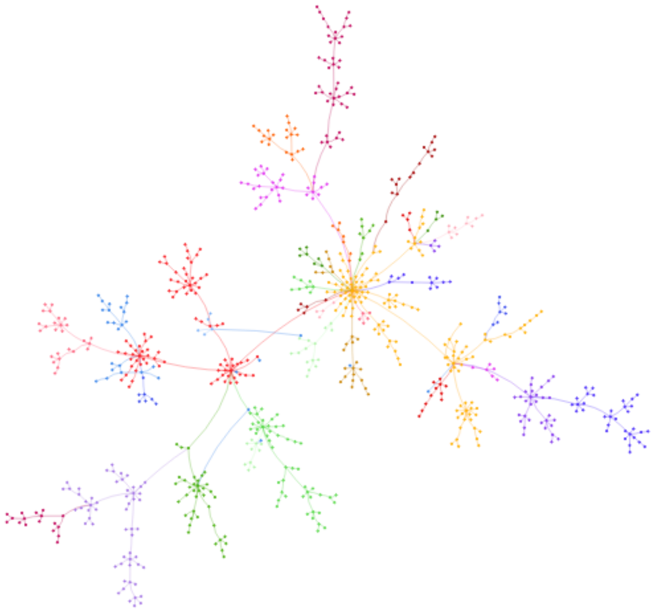
  - $\mathbb{E}$ *is the expectation with respect to the Gibbs posterior* $\hat{p}_{t+1} \sim m_\lambda(\cdot)$*,*
  - $r(\lambda, t) > 0$ *is the regret, a function of inverse temperature and time that shoud be as small as possible and, in particular, sublinear in* T*.*
  - *Adaptative temperature* $\lambda(t)$ *are possible !*

- *Jean-Yves Audibert.*
  Fast learning rates in statistical inference through aggregation.
  Annals of Statistics, 2009.

- *Sébastien Gerchinovitz.*
  Sparsity regret bounds for individual sequences in online linear regression.
  Journal of Machine Learning Research 14 (2011) 729-769.

- *Le Li, Benjamin Guedj and Sébastien Loustau.*
  Quasi-Bayesian Online Clustering.
  Submitted to JMLR, 2016.

-- Thank you for your attention ! --



mail : artfact64@gmail.com
webpage : www.artfact-online.fr/algorithms
Twitter : @seb_loustau or @artfact-lab
Youtube : Le Machine Learning par l'exemple
Meetup : Pau Rennes