Projet Cassiopée

Compte-Rendu

Processus Ponctuels Déterminantaux

Zacharie BUISSON, Théo GIGANT et Eric TROUCHE

Encadrés par
François DESBOUVRIES & Marc CASTELLA

10 juin 2020

Table des matières

Chapitre 1

Processus Ponctuels Déterminantaux

1.1 Introduction

1.1.1 Processus ponctuels :

On commence par se placer dans le cadre discret. On pose donc un espace discret que l'on note $\mathcal{Y}$=$\{1,..,N\}$, avec $N \in \mathbb{N}^*$

Definition 1 *Un processus $\mathbb{P}$ est dit ponctuel si $\mathbb{P}$ est une mesure de probabilité sur $\mathcal{P}(\mathcal{Y})$, avec $\mathcal{P}(\mathcal{Y})$ l'ensemble de tous les sous-ensembles de $\mathcal{Y}$.*

On rappelle qu'une mesure sur une tribu $\mathcal{P}(\mathcal{Y})$ est une application $\mu : \mathcal{P}(\mathcal{Y}) \to [0\,;\infty]$ telle que :
— $\mu(\emptyset) = 0$
— $\forall$ famille $[A_n]_{n\in\mathbb{N}}$ finie ou dénombrable d'éléments de $\mathcal{P}(\mathcal{Y})$ 2 à 2 disjoints, on a : $\mu(\bigcup_{i=1}^{\infty} A_i)=\sum_{i=1}^{\infty} \mu(A_i)$

Rq : On remarque que $\mathcal{P}(\mathcal{Y})$ est une tribu car on a un ensemble discret. (on appelle cette tribu la tribu discrète).

1.1.2 Processus ponctuel déterminantal

Commençons par introduire les éléments qui nous serviront à définir nos processus.

Definition 2 *Soit K une matrice semie-définie positive telle que $K \preceq I$, c'est à dire que pour $\lambda_1, .., \lambda_N$ les N valeurs propres de K, on a $\forall i \in [\![1, N]\!], 0 \leq \lambda_i \leq 1$.*
K est indexé sur $\mathcal{Y}$, c'est à dire K=$[K_{i,j}]_{(i,j)\in\mathcal{Y}^2}$. On nomme K la matrice de noyau marginale. De plus, on note K_A la restriction de K à l'ensemble A, K_A=$[K_{i,j}]_{(i,j)\in A}$

1.1.3 Une première construction :

On peut désormais définir les processus ponctuels déterminantaux.
Soit $Y \in \mathcal{P}(\mathcal{Y})$.

Definition 3 *: Un processus ponctuel $\mathbb{P}$ est dit determinantal si il existe une matrice K, appelée matrice de noyau marginale (ou parfois appelée matrice de corrélation), de taille card($\mathcal{Y}$) $\times$ card($\mathcal{Y}$), indexée sur les éléments de $\mathcal{Y}$, tel que la fonction de corrélation $\rho(\mathcal{Y}) = \mathbb{P}(X \in \mathcal{P}(\mathcal{Y})|Y \subset X), Y \subset \mathcal{Y}$ soit égale au déterminant de K_Y, avec K_Y désignant la restriction de K aux éléments de Y. On a donc :*

$$\forall Y \subset \mathcal{Y}, \mathbb{P}(X \subset \mathcal{P}(\mathcal{Y})|Y \subset X) = \det(K_Y)$$

A partir des cas les plus triviaux, c'est à dire avec les espaces A les plus petits possibles, on peut se permettre quelques observations :

— Si $A = \{i\}$, alors on a $\mathbb{P}(i \in Y) = K_{i,i}$

— Si $A = \{i, j\}$, $\mathbb{P}(\ i, j \in Y^2) = K_{i,i}K_{j,j}$ - $K_{i,j}K_{j,i}$, c'est à dire $\mathbb{P}(\ i, j \in Y^2) = \mathbb{P}(i \in Y)\mathbb{P}(j \in Y)$ - $K_{i,j}^2$

En fait, pour le cas où l'on a la dimension 2, on peut écrire cela d'une autre manière :

$$\mathbb{P}(i, j \in Y^2) = K_{i,i}K_{j,j} - K_{i,j}K_{j,i} \tag{1.1}$$

$$= K_{i,i}K_{j,j} \underbrace{[1 - \frac{K_{i,j}^2}{K_{i,i}K_{j,j}}]}_{Auto-correlation} \tag{1.2}$$

$$Et \quad \mathbb{P}(i, j) < K_{i,i}K_{j,j} \tag{1.3}$$

On voit que l'on a un terme d'auto-corrélation qui est compris entre 0 et 1, et celui-ci permet de comprendre, sur le cas d'une matrice 2×2, le coté répulsif de la méthode. Notons deux cas particulier à ce cas.

K est diagonale

Si la matrice K est diagonale, on peut voir que le coefficient d'auto-corrélation vaut 1. Ainsi, les éléments de $\mathcal{Y}$ sont tous indépendants, et ceci conduit à une répartition complètement aléatoire.

Auto-corrélation nulle

Dans ce cas, on a ici la valeur de $K_{i,j}$=$\sqrt{K_{i,i}K_{j,j}}$. Le coefficient d'auto-corrélation étant nul, on peut voir que tous les éléments sont parfaitement similaires. Ainsi, on peut voir que l'on a une probabilité très faible que deux éléments i et j respectant ces contraintes apparaissent proches presque sûrement.

1.2 Répulsion

1.2.1 Noyau et expulsion probabiliste

Ces modèles sont considérés comme répulsifs car ils capturent les exclusions mutuelles probabilistes entre les tirages par une matrice de noyau, qui déterminent les proximités entre plusieurs éléments.

Reprenons ici les exemples que nous avons sur les matrices de petites tailles : d'une part, on peut voir que l'on a les éléments diagonaux de K, à savoir $\mathbb{P}(y_i \in Y)$ et $\mathbb{P}(y_j \in Y)$, qui donnent les inclusions marginales de chacun des éléments de Y, là où les éléments extérieurs à la diagonale nous donne des critères de corrélation et de corrélation négative des éléments de Y.

1.3 Constructions de DPP

1.3.1 Construction par les L-ensembles

Il existe de nombreuses façons de construire les DPP. On commence par voir le cas où l'on passe par des L-ensembles et des matrices reliées, ce qui apparait comme un choix plus pertinent ici.

Definition 4 *: Un L-ensemble est un processus qui définit une probabilité* $\mathbb{P}_L(Y = y) \propto det(L_Y)$*, avec la matrice* L_Y *qui est une matrice de noyau de vraisemblance.*

Rq : Dans la suite, on notera $\mathbb{P}_L(Y = y)$ comme étant $\mathbb{P}_L(Y)$.

Pourquoi cette construction est intéressante ?

K et L offrent la possibilité de décrire une représentation de DPP, avec $\mathbb{P}$ pour K, avec $\mathbb{P}_L$ pour L. On peut toutefois lier les deux matrices avec la relation :

$$\boxed{K = L(L+I)^{-1}} \tag{1.4}$$

De plus, si on décompose L selon les éléments propres, c'est à dire avec $L = \sum_{n=1}^{N} \lambda_n v_n v_n^T$, on peut décrire la matrice K selon les éléments propres de la matrice L, sous la forme $K = \sum_{n=1}^{N} \frac{\lambda_n}{\lambda_n+1} v_n v_n^T$

Dans la même lignée, et sous réserve d'existence des inverses, on peut écrire $L = K.(I-K)^{-1}$

1.3.2 Quelques théorèmes

Théorème 1 *Pour* $A \subseteq \mathcal{Y}$, $\sum_{A \subseteq Y \subseteq \mathcal{Y}} det(L_Y) = det(L + I_{\bar{A}})$

Demo : Voir Annexe

On peut utiliser le résultat de la question précédente et on obtient :

Definition 5 *: Soit* $Y \in \mathcal{P}(\mathcal{Y})$*. On peut écrire la valeur exacte de* $\mathbb{P}_L(\boldsymbol{Y} = Y)$ *et cela vaut :*

$$\mathbb{P}_L(\boldsymbol{Y} = Y) = \frac{det(L_Y)}{det(L+I)}$$

Rq : On notera que l'on écrira à l'avenir $\mathbb{P}_L$ à la place de $\mathbb{P}_L(\mathbf{Y} = Y)$

Théorème 2 *On peut affirmer qu'un L ensemble est un DPP, et que son noyau marginal vaut :* $K = L(L+I)^{-1} = I - (L+I)^{-1}$

Démo : Voir Annexe

On a posé ici $K = I - (L+I)^{-1}$. On peut également décomposer K grâce à ses vecteurs propres. On a donc :

$$K = \sum_{n=1}^{N} \frac{\lambda_n}{\lambda_n + 1} v_n v_n^T \tag{1.5}$$

1.4 Propriétés sur les DPP

Restriction :

Théorème 3 *: Si* $\boldsymbol{Y}$ *est distribuée selon un DPP de noyau marginal K, alors* $A \cap \boldsymbol{Y}$*, avec* $A \subseteq \mathcal{Y}$ *est également un DPP de noyau marginal* K_A*.*

Complémentaire

Théorème 4 *: Si* $\boldsymbol{Y}$ *est distribuée selon un DPP de noyau marginal K, alors* $\mathcal{Y} \setminus \boldsymbol{Y}$ *est aussi un DPP, et on a* $\mathbb{P}(A \cap \mathcal{Y} = \emptyset) = det(I - K_A)$

Domination

Théorème 5 *: Si* $K \preceq K'$*, telle que* $K' \setminus K$ *est définie semie-positive, alors pour tout* $A \subseteq \mathcal{Y}$*,* $det(K_A) \leq det(K'_A)$

Proportionnalité

Théorème 6 *: Si* $K = \gamma K$*', pour tout* $\gamma \in [0,1]$*, alors pour tout* $A \subseteq \mathcal{Y}$*, on a* $det(K_A) = \gamma^{card(A)} det(K'_A)$

Cardinalité

Théorème 7 *: Soient* $\lambda_1, .., \lambda_N$ *des vecteurs propres. Alors card(*$\boldsymbol{Y}$*) est distribuée comme une somme de tirages de Bernoulli, dans laquelle la probabilité de succès vaut* $\frac{\lambda_n}{\lambda_n+1}$

On peut donc calculer variance et espérance :

$\mathbb{E}[card(\boldsymbol{Y})] = \sum_{n=1}^{N} \frac{\lambda_n}{\lambda_n+1} = trace(K)$ *et* $\mathbb{V}[card(\boldsymbol{Y})] = \sum_{n=1}^{N} \frac{\lambda_n}{(\lambda_n+1)^2}$

Toutes ces propriétés nous donnent des idées sur le fonctionnement des DPP, et sur les applications et les formules qui pourraient en découler.

Chapitre 2

Échantillonnage des DPP

2.1 Références de cette partie

Cette partie s'appuie sur le travail de recherche : **DPP for machine learning** de Alex Kulesza, publié en 2012. Nous avons donc traduit la partie de l'algorithme numéro 1, et on a détaillé la preuve dans cette partie.

2.2 Normalisation

L'intérêt de passer par ces méthodes de DPP est de faire baisser la complexité. Dans les faits, on peut voir que la complexité serait exponentielle (2^N), mais le passage par ces méthodes permet d'étudier des déterminants de matrices $N \times N$, et ceci permet, grâce aux méthodes de calcul de déterminants, d'avoir des complexités en $\mathrm{O}(N^3)$ à la place.

2.3 Marginalisation et Conditionnement

L'étude des probabilités marginales a déjà été réalisée. On a vu certaines propriétés. On peut avoir les matrices K qui sont réduites, par des méthodes d'élimination de redondance. Ceci contribue à diminuer les valeurs.

Toutefois, il est possible de voir ce que cela donne en passant par des probabilités conditionnelles. Celles-ci peuvent apporter une nouvelle vision et de nouvelles informations. Ceci permettra aussi de faire des l'apprentissage, et ceci pourra nous permettre d'avoir un premier algorithme d'échantillonnage.

De surcroit, ces méthodes avec les conditionnement de variables aléatoires permettent une introduction des k-DPP, c'est à dire des DPP avec précisément k éléments tirés.

2.3.1 Introduction des DPP à probabilités conditionnelles : Cas nul

Soient $A, B \in Y^2$, tels que $A \cap B = \emptyset$.

On peut commencer par introduire des probabilités conditionnelles dans un cas simple, où l'intersection de

deux événements est nulle :

$$\mathbb{P}_L(\mathbf{Y} = B | A \cap \mathbf{Y} = \emptyset) = \frac{\mathbb{P}_L(\mathbf{Y} = B)}{\mathbb{P}_L(A \cap \mathbf{Y} = \emptyset)} \tag{2.1}$$

$$= \frac{\det(L_B)}{\sum\limits_{\substack{B' \\ B' \cap A = \emptyset}} \det(L_{B'})} \tag{2.2}$$

$$\boxed{\mathbb{P}_L(\mathbf{Y} = B | A \cap \mathbf{Y} = \emptyset) \ = \frac{\det(L_B)}{\det(L_{\bar{A}} + I)}} \tag{2.3}$$

Dans ce modèle, on note que $L_{\bar{A}}$ est la restriction de L à toutes les lignes et les colonnes indexées par $\mathcal{Y} - I$. Notons que l'on peut calculer le noyau, qui est :

$$\boxed{L^A \ = ([(L + I_{\bar{A}})^{-1}]_{\bar{A}})^{-1} - I} \tag{2.4}$$

2.3.2 Introduction des DPP à probabilités conditionnelles : Cas total

On se place dans le cas où l'on a un DPP sur un ensemble où tous les éléments d'un ensemble A sont observés. Ainsi, on peut écrire que :

$$\mathbb{P}_L(\mathbf{Y} = A \cup B | A \subseteq \mathbf{Y}) = \frac{\mathbb{P}_L(\mathbf{Y} = A \cup B)}{\mathbb{P}_L(A \subseteq \mathbf{Y})} \tag{2.5}$$

$$= \frac{\det(L_{A \cup B})}{\det(L + I_{\bar{A}})} \tag{2.6}$$

Cette fois, on note que la matrice $I_{\bar{A}}$ est une matrices avec des 1 en diagonales, indexée sur les éléments de $\mathcal{Y} \setminus A$, et des 0 ailleurs.

2.3.3 Introduction des DPP à probabilités conditionnelles : Cas général

On peut toutefois écrire le cas général, avec une formule un peu plus complexe. On obtient alors une combinaison d'éléments apparaissant et n'apparaissant pas dans le DPP.

$$\boxed{\mathbb{P}_L(\mathbf{Y} = A^{in} \cup B | A^{in} \subseteq \mathbf{Y}, A^{out} \cap \mathbf{Y} = \emptyset) \ = \frac{\det(L_{A^{in} \cup B})}{\det(L_{\bar{A}^{out}} + I_{\bar{A}^{in}})}} \tag{2.7}$$

Ces trois formules nous permettent de définir la formule de loi de probabilité marginale dans tous les cas. En effet, on peut trouver que dans le cas général, en appliquant la formule décrivant L^A (1.20) avec la définition de K (1.4), on peut obtenir la formule du noyau marginal pour un DPP donnant l'apparence de A :

$$\boxed{K^A \ = I - [(L + I_{\bar{A}})^{-1}]_{\bar{A}}} \tag{2.8}$$

Ceci permet d'écrire la formule qui nous intéresse le plus :

$$\boxed{\mathbb{P}(B \subseteq \mathbf{Y} | A \subseteq \mathbf{Y}) \ = \det(K_B^A)} \tag{2.9}$$

On peut donc calculer la valeur de nos probabilités conditionnelles, quand on a la matrice K, qu'on l'indexe selon des éléments de $\mathcal{Y} - I$, et dont on prend la restriction selon B. (Quand A et B sont disjoints).

Maintenant, en utilisant la formule que l'on a écrite dans le théorème 4, on peut écrire la probabilité marginale de n'importe quel ensemble, et obtenir la formule suivante :

$$\mathbb{P}(A \subseteq \mathbf{Y}, B \cup \mathbf{Y} = \emptyset) = \mathbb{P}(A \subseteq \mathbf{Y})\mathbb{P}(B \cup \mathbf{Y} = \emptyset | A \subseteq \mathbf{Y}) \tag{2.10}$$

$$\boxed{\mathbb{P}(A \subseteq \mathbf{Y}, B \cup \mathbf{Y} = \emptyset) \ = \det(K_A) \det(I - K_B^A)} \tag{2.11}$$

Grâce à cette formule, nous allons pouvoir commencer à obtenir différents algorithmes d'échantillonnage.

2.4 Échantillonnage

2.4.1 L'algorithme de la décomposition

Maintenant que l'on a ces formules, on peut commencer à réaliser les premiers algorithmes.

Un algorithme d'échantillonnage assez performant, pour échantillonner une configuration $Y \in \mathcal{Y}$ d'un DPP. On peut donner une programmation en pseudo-code :

Algorithm 1 Échantillonnage à partir d'un DPP : Méthode spectrale

Entrée : décomposition en vecteurs propres $\{(v_n, \lambda_n)\}_{n=1}^N$ de L
$J \leftarrow \emptyset$
for $n = 1, 2, .., N$ **do**
  $J \leftarrow J \cup \{n\}$ *avec une probabilite* $\frac{\lambda_n}{\lambda_n + 1}$
end for
$V \leftarrow \{v_n\}_{n \in J}$
$Y \leftarrow \emptyset$
while $card(V) > 0$ **do**
  Choisir i dans $\mathcal{Y}$ avec Pr(i)= $\frac{1}{card(V)} \sum_{v \in V} (v^T e_i)^2$
  $Y \leftarrow Y \cup i$
  $V \leftarrow V_\perp$ une base orthonormée pour un sous espace de V orthogonal à e_i
end while
Sortie : Y

Notons ici que le $e_i \in \mathcal{M}_{N \times 1}(\mathbb{R})$ telle que :

$$e_i = \begin{pmatrix} 0 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad \textit{avec le } 1 \textit{ a la position } i$$

.

2.4.2 Intérêt de cet algorithme :

Théorème 8 *Soit $L = \sum_{n=1}^{N} \lambda_n v_n v_n^T$ une décomposition en vecteurs propres d'une matrice semie-définie positive L. Alors l'algorithme 1 échantillonne $\boldsymbol{Y} \sim \mathbb{P}_L$*

2.4.3 Préliminaires de démonstration : DPP élémentaires

On peut déjà montrer que l'on peut écrire un DPP en fonction de DPP élémentaire. Il faut pour cela définir ce que l'on appelle un DPP élémentaire et ensuite voir les impacts de la boucle 1 et 2.

Definition 6 *: Un DPP est dit élémentaire si chaque valeur propre du noyau marginal est dans {0,1}. On peut donc noter $\mathbb{P}^V$, où V est un ensemble de vecteurs orthonormaux, et cela correspond à un DPP avec un noyau marginal $K^V = \sum_{v \in V} vv^T$*
K^V est aussi la matrice de projection orthonormale au sous-espace vectoriel engendré par les vecteurs de V.

2.4.4 Lemme : Développement de déterminant

Lemma 1 *: Soit $n = 1, 2, .., N$, soit W_n une séquence arbitraire de matrices $k \times k$ de rang 1, et soit $(W_n)_i$ qui dénote la ième colonne de W_n. On pose $W_J = \sum_{n \in J} W_n$, avec $J \subseteq \mathcal{P}(\{1, .., N\})$. On peut donc écrire :*

$$\det(W_J) = \sum_{\substack{n_1, n_2, \ldots, n_k \in J \\ distinct}} \det([(W_{n_1})_1, (W_{n_2})_2, .., (W_{n_k})_k])$$

où $(W_{n_i})_i$ désigne la i-ème colonne de la matrice W_{n_i}

Demo : Voir Annexe

2.4.5 Lemme : Décomposition de DPP en DPP élémentaire

Lemma 2 *: Un DPP de noyau $L = \sum_{n=1}^{N} \lambda_n v_n v_n^T$ est un mélange de DPP élémentaires tels que :*

$$\mathbb{P}_L = \frac{1}{\det(L+I)} \sum_{J \subseteq \{1,2,\ldots,N\}} \mathbb{P}^{V_J} \prod_{n \in J} \lambda_n$$

où $\mathbb{P}^{V_J} = \{v_n\}_{n \in J}$.

Demo : Voir Annexe

2.4.6 Lemme : Cardinaux des DPP élémentaires

Lemma 3 *Si* $\boldsymbol{Y}$ *suit un DPP élémentaire* $\mathbb{P}^V$*, alors on a* $card(\boldsymbol{Y}) = card(V)$ *avec une probabilité 1.*

Demo : Voir Annexe

2.4.7 Démonstration du théorème 8

Déjà, on peut voir que l'algorithme est décomposé en deux boucles. Dans la première, on sélectionne au hasard un groupe de vecteurs propres, et dans la seconde, un échantillon Y est produit sur la base de ces vecteurs. On remarque qu'à chaque itérations, le cardinal de Y augmente de 1, lorsque celui de V diminue de 1.

Soit $L = \sum_{n=1}^{N} \lambda_n v_n v_n^T$ une matrice de noyau d'un DPP et A un sous-ensemble de $\mathcal{Y}$ de cardinal k. Soit $Y \in \mathcal{P}(\mathcal{Y})$, avec les définitions précédentes.

Calculons dans un premier temps la valeur de $\mathbb{E}[\mathbb{P}^{V_J}] = \sum_{J \subseteq \{1,\dots,N\}} (\mathbb{P}^{V_J}(A \subseteq \mathcal{Y}) \times \mathbb{P}_r(J))$, avec $\mathbb{P}_r(J)$ qui correspond au tirage de J selon une loi de Bernoulli, où chaque élément n a une probabilité $\frac{\lambda_n}{\lambda_n+1}$ d'appartenir à J. On montre alors :

Lemma 4 *: Si on tire n'importe quel* $J \subseteq \mathcal{Y}$ *selon une loi de Bernoulli, de paramètre* $\frac{\lambda_n}{\lambda_n+1}$*, alors :*$\forall A \subseteq Y, \mathbb{E}[\mathbb{P}^{V_J}] = \mathbb{P}_L(A \subseteq Y)$

Demo : Voir Annexe

En particulier, dans notre algorithme, il est clair que la première boucle tire un échantillon $J \subseteq \mathcal{Y}$ avec une loi de Bernoulli de paramètre $\frac{\lambda_n}{\lambda_n+1}$. Cela prouve donc la première partie de l'algorithme.

Lemma 5 *La deuxième boucle de l'algorithme échantillonne bien un élément* Y *de* $\mathcal{Y}$ *avec une loi de probabilité* $\mathbb{P}^{V_J}$

Demo : Voir Annexe

On a montré que la boucle une de l'algorithme faisait le tirage selon une loi de Bernoulli de probabilité voulue, et que la deuxième boucle échantillonne en réalité un élément Y de $\mathcal{Y}$ avec une loi de probabilité $\mathbb{P}^{V_J}$: on vient de prouver la théorème.

2.5 Machine Learning et DPP

2.5.1 Introduction du Machine Learning au DPP

Dans cette partien nous allons voir comment construire un noyau de DPP avec des méthodes de machine-learning.

Tout d'abord, pour une meilleure flexibilité et des algorithmes qui s'adaptent mieux à différentes situations, nous allons conditionner l'ensemble duquel on tire un DPP $\mathcal{Y}$ par rapport à un entrée X issu d'un ensemble de départ $\mathcal{X}$. Nous allons donc tirer selon un DPP un ensemble Y inclu dans $\mathcal{Y}(X)$.

Les données d'entrainement et de test auront donc la forme suivante : $(X^{(t)}, Y^{(t)})_{t=1}^{T}$, tirés de façon identique et indépendante selon une distribution D de l'espace $\mathcal{X} \times \mathcal{P}(\mathcal{Y}(X))$

On définit alors la probabilité conditionnelle de tirer le sous Y selon l'entrée X par rapport au noyau $L(X) \in \mathcal{M}(\mathbb{R}^+)$ de dimension $card(\mathcal{Y}(X))^2$, comme étant :

$$\boxed{\mathbb{P}(Y|X) \ = \frac{\det\left(L_Y(X)\right)}{\det\left(L(X)+I\right)}} \tag{2.12}$$

On peut décomposer $L(X)$ comme une matrice de Gram de la forme :

$$\boxed{L_{i,j}(X) \ = q_i(X)\Phi_i(X)^T\Phi_j(X)q_j(X)} \tag{2.13}$$

Dans lequel on aurait $||\Phi_i(X)|| = 1$ et $q_i(X) \geq 0$. En posant $S_{i,j} = \Phi_i(X)^T\Phi_j(X)$, on obtient $L_{i,j} = q_i(X)S_{i,j}(X)q_j(X)$. On rappelle que les q_i et les q_j sont les facteurs de qualité, et que $S_{i,j}$ est le facteur de diversité.

Comme nous sommes dans un problème d'apprentissage supervisé, nous allons supposer que le noyau conditionnel $L(X, \theta)$ est paramétré par un vecteur θ, et on a donc :

$$\mathbb{P}_\theta(Y|X) = \frac{\det\left(L_Y(X,\theta)\right)}{\det\left(L(X,\theta)+I\right)} \tag{2.14}$$

2.5.2 Calcul de la log-vraissemblance

A ce stade, il faut définir la log-vraissemblance de notre algorithme Tout l'enjeu de l'apprentissage supervisé est de maximiser cette log-vraissemblance

O note $\mathcal{L}$ la fonction log-vrassemblance, et on a donc :

$$\mathcal{L}(\theta) = \log\left(\prod_{t=1}^{T}\mathbb{P}_\theta(Y^{(t)}|X^{(t)})\right) \tag{2.15}$$

$$= \sum_{t=1}^{T}\log\left(\mathbb{P}_\theta(Y^{(t)}|X^{(t)})\right) \tag{2.16}$$

$$= \sum_{t=1}^{T}\left[\log\left(\det\left(L_{Y^{(t)}}(X^{(t)},\theta)\right)\right) - \log\left(\det\left(L(X^{(t)},\theta)+I\right)\right)\right] \tag{2.17}$$

2.5.3 Minimisation de la log vraissemnblance

Pour optimiser la log-vraissemblance, nous utilisons l'algorithme de descente de gradient. Nous calculerons donc $\nabla\mathcal{L}$, c'est-à-dire le gradient de $\mathcal{L}$, et dont l'existence est avérée ici (différentiabilité de fonctions usuelles). Or ici, on sait que θ converge vers l'optimum si la fonction $\mathcal{L}$ est concave par rapport à θ.

Dans cette partie ici, nous allons supposer que seul la qualité est paramétrée par θ. On a donc :

$$\boxed{L_{i,j}(X,\theta) \ = q_i(X,\theta)S_{i,j}(X)q_j(X,\theta)} \tag{2.18}$$

De plus, on optimise les facteurs de qualité en utilisant un modèle log-linéaire, soit avec $q_i(X,\theta) = exp\left(\frac{1}{2}\theta^t f_i(X)\right)$, dans lequel on utilise $\theta \in \mathbb{R}^n$, qui est le paramètre à optimiser et $f_i(X)$ un vecteur caractéristique de l'élement i en fonction de l'entrée X. Enfin, pour simplifier les notations, on suppose que T=1.

On a alors :

$$\mathbb{P}_\theta(Y|X) = \frac{\det\left(L_Y(X,\theta)\right)}{\det\left(L(X,\theta)+I\right)} \tag{2.19}$$

$$= \frac{\det\left(L_Y(X,\theta)\right)}{\sum\limits_{Y'\subseteq\mathcal{Y}(X)}\det\left(L_{Y'}(X,\theta)\right)} \tag{2.20}$$

$$= \frac{\prod\limits_{i\in Y} q_i(X,\theta)^2\det\left(S_Y\right)}{\sum\limits_{Y'\subseteq\mathcal{Y}(X)}\prod\limits_{i\in Y'} q_i(X,\theta)^2\det(S_{Y'})} \tag{2.21}$$

$$= \frac{\prod\limits_{i\in Y}\exp\left(\theta^t f_i(t)\right)\det\left(S_Y\right)}{\sum\limits_{Y'\subseteq\mathcal{Y}(X)}\prod\limits_{i\in Y'}\exp\left(\theta^t f_i(t)\right)\det(S_{Y'})} \tag{2.22}$$

Avec ce petit calcul, on se permet de regarder la concavité de la fonction.

Théorème 9 *: La fonction $\mathcal{L}$ est concave par rapport à θ*

Demo : Voir Annexe

On peut donc calculer le gradient, et on obtient, grâce aux calculs de gradient en annexe, obtenir

$$\boxed{\nabla\mathcal{L}(\theta) \ = \sum_{i\in Y} f_i(X) - \sum_{Y'\subseteq\mathcal{Y}(X)}\mathbb{P}(Y'|X)\sum_{i\in Y'} f_i(X)} \tag{2.23}$$

Cette formule est évidemment très compliquée à calculer dans la pratique, pusique le nombre de calculs augmente de manière exponentielle. Toutefois, on peut déjà avancer que :

$$\boxed{\sum_{Y'\subseteq\mathcal{Y}(X)}\mathbb{P}(Y'|X)\times\sum_{i\in Y'} f_i(X) \ = \sum_{i\in\mathcal{Y}(X)} f_i(X)\times\sum_{Y'\supseteq\{i\}}\mathbb{P}(Y'|X)} \tag{2.24}$$

Ainsi, au lieu de sommer sur tous les sous-ensembles $Y' de \mathcal{Y}'(X)$ puis sur tous les i de Y', on somme sur tous les i de $\mathcal{Y}(X)$ puis sur tous les Y' contenant les {i}.

De plus, $\sum_{Y' \supseteq \{i\}} \mathbb{P}_\theta(Y'|X)$ est en réalité la probabilité marginale d'inclusion de l'élément i dans l'ensemble Y' tiré selon $\mathbb{P}_\theta$. Or on connait la valeur de cette probabilité puisque $\mathbb{P}_\theta(\{i \subseteq Y'|X\}) = K_{i,i}(X)$. On a donc :

$$\boxed{\sum_{Y' \subseteq \mathcal{Y}(X)} \mathbb{P}(Y'|X) \times \sum_{i \in Y'} f_i(X) \; = \sum_{i \in Y'} f_i(X) K_{i,i}(X)} \tag{2.25}$$

Enfin, si on connait la décomposition spectrale, on peut facilement calculer les $K_{i,i}$. Grâce à toutes ces informations, on peut construire un algorithme plutôt efficace pour calculer le gradient.

Algorithm 2 Algorithme 2 : Algorithme de calcul de gradient

Entrée : Doublet (X, Y) et le paramètre θ
On calcule $\mathcal{L}(X, \theta) = q_i(X, \theta) \times S_Y(X) \times q_j(x, \theta)$
On réalise la décomposition spectrale de $\mathcal{L}(X, \theta) = \sum_{n=1}^{N} \lambda_n v_n v_n^t$
for $i \in \mathcal{Y}(X)$ **do**
$K_{i,i} \leftarrow \sum_{n=1}^{N} \frac{\lambda_n}{\lambda_n + 1} \times v_{n,i}^2$
end for
$\nabla \mathcal{L}(\theta) \leftarrow \sum_{i \in Y} f_i(X) - \sum_{i \in \mathcal{Y}(X)} f_i(X) K_{i,i}$ **Sortie :** $\nabla \mathcal{L}(\theta)$

Cet algorithme calcule donc le plus efficacement possible $\nabla \mathcal{L}(\theta)$. En effet, la partie calcul de gradient demande $O(N^2)$ en temps. Malheureusement, le passage à la décomposition spectrale de $\mathcal{L}(X, \theta)$ demande $O(N^3)$ en temps, ce qui fait tout de monde de cet algorithme un algorithme assez lourd.

Chapitre 3

Annexe : Démonstration

3.1 Partie 1 : Processus Ponctuels Determinantaux

3.1.1 Démonstration Théorème 1

Démo : On peut écrire, sachant que l'on a $card(\bar{A}) = k > 0$, et que i correspond à un élément de $\mathcal{Y}$, avec $i \in \bar{A}$. On peut donc écrire que :

$$L + I_{\bar{A}} = \begin{pmatrix} L_{i,i} + 1 & L_{i,\bar{i}} \\ L_{\bar{i},i} & L_{\mathcal{Y}-\{i\}} + I_{\mathcal{Y}-\{i\}-A} \end{pmatrix} \tag{3.1}$$

Ainsi, on peut écrire que l'on a

$$det(L + I_{\bar{A}}) = det(L + I_{A \cup \bar{\{i\}}}) + det(L_{\mathcal{Y}-\{i\}} + I_{\mathcal{Y}-\{i\}-A}) \tag{3.2}$$

$$= \sum_{A \cup \{i\} \subseteq Y \subseteq \mathcal{Y}} det(L_Y) + \sum_{A \subseteq Y \subseteq \mathcal{Y}-\{i\}} det(L_Y) \tag{3.3}$$

$$= \sum_{A \subseteq Y \subseteq \mathcal{Y}} det(L_Y) \tag{3.4}$$

En effet, on peut voir que chaque Y contiens i, et est inclus dans la première somme, ou ne contient pas i est est inclus seulement dans la deuxième somme.

3.1.2 Démonstration Théorème 2 :

Démo : En utilisant le théorème 1, on peut trouver que :

$$\mathbb{P}_L(A \subseteq \mathbf{Y}) = \frac{\sum\limits_{A \subseteq Y \subseteq \mathcal{Y}} det(L_Y)}{\sum\limits_{Y \subseteq \mathcal{Y}} det(L_Y)} \tag{3.5}$$

$$= \frac{det(L + I_{\bar{A}})}{det(L + I)} \tag{3.6}$$

$$= det((L + I_{\bar{A}})(L + I)^{-1}) \tag{3.7}$$

$$\tag{3.8}$$

A ce stade, et sachant que $L(L+I)^{-1} = I - (L+I)^{-1}$ et on a :

$$\mathbb{P}_L(A \subseteq \mathbf{Y}) = det(I_{\bar{A}}(L + I)^{-1} + I - (L + I)^{-1}) \tag{3.9}$$

$$= det(I - I_A(L + I)^{-1}) \tag{3.10}$$

$$= det(I_{\bar{A}} + I_A K) \tag{3.11}$$

3.2 Partie 2 : Inférences

3.2.1 Démonstration Lemme 1 :

Démo : On utilise la multi-linéarité du déterminant sur la première colonne de W_J :

$$\det(W_J) = \det((W_J)_1, .., (W_J)_k) \tag{3.12}$$

$$= \det\left(\sum_{n \in J}((W_n)_1, .., (W_J)_k)\right) \tag{3.13}$$

$$= \sum_{n \in J} \det([(W_n)_1, (W_J)_2, (W_J)_3, .., (W_J)_k]) \tag{3.14}$$

$$= \sum_{n_1 \in J}\left(\sum_{n_2 \in J} \det((W_{n_1})_1, (W_{n_2})_2, ..., (W_J)_k)\right) \tag{3.15}$$

$$\tag{3.16}$$

Or, pour $\forall n \in \{1, .., N\}, W_n$ est de rang 1. Ce qui implique que $n_1 = n_2, ((W_{n_1})_1, (W_{n_2})_2, ..., (W_J)_k) = 0$. On a donc :

$$\det(W_J) = \sum_{n_1 \in J} \sum_{\substack{n_2 \in J \\ n_1 \neq n_2}} \det([(W_{n_1})_1, (W_{n_2})_2, .., (W_J)_k]) \tag{3.17}$$

On réitère cette étape autant de fois que nécessaire, pour chacune des k colonnes, et on obtient :

$$\det(W_J) = \sum_{n_1 \in J} ... \left(\sum_{\substack{n_k \in J \\ (n_1,..,n_k) \text{ distincts}}} \det([(W_{n_1})_1, (W_{n_2})_2, .., (W_{n_k})_k]) \right) \tag{3.18}$$

$$= \sum_{\substack{n_1,..,n_k \in J \\ \text{distincts}}} \det([(W_{n_1})_1, (W_{n_2})_2, .., (W_{n_k})_k]) \tag{3.19}$$

3.2.2 Démonstration Lemme 2 :

Démo : Le but de cette démonstration est de montrer que pour tout ensemble les probabilités marginales sont égales, ce qui va nous permettre de conclure l'égalité des mesures de probabilités :

$$\mathbb{P}_L \text{ et } \frac{1}{\det(L+I)} \sum_{J \subseteq \{1,2,...,N\}} \mathbb{P}^{V_J} \prod_{n \in J} \lambda_n.$$

Déjà, on peut calculer K, la matrice de noyau marginal associée à $\mathbb{P}_L$, et cela nous done :

$$K = \sum_{n=1}^{N} \frac{\lambda_n}{1+\lambda_n} v_n v_n^T \tag{3.20}$$

Ensuite, soit A un ensemble arbitraire, de cardinal k. On note $\{v_n\}_{n \in J}$ le sous espace engendré par les valeurs propres v_n telles que $n \in J \subseteq \mathcal{Y}$. On note aussi K^{V_J} la matrice marginale de projection, $K^{V_J} = \sum_{n \in J} v_n v_n^T$ Comme précédemment, on peut alors définir $K_A^{V_J} = \sum_{n \in J} [v_n v_n^T]_A$ sa réduction. On note $\mathbb{P}^{V_J}$ le DPP associé à K^{V_J}.

Soit $W_{n,A} = [v_n v_n^T]_A$ une suite de matrice, de rang égal à 1, avec $n \in [\![1, N]\!]$, de dimension $k \times k$. On sait qu'avec la définition de K^{V_J}, la distribution de la probabilité marginale de A vaut :

$$\mathbb{P}(A \subseteq Y) = \det(K_A) \tag{3.21}$$

Maintenant que nous avons défini tout ce dont nous avions besoin, nous allons nous intéresser à la probabilité marginale de A, du mélange de DPP élémentaires.

Soit $Y \subseteq \mathcal{Y}$.

$$\frac{1}{\det(L+I)} \sum_{J \subseteq \{1,2,...,N\}} \mathbb{P}^{V_J}(A \subseteq Y) \prod_{n \in J} \lambda_n = \frac{1}{\det(L+I)} \sum_{J \subseteq \{1,2,...,N\}} \det K_A^{V_j} \prod_{n \in J} \lambda_n \tag{3.22}$$

$$= \frac{1}{\det(L+I)} \sum_{J \subseteq \{1,2,...,N\}} \det \left(\sum_{n \in J} [v_n v_n^T]_A \right) \prod_{n \in J} \lambda_n \tag{3.23}$$

$$= \frac{1}{\det(L+I)} \sum_{J \subseteq \{1,2,...,N\}} \det \left(\sum_{n \in J} [W_{n,A}] \right) \prod_{n \in J} \lambda_n \tag{3.24}$$

On peut donc appliquer la formule obtenue dans le lemme 1, afin d'obtenir les égalités suivantes :

$$(2.24) = \frac{1}{\det(L+I)} \sum_{J\subseteq\{1,2,..,N\}} \sum_{\substack{n_1,n_2,...,n_k\in J \\ distinct}} \det([(W_{n_1})_1,(W_{n_2})_2,..,(W_{n_k})_k]) \prod_{n\in J}\lambda_n \tag{3.25}$$

Pour chaque $J \subseteq \mathcal{Y}$, on regarde tous ses k-uplets, et pour chacun d'entre eux, on calcule le déterminant de la famille $([(W_{n_1,A})_1,(W_{n_2,A})_2,..,(W_{n_k,A})_k])$, que l'on multiplie par le produit des valeurs propres de tous les éléments indexés par J. On va donc parcourir tous les k-uplets possibles d'éléments présents dans {1,..N}, sans relation d'ordre, puisqu'on parcourt tous les sous-ensembles J inclus dans $\mathcal{Y}$. On remarque aussi qu'un k-uplet peut être présent dans plusieurs sous ensembles J Par contre, le coefficient multiplicateur $\prod_{N\in J}$ est unique à chaque J.

Pour résumer, cette double somme parcourt tous les k-uplets possibles, sans relation d'ordre, calcule au moins une fois le déterminant de la famille associée, en la multipliant par un coefficient unique à chaque sous ensemble.

$$(1.40) = \frac{1}{\det(L+I)} \sum_{\substack{n_1,n_2,...,n_k=1 \\ distinct}} \det([(W_{n_1})_1,(W_{n_2})_2,..,(W_{n_k})_k]) \sum_{J\supseteq\{n_1,..,n_k\}} \prod_{n\in J}\lambda_n \tag{3.26}$$

Ici, comme expliquer plus haut, on va parcourir tous les k-uplets sans ordre possible, calculer le déterminant pour chacun d'entre eux, et le multiplier par un coefficient qui vaut la somme sur toutes les valeurs du sous-ensemble.

$$\sum_{J\supseteq\{n_1,..,n_k\}} \prod_{n\in J}\lambda_n \equiv \textit{Somme sur tous les } J \textit{ contenant } \{n_1,...,n_k\} \tag{3.27}$$

Nous allons maintenant détailler $\sum_{J\supseteq\{n_1,..,n_k\}} \prod_{n\in J}\lambda_n$, qui est le produit de toutes les valeurs propres des sous-ensembles contenant $\{n_1,...,n_k\}$. On peut donc fixer $\{n_1,..,n_k\}$ et écrire :

$$\sum_{J\supseteq\{n_1,..,n_k\}} \prod_{n\in J}\lambda_n = \sum_{J\supseteq\{n_1,..,n_k\}} \prod_{n\in J/\{n_1,..,n_k\}} (\lambda_{n_1}\times...\times\lambda_{n_k})\lambda_n \tag{3.28}$$

$$= (\lambda_{n_1}\times...\times\lambda_{n_k}) \sum_{J\supseteq\{n_1,..,n_k\}} \prod_{n\in J/\{n_1,...,n_k\}} \lambda_n \tag{3.29}$$

Si on observe tous les sous-ensembles J correspondant $\{n_1,...,n_k\}$, on se rend compte que ce sont des ensembles construit de la forme suivante : $J = \{n_1,...,n_k\}\cup I$, avec $I \in \mathcal{P}(\mathcal{Y}\backslash\{n_1,...,n_k\})$. L'égalité précédente devient alors :

$$\sum_{J\supseteq\{n_1,..,n_k\}} \prod_{n\in J}\lambda_n = (\lambda_{n_1}\times...\times\lambda_{n_k}) \times \sum_{I\subseteq\mathcal{Y}\backslash\{n_1,..,n_k\}} \prod_{n\in I}\lambda_n \tag{3.30}$$

$$= (\lambda_{n_1}\times...\times\lambda_{n_k}) \prod_{n\in\mathcal{Y}\backslash\{n_1,...,n_k\}} (\lambda_n+1) \tag{3.31}$$

$$= \frac{\lambda_{n_1}}{\lambda_{n_1}+1}\times..\times\frac{\lambda_{n_k}}{\lambda_{n_k}+1}\times\prod_{n=1}^{N}(\lambda_n+1) \tag{3.32}$$

On peut donc ré-injecter dans notre équation, et on a donc :

$$(1.40) = \frac{1}{\det(L+I)} \sum_{\substack{n_1,n_2,...,n_k=1 \\ distinct}} \det([(W_{n_1})_1, (W_{n_2})_2, .., (W_{n_k})_k]) \frac{\lambda_{n_1}}{\lambda_{n_1}+1} ... \frac{\lambda_{n_k}}{\lambda_{n_k}+1} \prod_{n=1}^{N} (\lambda_n + 1) \tag{3.33}$$

En appliquant une nouvelle fois le lemme précédent, et en sachant que $\det(L+I) = \prod_{n=1}^{N} (\lambda_n+1)$, on obtient :

$$(2.33) = \det\left(\sum_{n=1}^{N} \frac{\lambda_n}{\lambda_n+1} W_{n,A} \right) \tag{3.34}$$

$$= \det\left(\sum_{n=1}^{N} \frac{\lambda_n}{\lambda_n+1} [v_n v_n^T]_A \right) \tag{3.35}$$

$$= \det K_A \tag{3.36}$$

$$= \mathbb{P}_L(A \subseteq Y) \tag{3.37}$$

Comme $\forall A \subseteq \mathcal{Y}$, les probabilités marginales sont égales, et donc ces deux probabilités sont égales en loi.

3.2.3 Démonstration Lemme 3 :

Démo : Comme K^V a comme rang $card(V)$, $\mathbb{P}^V(Y \subseteq \mathbf{Y}) = 0$ quand $card(\mathbf{Y}) > card(V)$; cela signifie que $card(\mathbf{Y}) \leq card(V)$.

De surcroit, on peut écrire que :

$$\mathbb{E}[card(\mathbf{Y})] = \mathbb{E}\left[\sum_{n=1}^{N} \mathbb{I}_{\mathbf{Y}}(n) \right] \tag{3.38}$$

$$= \sum_{n=1}^{N} \mathbb{E}[\mathbb{I}_{\mathbf{Y}}(n)] \tag{3.39}$$

$$= \sum_{n=1}^{N} \mathbb{P}(\{n\} \subseteq \mathcal{Y}) \tag{3.40}$$

$$= \sum_{n=1}^{N} K_{n,n} = tr(K) \tag{3.41}$$

$$\boxed{\mathbb{E}[card(\mathbf{Y})] = card(V)} \tag{3.42}$$

Ainsi, on a $card(\mathbf{Y}) = card(V)$ presque surement.

3.2.4 Démonstration Lemme 4 :

Démo : On a :

$$\mathbb{E}[\mathbb{P}^{V_J}] = \sum_{J\subseteq\{1,...,N\}} \mathbb{P}^{V_J}(A\subseteq Y)\times\prod_{n\in J}\frac{\lambda_n}{\lambda_n+1}\times\prod_{n\in\mathcal{Y}\setminus J}\left(1-\frac{\lambda_n}{\lambda_n+1}\right) \tag{3.43}$$

$$= \sum_{J\subseteq\{1,...,N\}} \mathbb{P}^{V_J}(A\subseteq Y)\times\prod_{n\in J}\frac{\lambda_n}{\lambda_n+1}\times\prod_{n\in\mathcal{Y}\setminus J}\left(\frac{1}{\lambda_n+1}\right) \tag{3.44}$$

$$= \prod_{n=1}^{N}\left(\frac{1}{\lambda_n+1}\right)\times\sum_{J\subseteq\{1,...,N\}} \mathbb{P}^{V_J}(A\subseteq Y)\times\prod_{n\in J}\lambda_n \tag{3.45}$$

$$= \frac{1}{\det(L+I)}\times\sum_{J\subseteq\{1,...,N\}} \mathbb{P}^{V_J}(A\subseteq Y)\times\prod_{n\in J}\lambda_n \tag{3.46}$$

$$= \mathbb{P}_L(A\subseteq Y) \tag{3.47}$$

Dans les faits, si on tire n'importe quel $J\subseteq\mathcal{Y}$ selon une loi de Bernoulli, de paramètre $\frac{\lambda_n}{\lambda_n+1}$, alors :

$$\boxed{\forall A\subseteq Y, \mathbb{E}[\mathbb{P}^{V_J}] = \mathbb{P}_L(A\subseteq Y)} \tag{3.48}$$

3.2.5 Démonstration Lemme 5 :

Démo : Grâce aux résultats préliminaires, il ne nous reste plus qu'à prouver ce que fait la deuxième boucle. On montre ici que la boucle de l'algorithme bien un élément Y de $\mathcal{Y}$ avec une loi de probabilité $\mathbb{P}^{V_J}$.

On se place dans l'espace préhilbertien $(E, <>)$, de dimension N, engendré par la base canonique $(e_1, ..., e_N)$, munie du produit scalaire usuel $< x, y >= \sum_{k=1}^{N} x_k y_k$. On définit B une matrice dont les lignes sont les vecteurs propres de la matrice de noyau de K^V, ce qui donne que $K^V = B^T B$. On note ici $k = card(V)$ et $\left(B = (v_1 \quad \cdots \quad v_k)^T\right)$. Comme la famille $\{v_i\}_{i=1}^N$ est orthonormale, la famille $\{v_1, ..., v_n\}$ engendre un volume de taille k, dont les composantes dans la base canonique de E sont données par la matrice B. Dans ce cas, la i-ème ligne de B noté B_i donne la projection du vecteur e_i de la base canonique dans le volume engendré la base canonique par la famille $\{v_1, ..., v_n\}$.

Grâce au lemme précédent, on peut peut prévoir avec une probabilité de 1 que $card(Y) = card(V) = k$. On peut donc poser sans perte de généralité $Y = \{1, ..., k\} \subseteq \mathcal{Y}$. De plus, grâce à ce que nous avons expliqué dans la partie sur l'interprétation géométrique du déterminant, on a :

$$\boxed{\forall X\subseteq\mathcal{Y}, \det(K_Y^V) = Vol^2(\{B_i\}_{i\in Y}) = \mathbb{P}^V(Y\subseteq X)} \tag{3.49}$$

Le déterminant de la restriction de la matrice K^V aux éléments Y est égale au carré du volume engendré par la restriction de la famille B_i aux éléments de Y.

Nous allons calculer $Vol(\{B_i\}_{i\in Y})$ pour cela, nous allons utiliser une petite propriété, soit x un vecteur de $Vect(\{v_1, ..., v_n\})$, tel que $< x; e_i >= 0$, alors on a $< x; B_i >= 0$, c'est à dire que pour tout vecteur x de $Vect(\{v_1, ..., v_n\})$, perpendiculaire à e_i, alors il est perpendiculaire à B_i. Nous allons le montrer :

Soit $x = \mu_1 v_1 + ... + \mu_k v_k$, $avec$ $< x; e_i >= 0$

$$< x; B_i > =< \sum_{p=1}^{k} \mu_p v_p; \sum_{q=1}^{k} < e_i; v_q > v_q > \tag{3.50}$$

$$= \sum_{p=1}^{k} \mu_p \left(\sum_{q=1}^{k} < e_i; v_q > (< v_p; v_q >) \right) \tag{3.51}$$

$$= \sum_{p=1}^{k} \left(\sum_{q=1}^{k} \mu_p < e_i; v_q > \delta_{p,q} \right) \tag{3.52}$$

$$= \sum_{p=1}^{k} \left(\mu_p < e_i; v_p > \right) \tag{3.53}$$

$$=< e_i; \sum_{p=1}^{k} (\mu_p v_p) > \tag{3.54}$$

$$=< e_i; x >= 0 \tag{3.55}$$

Pour calculer ce volume engendré par ces vecteurs, nous allons utiliser le fait qu'un volume se calcule en multipliant la base par la hauteur, si Base $\perp$ Hauteur.

Ici, nous allons choisir comme hauteur la première colonne B_1, donc notre hauteur vaut $||B_1||$. Comme on veut l'orthogonalité entre la hauteur et la base, on va multiplier par la projection des vecteurs $[B_i]_{i=1}^k$ dans l'espace perpendiculaire à B_1, ce qui, grâce à notre petite propriété expliquée précédemment revient à projeter dans l'espace perpendiculaire à e_1. Ce qui nous donne :

$$\boxed{Vol(\{B_i\}_{i \in Y}) = ||B_1|| \times Vol(\{Proj_{\perp_{e_1}} \times B_i\}_{i=2}^k)} \tag{3.56}$$

Pour réaliser cette projection, on va procéder de la façon suivante :

— On connait une première base orthonormée de cet espace à e_1. Puisque les vecteurs $[e_2, ..., e_k]$ dont tous perpendiculaires à e_1 et forment une base de cet espace (on peut regarder cela en dimension 3 pour le comprendre : si on prend (e_1, e_2, e_3) base orthonormée de $\mathbb{R}^3$, on a le projeté orthogonal à e_1 est un projeté sur la surface grise et il est évident que les vecteurs (e_2, e_3) forment une base de cette espace).

— Nous pouvons donc déterminer le projeté de notre base orthonormée $\{v_1, ..., v_k\}$ sur cet espace par la formule $p_{\perp_{e_1}}(v_i) = \sum_{p=2}^{k} < v_i; e_p > e_p$. On projette donc cette base de taille k sur cet ensemble de taille $k-1$. Cette famille sera donc liée. Toutefois, comme toute projection entre deux espaces de dimensions différentes et non nulles, $p_{\perp_{e_1}}$ est une application surjective, et comme l'image d'une base par une application surjective donne une famille génératrice, nous avons $(p_{\perp_{e_1}}(v_1), ..., p_{\perp_{e_k}}(v_k))$ qui est une famille génératrice des sous-ensembles perpendiculaires à e_1. De cette famille génératrice, on peut extraire une base du sous ensemble perpendiculaire à e_1. Comme cette base à $k-1$ vecteurs, il suffit de trouver les vecteurs de la famille $(p_{\perp_{e_1}}(v_1), ..., p_{\perp_{e_k}}(v_k))$ qui s'écrit comme une combinaison linéaire des autres vecteurs. On peut pour cela calculer les produits scalaires $<p_{\perp_{e_1}}(v_i); p_{\perp_{e_1}}(v_j) >$ pour tout $i, j \in \{1, .., k\}^2, i \neq j$. Une fois la base identifiée, il ne nous manque plus qu'à orthonormaliser la base grâce au procédé de Gram-Schmidt.

On obtient donc une nouvelle base orthonormale que nous noterons pour des raisons de simplification et sans perte de généralité $V = \{v_2, ..., v_k\}$, base de l'ensemble orthogonal à e_1.

On peut réécrire la famille $\{B_i\}_{i=2}^k$ en $\{proj_{\perp_{e_1}} B_i\}$ qui est l'écriture de la famille $\{e_2, ..., e_k\}$ dans la base $\{v_2, ..., v_k\}$ dont les coefficients s'écrivent :

$$\boxed{\{proj_{\perp_{e_1}} B_i\}_j = \ < v_j; e_i >= (v_j^T e_i)} \tag{3.57}$$

Nous avons donc $Vol(\{B_i\}_{i\in J}) = ||B_1|| \times Vol(\{Proj_{\perp_{e_1}} \times B_i\}_{i=2}^k)$.
On peut donc recommencer le processus pour chaque colonne, on obtient :

$$Vol(\{B_i\}_{i\in J}) = ||B_1|| \times Vol(\{Proj_{\perp_{e_1}} \times B_2\}) \times Vol(\{Proj_{\perp_{e_1,e_2}} \times B_3\}) \times ... \times Vol(\{Proj_{\perp_{e_1,..,e_{k-1}}} \times B_k\}) \tag{3.58}$$

Plaçons nous maintenant dans le seconde boucle de l'algorithme 1 à la j-ième itération. A ce stade, nous avons déjà tiré $j-1$ éléments, qui sont $y_1 = 1, ..., y_{j-1} = j-1$. On remarque alors qu'à la j-ième itération, on a $V = \{v_j, ..., v_k\}$ base orthonormée de l'espace orthogonal à $e_1, ..., e_{j-1}$. De plus,

$$\sum_{p=j}^{k} \left(\sum_{q=j}^{k} (v_q^T e_p)^2 \right) = \sum_{p=j}^{k} \left(\sum_{q=j}^{k} (Proj_{\perp_{e_1,...,e_{j-1}}} B_q)_p^2 \right) \tag{3.59}$$

$$= \sum_{q=j}^{k} \left(\sum_{p=j}^{k} (Proj_{\perp_{e_1,...,e_{j-1}}} B_q)_p^2 \right) \tag{3.60}$$

$$= \sum_{q=j}^{k} ||v_q||^2 \tag{3.61}$$

Comme la famille $(v_j, ..., v_k)$ est orthonormal, $\forall q \in \{j, ..., k\}, ||v_q|| = 1$. On a alors :

$$\sum_{p=j}^{k} \left(\sum_{q=j}^{k} (v_q^T e_p)^2 \right) = \sum_{p=j}^{k} \left(\sum_{v\in V} (v^T e_p)^2 \right) \tag{3.62}$$

$$= k - j + 1 \tag{3.63}$$

$$= card(V) \tag{3.64}$$

On peut donc établir une mesure de probabilité, $\forall j \in \{1, ..., k\}$ sur l'ensemble $\{j, ..., k\}$ et telle que :

$$\forall p \in \{j, ..., k\}, \mathbb{P}_r(p) = \frac{1}{card(V)} \sum_{v\in V} (v^T e_p)^2 \tag{3.65}$$

$$= \frac{1}{k-j+1} ||Proj_{\perp_{e_1,...,e_{j-1}}} \times B_p||^2 \tag{3.66}$$

Les éléments p ont une probabilité d'être tiré proportionnellement à la norme de la colonne. (B_p projeté dans l'espace perpendiculaire à $e_1, .., e_{j-1}$).

Si on généralise notre probabilité, puisque que nous avons posé $Y = \{1, .., k\}$, la probabilité de tirer une séquence $1, ..., k$ vaut l'ordre vaut :

$$\mathbb{P}_r(1) \times \mathbb{P}_r(2) \times ...\mathbb{P}_r(k) = \frac{1}{k}||B_1||^2 \times \frac{1}{k-1}||Proj_{\perp_{e_1}} \times B_2||^2 \times ... \times \frac{1}{1}||Proj_{\perp_{e_1,..,e_{k-1}}} \times B_k||^2 \tag{3.67}$$

$$= \frac{1}{k!} \times Vol^2(\{B_i\}_{i \in Y}) \tag{3.68}$$

Or ici, l'odre ne compte pas, de plus si on permute les colonnes de B, le volume calculé sera toujours la même à la fin, on a donc comme probabilité de tirer $Y = \{1, ..., k\}$:

$$\mathbb{P}_r(Y \subseteq \mathcal{Y}) = \sum_{\sigma \in \mathcal{S}_k} (\mathbb{P}_{r,\sigma}(1), ..., \mathbb{P}_{r,\sigma}(k)) \tag{3.69}$$

$$= k! \times \frac{1}{k!} Vol^2(\{B_i\}_{i \in Y}) \tag{3.70}$$

$$\boxed{\mathbb{P}_r(Y \subseteq \mathcal{Y}) \ = \det(K_Y^V)} \tag{3.71}$$

La boucle 2 de l'algorithme échantillonne donc bien Y selon une loi de probabilité qui est un DPP élémentaire, et $Y \sim \mathbb{P}^V$ ce qui conclut la preuve du théorème.

3.2.6 Démonstration théorème 9

Démo : Grâce à l'équation 2.22, on peut montrer que :

$$\mathcal{L}(\theta) = \underbrace{\theta^t \sum_{i \in Y} f_i(X)}_{1} + \underbrace{\log(\det(S_Y(X)))}_{2} + \underbrace{\left(-\log(\sum_{Y' \subseteq \mathcal{Y}(X)} \exp(\theta^t \sum_{i \in Y'} f_i(X)) \det(S_{Y'}(X))) \right)}_{3} \tag{3.72}$$

Or on sait que :
— 1 est linéaire par rapport à θ donc concave
— 2 est constante par rapport à θ, donc concave
— 3 est la composition d'une fonction concave et d'une fonction affine, on a donc une fonction concave
On a donc $\mathcal{L}$ qui est une fonction concave.

3.2.7 Calcul de gradient

$$\nabla\mathcal{L}(\theta) = \nabla\left(\theta^t \sum_{i \in Y} f_i(X) + \log(\det(S_Y(X))) + \left(-\log(\sum_{Y' \subseteq \mathcal{Y}(X)} \exp(\theta^t \sum_{i \in Y'} f_i(X)) \det(S_{Y'}(X))) \right) \right) \tag{3.73}$$

$$= \sum_{i \in Y} f_i(X) - \sum_{Y' \subseteq \mathcal{Y}(X)} \frac{\det(S_{Y'}(X)) \exp(\theta^t \sum_{i \in Y'} f_i(X))}{\sum_{Y'' \subseteq \mathcal{Y}(X)} \det(S_{Y''}(X)) \exp(\theta^t \sum_{i \in Y''} f_i(X))} \times \left(\sum_{i \in Y'} f_i(X) \right) \tag{3.74}$$

$$= \sum_{i \in Y} f_i(X) - \sum_{Y' \subseteq \mathcal{Y}(X)} \mathbb{P}(Y'|X) \sum_{i \in Y'} f_i(X) \tag{3.75}$$