1 Graphes aléatoires

1.1 Loi de Poisson

Cette catégorie de modèle effectue très peu d'hypothèses sur les graphes à générer et permet d'atteindre une grande quantité de graphes d'une taille donnée. Ces modèles possèdent deux paramètres. Le premier paramètre indique le nombre de nœuds du graphe à générer. Le deuxième paramètre porte sur les arrêtes du graphe ; il n'indique pas nécessairement le nombre d'arrêtes générées mais indique au moins leur probabilité d'occurence.

1.1.1 Modèle de Gilbert

Pour $n \in \mathbb{N}^*$ et $0 < p < 1$, le modèle de Gilbert, génère un graphe composé de n nœuds dont chaque arrête potentielle a une probabilité p d'être présente, indépendamment des autres arrêtes. En général, on choisit p en fonction de n et typiquement $p(n) = o(1)$. À noter que dans cette méthode, on ne peut connaître à l'avance le nombre d'arrêtes du graphe généré. Grâce à l'analyse de ce type fournie par [NSW01], on sait que la probabilité p_k qu'un nœud pris au hasard soit de degré k vaut :

$$p_k = \binom{|V|}{k} p^k (1-p)^{N-k}.$$

En notant $z = \mathbb{E}\left[|E|\right] = |V|p$, on a le résultat suivant :

$$p_k \underset{N \to +\infty}{\sim} \frac{z^k e^{-z}}{k!},$$

ce qui indique que ce modèle tend asymptotiquement vers une distribution des degrés suivant une loi de Poisson.

1.1.2 Modèle d'Erdös-Rényi

2 Méthodes MCMC

Soit $G = (V, E)$ un graphe non orienté. Pour $i \in V$, on note d_i le degré du nœud i. On note également $D^{(1)}$ la distribution des degrés du graphe. Autrement dit, pour $n \in \mathbb{N}$,

$$D^{(1)}(n) = \frac{|\{i \in V, d_i = n\}|}{|V|}.$$

Enfin, on note $D^{(2)}$ la distribution jointe des degrés, définie pour $(n,p) \in \mathbb{N} \times \mathbb{N}$ par :

$$D^{(2)}(n,p) = \frac{|\left\{(i,j) \in E, (d_i, d_j) = (n,p) \text{ ou } (d_i, d_j) = (p,n)\right\}|}{|E|}.$$

Les méthodes MCMC pour la génération de graphes sont très vastement utilisées ([RJB96]). Leur principe général est de partir d'un graphe existant et de modifier aléatoirement ses arrêtes, tout en conservant certaines propriétés du graphe. L'Algorithme 1 sélectionne aléatoirement deux arrêtes, puis les échange si cela n'entraine ni la création de boucle ni de double arrête.

Algorithm 1 Algorithme MCMC conservant les degrés des nœuds.

Require: V, E, N_{max}

$E^{(0)} \leftarrow E$
$i \leftarrow 1$
while $i \leq N_{max}$ **do**
  $(e_1, e_2) \leftarrow \text{rand}(E^{(i-1)})$
  $(f_1, f_2) \leftarrow \text{rand}(E^{(i-1)})$
  if no loop and no double edge created **then**
    $E^{(i)} \leftarrow E^{(i-1)} \setminus \{(e_1, e_2), (f_1, f_2)\}$
    $E^{(i)} \leftarrow E^{(i)} \cup \{(e_1, f_2), (f_1, e_2)\}$
  else
    $E^{(i)} \leftarrow E^{(i-1)}$
  end if
  $i \leftarrow i + 1$
end while

Cet algorithme est simple à implémenter et ne nécessite qu'un paramètre : le nombre d'itérations N_{max}. En général, on choisit N_{max} de l'ordre de $100 \times |E|$ pour s'assurer un graphe généré indépendant de l'original. Cependant, ce choix est purement expérimental et il existe peu de travaux théoriques sur ces critères ; [RPS12] présente toutefois des résultats intéressants sur des critères d'arrêts spécifiques.

Algorithm 2 Algorithme MCMC pour la génération de graphe.

Require: V, E, N_{max}

$E^{(0)} \leftarrow E$
$i \leftarrow 1$
while $i \leq N_{max}$ **do**
  $(e_1, e_2) \leftarrow \text{rand}(E^{(i-1)})$
  $(f_1, f_2) \leftarrow \text{rand}(\{(u, v) \in E^{(i-1)}, d_u = d_{e_1}\})$
  $E^{(i)} \leftarrow E^{(i-1)} \setminus \{(e_1, e_2), (f_1, f_2)\}$
  $E^{(i)} \leftarrow E^{(i)} \cup \{(e_1, f_2), (f_1, e_2)\}$
  $i \leftarrow i + 1$
end while
return $(V, E^{(N_{max})})$

Références

[NSW01] Mark EJ Newman, Steven H Strogatz, and Duncan J Watts. Random graphs with arbitrary degree distributions and their applications. *Physical Review E*, 64(2) :026118, 2001.

[RJB96] A Ramachandra Rao, Rabindranath Jana, and Suraj Bandyopadhyay. A markov chain monte carlo method for generating random (0, 1)-matrices with given marginals. *Sankhyā : The Indian Journal of Statistics, Series A*, pages 225–242, 1996.

[RPS12] Jaideep Ray, Ali Pinar, and C Seshadhri. Are we there yet ? when to stop a markov chain while generating random graphs. In *Algorithms and Models for the Web Graph*, pages 153–164. Springer, 2012.